INTERNATIONAL STANDARD

ISO/IEC 15938-13

First edition
2015-09-01

Information technology — Multimedia content description interface —

Part 13: Compact descriptors for visual search

Technologies de l'information — Interface de description du contenu multimédia —

Partie 13: Descripteurs compacts pour recherche visuelle

Reference number
ISO/IEC 15938-13:2015(E)

AVA Documents
Document Sharing Hub
© ISO/IEC 2015

Full document details: ISO IEC 15938-13-2015

COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2015, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

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

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub

Full document details: ISO IEC 15938-13-2015

Contents

Page

Foreword ... **v**

Introduction ... **vi**

1 Scope ... **1**

2 Terms and definitions ... **1**

3 Symbols and abbreviated terms ... **2**
3.1 General ... 2
3.2 Abbreviations ... 2
3.3 Arithmetic operations ... 3
3.4 Logical operators ... 3
3.5 Relational operators ... 3
3.6 Bitwise operators ... 4
3.7 Assignment ... 4
3.8 Mnemonics ... 4
3.9 Constants ... 4
3.10 Functions ... 4

4 CDVS syntax ... **5**
4.1 Binary representation syntax ... 5
4.2 Descriptor component semantics ... 6

5 CDVS encoding ... **9**
5.1 General ... 9
5.2 Original image preprocessing ... 9
5.3 Interest point detection ... 9
5.3.1 Introduction ... 9
5.3.2 Scale space construction ... 9
5.3.3 Detection of scale-space extrema ... 10
5.3.4 Coordinate refinement to subpixel precision. ... 14
5.3.5 Transformation of coordinates and scale to the converted image resolution ... 17
5.3.6 Elimination of duplicates ... 17
5.3.7 Orientation Assignment ... 17
5.3.8 Interest point characteristics ... 19
5.4 Local feature selection ... 19
5.4.1 Operation ... 19
5.4.2 Descriptor components ... 20
5.5 Local feature description ... 21
5.6 Local feature descriptor aggregation ... 23
5.6.1 Operation ... 23
5.6.2 Descriptor components ... 26
5.7 Local feature descriptor compression ... 27
5.7.1 Operation ... 27
5.7.2 Descriptor components ... 30
5.8 Local feature location compression ... 31
5.8.1 Operation ... 31
5.8.2 Descriptor components ... 36
5.9 Encoding order of compressed local feature descriptors and relevance bits ... 37
5.10 Computation of the number of compressed local feature descriptors at different image descriptor lengths ... 37

Annex A (informative) **CDVS encoder organization** ... **38**

Annex B (normative) **Coefficients for coordinate refinement** ... **39**

Annex C (normative) **Probability values for the feature selection** ... **41**

Annex D (normative) **PCA projection matrix for local feature descriptor aggregation** ... **44**

Documents
Document Sharing Hub

© ISO/IEC 2015 – All rights reserved

Full document details: ISO IEC 15938-13-2015

Annex E (normative) **GMM parameters for local feature descriptor aggregation** **55**
Annex F (normative) **Gaussian function selection parameters for local feature descriptor aggregation** **135**
Annex G (normative) **Bit selection masks for local feature descriptor aggregation** **136**
Annex H (normative) **Scalar quantization thresholds for local feature descriptor compression** **138**
Annex I (normative) **Histogram count arithmetic coding model probabilities** **142**
Annex J (normative) **Histogram map arithmetic coding model probabilities** **144**
Annex K (informative) **CDVS decoding** **145**

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

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 meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*, SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ISO/IEC 15938 consists of the following parts, under the general title *Information technology — Multimedia content description interface*:

— *Part 1: Systems*

— *Part 2: Description definition language*

— *Part 3: Visual*

— *Part 4: Audio*

— *Part 5: Multimedia description schemes*

— *Part 6: Reference software*

— *Part 7: Conformance testing*

— *Part 8: Extraction and use of MPEG-7 descriptions*

— *Part 9: Profiles and levels*

— *Part 10: Schema definition*

— *Part 11: MPEG-7 profile schemas*

— *Part 12: Query format*

— *Part 13: Compact descriptors for visual search*

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Introduction

This International Standard, also known as "Multimedia Content Description Interface," provides a standardized set of technologies for describing multimedia content. It addresses a broad spectrum of multimedia applications and requirements by providing a metadata system for describing the features of multimedia content.

The following are specified in this International Standard:

— **Description schemes (DS)** describe entities or relationships pertaining to multimedia content. Description schemes specify the structure and semantics of their components, which may be Description Schemes, descriptors, or datatypes.

— **Descriptors (D)** describe features, attributes, or groups of attributes of multimedia content.

— **Datatypes** are the basic reusable datatypes employed by description schemes and descriptors.

— **Systems tools** support delivery of descriptions, multiplexing of descriptions with multimedia content, synchronization, file format, and so forth.

This International Standard is subdivided into 13 parts:

— **Part 1 — Systems**: specifies the tools for preparing descriptions for efficient transport and storage, compressing descriptions, and allowing synchronization between content and descriptions.

— **Part 2 — Description definition language**: specifies the language for defining the International Standard set of description tools (DSs, Ds, and datatypes) and for defining new description tools.

— **Part 3 — Visual**: specifies the description tools pertaining to visual content.

— **Part 4 — Audio**: specifies the description tools pertaining to audio content.

— **Part 5 — Multimedia description schemes**: specifies the generic description tools pertaining to multimedia including audio and visual content.

— **Part 6 — Reference software**: provides a software implementation of the International Standard.

— **Part 7 — Conformance testing**: specifies the guidelines and procedures for testing conformance of implementations of the International Standard.

— **Part 8 — Extraction and use of MPEG-7 descriptions**: provides guidelines and examples of the extraction and use of descriptions.

— **Part 9 — Profiles and levels**: provides guidelines and standard profiles.

— **Part 10 — Schema definition**: specifies the schema using description definition language.

— **Part 11 — Profile Schemas**: listing of profile schemas using description definition language.

— **Part 12 — Query format**: contains the tools of the MPEG Query Format (MPQF).

— **Part 13 — Compact descriptors for visual search**: specifies an image description tool for visual search applications.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Information technology — Multimedia content description interface —

Part 13: Compact descriptors for visual search

1 Scope

The structure of this part of ISO/IEC 15938 is as follows. Clauses 2 and 3 specify the terms, abbreviations, symbols, and conventions used in the International Standard. Clause 4 specifies the binary representation syntax and descriptor component semantics for a CDVS image descriptor. Clause 5 specifies the extraction and encoding process for a CDVS image descriptor. Annexes A-J specify information relevant to the encoding process of Clause 5. Annex K contains an informative description of the decoding process of a CDVS image descriptor.

This part of the MPEG-7 standard specifies an image description tool designed to enable efficient and interoperable visual search applications, allowing visual content matching in images. Visual content matching includes matching of views of objects, landmarks, and printed documents, while being robust to partial occlusions as well as changes in viewpoint, camera parameters, and lighting conditions.

2 Terms and definitions

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

2.1
image descriptor
descriptor extracted from one image

2.2
image descriptor length
size of an image descriptor in bytes

Note 1 to entry: This International Standard specifies six average (i.e. over a large number of images) image descriptor lengths, i.e. 512 bytes, 1024 bytes, 2048 bytes, 4096 bytes, 8192 bytes, and 16384 bytes, and the encoding process for each image descriptor length.

2.3
original image
input image to the image descriptor encoder

2.4
converted image
image which is a spatially resampled version of the original image and from which the image descriptor is extracted

2.5
pixel
indexable element of the original image or the converted image, comprising spatial coordinates and a luminance value

2.6
interest point
point in an image showing detection stability under local and global perturbations in the image domain, including perspective transformations, changes in image scale, and illumination variations

© ISO/IEC 2015 – All rights reserved

AVA Documents Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

2.7
local region
area in an image in the neighbourhood of an interest point, used to generate local feature descriptors

2.8
cell
each of the 4x4 subdivisions of a local region

2.9
cell histogram
histogram of gradients computed from the cell

2.10
local feature descriptor
descriptor of a local region, computed from the cell histograms

2.11
global descriptor
aggregation of local feature descriptors into a compact representation of the image

2.12
compressed local feature descriptor
compressed representation of a local feature descriptor

2.13
interest point coordinate
horizontal and vertical pixel coordinates indicating the position of an interest point in the converted image resolution, rounded to the nearest integer

2.14
location quantization factor
size of the blocks of the spatial grid superimposed on top of the converted image in order to obtain quantized interest point coordinates' values

2.15
histogram map
binary representation of the converted image scaled down by the location quantization factor, indicating whether each bin generated through the superimposition of the spatial grid on top of the converted image is populated with at least one interest point

2.16
histogram count
vector indicating the number of interest points that populate each non-empty bin generated through the superimposition of a spatial grid on top of the converted image

3 Symbols and abbreviated terms

3.1 General

NOTE The mathematical operators used in this part of ISO/IEC 15938 are similar to those used in the C programming language. Unless otherwise indicated, all the arithmetic operations are performed with real values. Numbering and counting conventions generally begin from 0.

3.2 Abbreviations

CDVS Compact Descriptors for Visual Search

LoG Laplacian-of-Gaussian

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

MPEG	Moving Picture Experts Group
MPEG-7	ISO/IEC 15938

3.3 Arithmetic operations

+	Addition
-	Subtraction (as a binary operator) or negation (as a unary operator)
++	Increment by 1, i.e. x++ is equivalent to x=x+1
--	Decrement by 1, i.e. x-- is equivalent to x=x-1
+=	Increment by value, i.e. x+=y is equivalent to x=x+y
-=	Decrement by value, i.e. x-=y is equivalent to x=x-y
*****	Multiplication (in binary representation syntax and pseudo-code) or convolution (elsewhere)
×	Multiplication
·	Multiplication
/	Division
÷	Division
%	Modulo operator

3.4 Logical operators

||	Logical OR
v	Logical OR
&&	Logical AND
∧	Logical AND
!	Logical NOT

3.5 Relational operators

>	Greater than
>=	Greater than or equal to
≥	Greater than or equal to
<	Less than
<=	Less than or equal to
≤	Less than or equal to
==	Equal to
!=	Not equal to

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

3.6 Bitwise operators

|	OR
&	AND

3.7 Assignment

=	Assignment operator
←	Assignment operator

3.8 Mnemonics

The following mnemonics are defined to describe the different data types used in the coded bitstream.

bslbf	Bit string, left bit first, where "left" is the order in which bits are written in the bitstream.
uimsbf	Unsigned integer, most significant bit first.
vlclbf	Variable length code, left bit first, where "left" refers to the order in which the VLC codes are written in the bitstream and where the byte order of multibyte words is most significant byte first.

3.9 Constants

π	3.141 592 653 58…
e	2.718 281 828 45…

3.10 Functions

$\log_n(\,)$	Base-n logarithm
max( )	Maximum value in argument list
min( )	Minimum value in argument list
sgn( )	Sign function, i.e. sgn(x) = -1, 0 or +1 when x < 0, x == 0 or x > 0, respectively
$\lvert\ \rvert$	Absolute value of scalar or a vector norm
$\lfloor\ \rfloor$	Floor function which returns the maximum integer number less than or equal to the given real number
$\lceil\ \rceil$	Ceiling function which returns the minimum integer number greater than or equal to the given real number
$\downarrow_{2x2}$	Downsamples an image by keeping only the even rows and even columns of the image, without anti-alias filtering

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

4 CDVS syntax

4.1 Binary representation syntax

CDVSDescriptor {	Number of bits	Mnemonics
VersionID	3	bslbf
ModeID	8	uimsbf
GlobalHasBitSelection	1	bslbf
GlobalHasVariance	1	bslbf
RelevanceBitsPresent	1	bslbf
ReservedBits	2	bslbf
OriginalImageXResolution	16	uimsbf
OriginalImageYResolution	16	uimsbf
NumberOfLocalDescriptors	16	uimsbf
if(NumberOfLocalDescriptors>0) {		
for(k=0; k<NumberOfGlobalFunctions; k++) {		
GlobalFunctionPresent[k]	1	bslbf
}		
if(GlobalHasBitSelection) {		
for(k=0; k<NumberOfGlobalFunctions; k++) {		
if(GlobalFunctionPresent[k]) {		
GlobalFunctionMeanVector[k]	24	bslbf
}		
}		
}		
else {		
for(k=0; k<NumberOfGlobalFunctions; k++) {		
if(GlobalFunctionPresent[k]) {		
GlobalFunctionMeanVector[k]	32	bslbf
}		
}		
}		
if(GlobalHasVariance) {		
for(k=0; k<NumberOfGlobalFunctions; k++) {		
if(GlobalFunctionPresent[k]) {		
GlobalFunctionVarianceVector[k]	32	bslbf
}		
}		
}		
HistogramCountSize	16	uimsbf
HistogramMapSizeX	16	uimsbf
HistogramMapSizeY	16	uimsbf
HistogramCount (arithmetically coded block; see 5.8)	>=0	vlclbf

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

CDVSDescriptor {	Number of bits	Mnemonics
HistogramMap (arithmetically coded block; see 5.8)	>=0	vlclbf
NumberOfElementGroups	6	uimsbf
for(k=0; k<NumberOfLocalDescriptors; k++) {		
for(n=0; n<(4*NumberOfElementGroups); n++) {		
LocalDescriptorElements[k][n]	1-2	vlclbf
}		
}		
if(RelevanceBitsPresent) {		
for(k=0; k<NumberOfLocalDescriptors; k++)		
RelevanceBits[k]	1	bslbf
}		
}		
BitStuffing	0-7	vlclbf
}		
}		

VersionID = 1

NumberOfGlobalFunctions = 512

4.2 Descriptor component semantics

VersionID

This descriptor component specifies the CDVSDescriptor version. In this International Standard ISO/IEC 15938-13:2015, VersionID = 1.

ModeID

This descriptor component specifies the image descriptor length. There are six image descriptor lengths, and their corresponding ModeID values are shown in Table 1 below.

Table 1 — ModeID values for the six image descriptor lengths

Image descriptor length	ModeID
512 bytes	1
1024 bytes	2
2048 bytes	3
4096 bytes	4
8192 bytes	5
16384 bytes	6

GlobalHasBitSelection

This descriptor component specifies whether bit selection is applied or not to the GlobalFunctionMeanVector of each of the Gaussian functions which are present in the global descriptor of an image descriptor. If GlobalHasBitSelection == 1 then bit selection is applied, and if GlobalHasBitSelection == 0 then bit selection is not applied. More details are provided in 5.6.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

GlobalHasVariance

This descriptor component specifies whether the GlobalFunctionVarianceVector of each of the Gaussian functions which are present in the global descriptor of an image descriptor appears in the bitstream or not. If GlobalHasVariance == 1 then GlobalFunctionVarianceVector appears in the bitstream, and if GlobalHasVariance == 0 then GlobalFunctionVarianceVector does not appear in the bitstream. More details are provided in 5.6.

RelevanceBitsPresent

This descriptor component specifies if a relevance bit for each compressed local feature descriptor is present in the bitstream. If RelevanceBitsPresent == 1 then the relevance bits are present in the bitstream, and if RelevanceBitsPresent == 0 then the relevance bits are not present in the bitstream. More details are provided in 5.4.

ReservedBits

This descriptor component comprises two bits which are reserved for future use and they shall both be set to 0.

OriginalImageXResolution

This descriptor component specifies the width (in pixels) of the original image.

OriginalImageYResolution

This descriptor component specifies the height (in pixels) of the original image.

NumberOfLocalDescriptors

This descriptor component specifies the number of compressed local feature descriptors which are present in the bitstream. More details are provided in 5.10. NumberOfLocalDescriptors == 0 indicates that no local features were identified in the image.

NumberOfGlobalFunctions

This descriptor component specifies the maximum number of Gaussian functions used in the global descriptor and has a value NumberOfGlobalFunctions = 512. More details are provided in 5.6.

GlobalFunctionPresent

This descriptor component specifies a 1-D array of size NumberOfGlobalFunctions indicating which Gaussian functions are present in the global descriptor of a particular image descriptor. If a Gaussian function is present in the global descriptor the corresponding value in the array is 1, otherwise it is 0. More details are provided in 5.6.

GlobalFunctionMeanVector

This descriptor component specifies a 1-D array of size equal to the number of Gaussian functions which are present in the global descriptor, i.e. those Gaussian functions with a corresponding value of 1 in GlobalFunctionPresent. Each entry in the array is the binarized mean vector of the corresponding global descriptor Gaussian function, and the length of each vector is 24 bits if GlobalHasBitSelection == 1 and 32 bits if GlobalHasBitSelection == 0. More details are provided in 5.6.

GlobalFunctionVarianceVector

This descriptor component specifies a 1-D array of size equal to the number of Gaussian functions which are present in the global descriptor, i.e. those Gaussian functions with a corresponding value of 1 in GlobalFunctionPresent. Each entry in the array is the binarized variance vector of the corresponding global descriptor Gaussian function. More details are provided in 5.6.

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

HistogramCountSize

This descriptor component specifies the histogram count vector length for location coding. More details are provided in 5.8.

HistogramMapSizeX

This descriptor component specifies the horizontal x resolution of the histogram map for location coding. More details are provided in 5.8.

HistogramMapSizeY

This descriptor component specifies the vertical y resolution of the histogram map for location coding. More details are provided in 5.8.

HistogramCount

This descriptor component specifies a vector for location coding, containing the number of non-zero elements for each non-null block of the histogram map. More details are provided in 5.8.

HistogramMap

This descriptor component specifies a 2D-array for location coding, containing a block representation of the converted image. Each block can assume a binary value, indicating the occurrence or not of interest points within that block. The array is scanned according a procedure described in 5.8. The scanning terminates when all the non-null elements of the Histogram Map are encoded. More details are provided in 5.8.

NumberOfElementGroups

This descriptor component specifies the number of element groups in each compressed local feature descriptor. Each element group contains four elements and the number of elements in each compressed local feature descriptor is given by 4×NumberOfElementGroups. More details are provided in 5.7.

LocalDescriptorElements

This descriptor component specifies a 2-D array of compressed local feature descriptor elements. The size of the first dimension is NumberOfLocalDescriptors and the size of the second dimension is 4×NumberOfElementGroups. LocalDescriptorElements[k][n] is the nth element of the kth compressed local feature descriptor. For each compressed local feature descriptor, its elements are ordered as described in 5.7.

The compressed local feature descriptors themselves are ordered as described in 5.9.

RelevanceBits

This descriptor component specifies a 1-D array of size NumberOfLocalDescriptors indicating which compressed local feature descriptors correspond to the top 300 local features as determined in 5.4. If the kth local feature is one of the top 300 local features, then RelevaceBits[k] is set to 1, otherwise it is set to 0. If NumberOfLocalDescriptor<300, then all the values in RelevanceBits are set to 1. More details are provided in 5.4.

The relevance bits are ordered in the same order as the descriptors in LocalDescriptorElement, as described in 5.9.

BitStuffing

This descriptor component specifies stuffing bits (a sequence of '1's) to align the descriptor to a byte boundary.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

5 CDVS encoding

5.1 General

This clause specifies the encoder operations for computing an image descriptor. A simplified diagram of a complete CDVS encoder implementing these encoding operations is presented in informative Annex A.

5.2 Original image preprocessing

The original image is a luminance raster image containing values in the interval [0, 255] where increasing values correspond to increasing luminance. The exact mapping of luminance values within this interval is beyond the scope of the standard. If at least one of the dimensions of the original image is greater than 640 pixels then the original image shall be spatially resampled, maintaining the aspect ratio, so that the largest of the vertical and horizontal image dimensions is equal to 640 pixels, to obtain a converted image $J(x, y)$, in which $x \in \{0,\ldots,X-1\}$ and $y \in \{0,\ldots,Y-1\}$ are the horizontal and vertical pixel coordinates respectively, X and Y the pixel horizontal and vertical image dimensions respectively, and with coordinates (0,0) located at the top left corner of the image. For this resampling operation, a Lanczos filter with a = 3 should be used. If both the dimensions of the original image are no greater than 640 pixels, then no spatial resampling is performed and the content of the converted image shall be the same as the content of the original image.

5.3 Interest point detection

5.3.1 Introduction

This operation is performed using the ALP (A Low-degree Polynomial) detector. In order to find interest points, ALP approximates the result of the LoG filtering by means of polynomials, used to find extrema in the scale space and to refine the spatial position of the detected points.

5.3.2 Scale space construction

Let g denote the Gaussian kernel in two dimensions with positive scale parameter σ

$$g(x,y,\sigma)=\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

The filtering operations shall be done at 4 scales with values for the σ parameter in an exponentially increasing sequence

$$\sigma_k=\sigma_0\cdot 2^{\frac{k}{2}}, k=0,\ldots,3 \tag{2}$$

as provided in Table 2 below.

Table 2 — Values of the scale parameter

k	σ_k
0	1,600000
1	2,262742
2	3,200000
3	4,525483

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Interest points shall be identified by means of the scale-normalized Laplacian-of-Gaussian (LoG) kernel, which is realized as the convolution

$$h(\cdot,\cdot,\sigma)=\sigma^2\cdot\begin{bmatrix}0 & 1 & 0\\ 1 & -4 & 1\\ 0 & 1 & 0\end{bmatrix}*g(\cdot,\cdot,\sigma) \tag{3}$$

where g in this case is a truncated and spatially discrete Gaussian function, with width equal to $2\cdot\lceil 4\sigma\rceil+1$ where $\lceil\ \rceil$ denotes the ceiling function.

For the converted image $J(x, y)$ in which $x\in\{0,\ldots,X-1\}$ and $y\in\{0,\ldots,Y-1\}$ are the horizontal and vertical pixel coordinates respectively, X and Y the pixel horizontal and vertical image dimensions respectively, and with coordinates (0,0) located at the top left corner of the image $J(x, y)$, scale space shall be constructed as follows.

The image shall be processed in a scale space representation obtained by Gaussian blur with different scale factors σ. The scale space shall be structured in a number Q of octaves,

$$Q=\max\{\lfloor\log_2(\max\{X,Y\})-3\rfloor,1\} \tag{4}$$

with $\lfloor\ \rfloor$ denoting the floor function.

For each octave in scale space, 4 images shall be produced by filtering of a first image I with a Gaussian kernel. In any octave, these images shall be obtained by the following filtering operations

$$\begin{aligned}I_0&=I\\ I_1&=I_0*g(\delta_1)\\ I_2&=I_0*g(\delta_2)\\ I_3&=I_0*g(\delta_3)\end{aligned} \tag{5}$$

with the parameter $\delta_n=\sqrt{\sigma_n^2-\sigma_0^2}$ for n = 1,...,3. The first image in the first octave shall be obtained as

$$I=J*g(\sigma_0) \tag{6}$$

and in all other octaves the first image shall be obtained by downsampling

$$I=\downarrow_{2x2}(I_2^{prev}) \tag{7}$$

where I_2^{prev} denotes image I_2 in the previous octave. Anti-alias filtering shall not be applied since the downsampling is applied to images which are already low-pass filtered.

Additionally, in any octave 4 images shall be produced by scale-normalized Laplacian filtering of the Gaussian-filtered images

$$\begin{aligned}L_0&=\sigma_0^2\cdot I_0*f\quad,\quad L_1=\sigma_1^2\cdot I_1*f\\ L_2&=\sigma_2^2\cdot I_2*f\quad,\quad L_3=\sigma_3^2\cdot I_3*f\end{aligned} \tag{8}$$

where $f=\begin{bmatrix}0 & 1 & 0\\ 1 & -4 & 1\\ 0 & 1 & 0\end{bmatrix}$ is the discrete Laplacian operator.

5.3.3 Detection of scale-space extrema

For each octave, two intervals in scale are defined. One is the outer interval $\overline{\Omega}$ and it shall contain a smaller one called the inner ***interval Ω***.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

The outer interval has the lowest and highest scales σ_0 and σ_3 as boundaries $\overline{\Omega}=[1.6,4.525483]$ and the inner interval has the boundaries $\overline{\Omega}=[1.7,4.0]$.

For each pixel (x, y) in the image, a polynomial approximation to the scale-space function

$$p(x,y,\sigma)=\alpha_3(x,y)\sigma^3+\alpha_2(x,y)\sigma^2+\alpha_1(x,y)\sigma+\alpha_0(x,y) \tag{9}$$

shall be searched for a local extremum over the outer interval $\overline{\Omega}$. The coefficients shall be obtained by computing weighted sums of the images $L_0,\ldots,L_3$

$$\begin{aligned}
\alpha_3(x,y)&=\sum_{k=0}^{K-1} a_k \cdot L_k(x,y)\\
\alpha_2(x,y)&=\sum_{k=0}^{K-1} b_k \cdot L_k(x,y)\\
\alpha_1(x,y)&=\sum_{k=0}^{K-1} c_k \cdot L_k(x,y)\\
\alpha_0(x,y)&=\sum_{k=0}^{K-1} d_k \cdot L_k(x,y)
\end{aligned} \tag{10}$$

where the coefficients a_k, b_k, c_k, d_k, corresponding to the 4 predefined scales σ_k, k =0,...,3, are listed in Table 3.

Table 3 — Coefficients for the equations for polynomial approximation

k	a_k	b_k	c_k	d_k
0	-0,2464	2,5021	-8,2007	8,6432
1	0,4934	-4,5636	12,9824	-10,8424
2	-0,2717	2,0108	-4,0449	2,1204
3	0,0140	0,1549	-1,0565	1,3886

In this manner, the polynomial approximation is obtained by filtering the original image with a weighted sum of Laplacian-of-Gaussian filters

$$\sigma^2 \cdot f * \sum_{k=0}^{3}(a_k\sigma^3+b_k\sigma^2+c_k\sigma+d_k)\cdot g(\sigma_k) \tag{11}$$

where each of the 4 weights is a polynomial in σ, as illustrated in Figure 1.

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

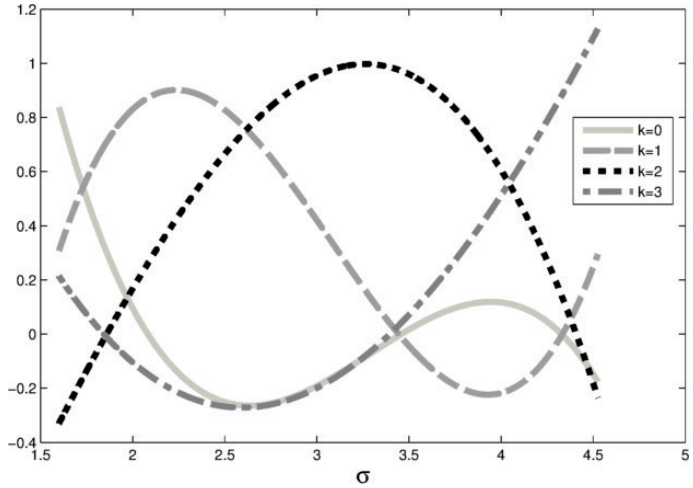


Figure 1 — Polynomial weights for approximating the scale-space function

The coefficients are computed by minimizing the approximation error

$$f * g(\sigma) - \sum_{k=0}^{3} (a_k \sigma^3 + b_k \sigma^2 + c_k \sigma + d_k) \cdot f * g(\sigma_k) \tag{12}$$

over a set of scales contained within the outer interval. Figure 2 depicts a Laplacian-of-Gaussian filter with σ = 2.5 and its approximation.

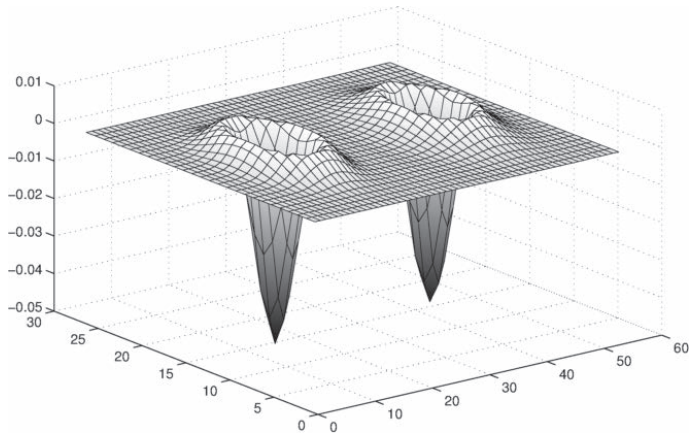


Figure 2 — Exact and approximated Laplacian-of-Gaussian filters at scale 2.5

A tentative scale $\sigma^*(x, y)$ shall be associated to each pixel location *x,y* as the most extreme over the outer interval $\overline{\Omega}$,

$$\sigma^*(x, y) = \arg\max_{\sigma \in \overline{\Omega}} p(x, y, \sigma) \tag{13}$$

or

$$\sigma^*(x, y) = \arg\min_{\sigma \in \overline{\Omega}} p(x, y, \sigma) \tag{14}$$

whichever of the two alternatives has the greatest absolute value. Therefore, for all pixels *x,y* such that

$$\alpha_2^2(x, y) - 3\alpha_1(x, y)\alpha_3(x, y) > 0 \tag{15}$$

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

only the ones that

$$\sigma^*(x,y)=\frac{-\alpha_2(x,y)+\sqrt{\alpha_2^2(x,y)-3\alpha_1(x,y)\alpha_3(x,y)}}{3\alpha_3(x,y)}\in\overline{\Omega} \tag{16}$$

or

$$\sigma^*(x,y)=\frac{-\alpha_2(x,y)-\sqrt{\alpha_2^2(x,y)-3\alpha_1(x,y)\alpha_3(x,y)}}{3\alpha_3(x,y)}\in\overline{\Omega} \tag{17}$$

shall be considered. Among the considered pixels, those with positive solutions σ^* larger than the polynomial at any boundary of the outer interval

$$p\left(\sigma^*(x,y)\right)>\max\{p(x,y,\sigma_0),p(x,y,\sigma_3)\} \tag{18}$$

as well as all pixels with negative solutions σ^* smaller than the polynomial at any boundary of the outer interval

$$p\left(\sigma^*(x,y)\right)<\min\{p(x,y,\sigma_0),p(x,y,\sigma_3)\} \tag{19}$$

shall be accepted as candidates, forming triples $\{x,y,\sigma^*(x,y)\}$. The other candidates are eliminated from further processing in the present octave. This mechanism eliminates also solutions that are not local extrema, i. e. with second derivative equal to zero.

Those candidates for which the solutions σ^* are within the inner interval $\sigma^*(x,y)\in\Omega$ shall be subjected to further processing, the other candidates being eliminated from further processing in the present octave.

Thereafter, any remaining candidate $\{x,y,\sigma^*(x,y)\}$ is eliminated from further processing in the present octave if the absolute value of the polynomial is below a first threshold equal to 0.4, i.e.

$$\left|p(x,y,\sigma^*(x,y))\right|<\theta_1 \text{ with } \theta_1=0.4 \tag{20}$$

Thereafter, any remaining candidate $\{x,y,\sigma^*(x,y)\}$ is eliminated from further processing in the present octave if the second derivative of the scale space function with regard to σ is below a second threshold set to 0.4, i.e.

$$\frac{\partial^2}{\partial\sigma^2}p(x,y,\sigma^*(x,y))=6\alpha_3(x,y)\sigma^2+2\alpha_2(x,y)<\theta_2 \text{ with } \theta_2=0.4 \tag{21}$$

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Thereafter, any remaining candidate $\{x, y, \sigma^*(x, y)\}$ is eliminated from further processing in the present octave if the polynomial value $p(x, y, \sigma^*(x, y))$ is surpassed by the polynomial value of any remaining candidates among its 8-neighbours. Specifically, for any $m \in \{-1,0,1\}$ and any $n \in \{-1,0,1\}$ excluding the combination $(m,n) = (0,0)$, if

$$p(x, y, \sigma^*(x, y)) \le p(x+m, y+n, \sigma); \sigma \in \overline{\Omega} \text{ when } p(x, y, \sigma^*(x, y)) \text{ is a maximum} \tag{22}$$

or

$$p(x, y, \sigma^*(x, y)) \ge p(x+m, y+n, \sigma); \sigma \in \overline{\Omega} \text{ when } p(x, y, \sigma^*(x, y)) \text{ is a minimum} \tag{23}$$

then the candidate is eliminated.

The remaining candidates $\{x, y, \sigma^*(x, y)\}$ are input to the next processing step.

5.3.4 Coordinate refinement to subpixel precision.

For the position refinement 9 pre-defined positions shall be used: all 9 combinations of $u \in \{-1,0,1\}$ and $v \in \{-1,0,1\}$, corresponding to shifts of the LoG kernels in the xy plane.

Firstly, candidates at local edges in the polynomial $p(x, y, \sigma^*(x, y))$ shall be eliminated by the following test:

The 3 × 3 pixels around any candidate are computed at the scale $\sigma^*(x, y)$ of the candidate

$$P(x, y, \sigma^*(x, y)) = \begin{bmatrix} p(x-1, y-1, \sigma^*(x, y)) & p(x, y-1, \sigma^*(x, y)) & p(x+1, y-1, \sigma^*(x, y)) \\ p(x-1, y, \sigma^*(x, y)) & p(x, y, \sigma^*(x, y)) & p(x+1, y, \sigma^*(x, y)) \\ p(x-1, y+1, \sigma^*(x, y)) & p(x, y+1, \sigma^*(x, y)) & p(x+1, y+1, \sigma^*(x, y)) \end{bmatrix} \tag{24}$$

For these pixels, three quantities shall be computed

$$\begin{aligned} p_{xx} &= P_{21} - 2P_{22} + P_{23}, \\ p_{yy} &= P_{12} - 2P_{22} + P_{32}, \\ p_{xy} &= \frac{P_{11} + P_{33} - P_{31} - P_{13}}{4} \end{aligned} \tag{25}$$

where P is shorthand for $P(x, y, \sigma^*(x, y))$ and P_{ij} denotes the element in row i and column j of P. The candidate is eliminated if the following quantity ρ exceeds a threshold equal to 12, i.e.

$$\rho = \left| \frac{(p_{xx} - p_{yy})^2}{p_{xx} \cdot p_{yy} - p_{xy}^2} \right| > \theta_3 \text{ with } \theta_3 = 12 \tag{26}$$

This number is the ratio of the squared trace of the Hessian (at the scale and location of the interest point) and the determinant of the same Hessian. It is related to the ratio of principal curvatures r as $\rho = (r + 1)^2/r$.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

For each remaining candidate (x, y, σ^*), a polynomial approximation to the scale-space function in the displacement parameters u,v

$$q(u,v;x,y\sigma^*) = \beta_5(x,y,\sigma^*)u^2 + \beta_4(x,y,\sigma^*)v^2 + \beta_3(x,y,\sigma^*)uv + \beta_2(x,y,\sigma^*)u + \beta_1(x,y,\sigma^*)v + \beta_0(x,y,\sigma^*) \tag{27}$$

shall be searched for a local extremum. The coefficients are derived from the matrix $P(x,y,\sigma^*(x,y))$ associated to the candidate (x, y, σ^*), as in the previous equations. Any coefficient is a weighted sum, in which the correspondence between term number k and the row number i and column number j is given by Table 4, found below.

Table 4 — Mapping from term number k to row number i(k) and column number j(k)

k	i(k)	j(k)
1	1	1
2	2	1
3	3	1
4	1	2
5	2	2
6	3	2
7	1	3
8	2	3
9	3	3

The coefficients for the candidate (x, y, σ^*) are thus given by weighted sums with K=9

$$\begin{aligned}
\beta_5(x,y,\sigma^*) &= \sum_{k=1}^{K} a_k \cdot P_{i(k),j(k)} \\
\beta_4(x,y,\sigma^*) &= \sum_{k=1}^{K} b_k \cdot P_{i(k),j(k)} \\
\beta_3(x,y,\sigma^*) &= \sum_{k=1}^{K} c_k \cdot P_{i(k),j(k)} \\
\beta_2(x,y,\sigma^*) &= \sum_{k=1}^{K} d_k \cdot P_{i(k),j(k)} \\
\beta_1(x,y,\sigma^*) &= \sum_{k=1}^{K} e_k \cdot P_{i(k),j(k)} \\
\beta_0(x,y,\sigma^*) &= \sum_{k=1}^{K} f_k \cdot P_{i(k),j(k)}
\end{aligned} \tag{28}$$

where $P_{i(k),j(k)}$ is shorthand for element $i(k),j(k)$ of the matrix $P(x,y,\sigma^*(x,y))$.

The coefficients a_k, b_k, c_k, d_k, e_k, f_k are dependent on scale; there are 4 sets and the one corresponding to the nearest neighbor to the scale σ among $\sigma_0,\ldots,\sigma_3$ shall be used. Normative Annex B provides the coefficient sets.

NOTE The coefficients are distinct from those contained in Table 3.

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

The polynomial q may be written (in shorthand, omitting the variables x, y, σ^*) as

$$q(u,v) = \sum_{k=1}^{K} (a_k u^2 + b_k v^2 + c_k uv + d_k u + e_k v + f_k) \cdot P_{i(k),j(k)} \tag{29}$$

The coefficients form a polynomial that provide an interpolation between shifted kernels. Indeed, the coefficients are computed to minimize the approximation error in

$$f * g(x-u, y-v, \sigma) \approx \sum_{k=1}^{K} (a_k u^2 + b_k v^2 + c_k uv + d_k u + e_k v + f_k) \cdot f * g(x+i(k)-2, y+j(k)-2) \tag{30}$$

over a set of displacements $\{(u,v)\} \subseteq [-1,1]^2$. The polynomial functions for all 9 terms are illustrated in Figure 3.

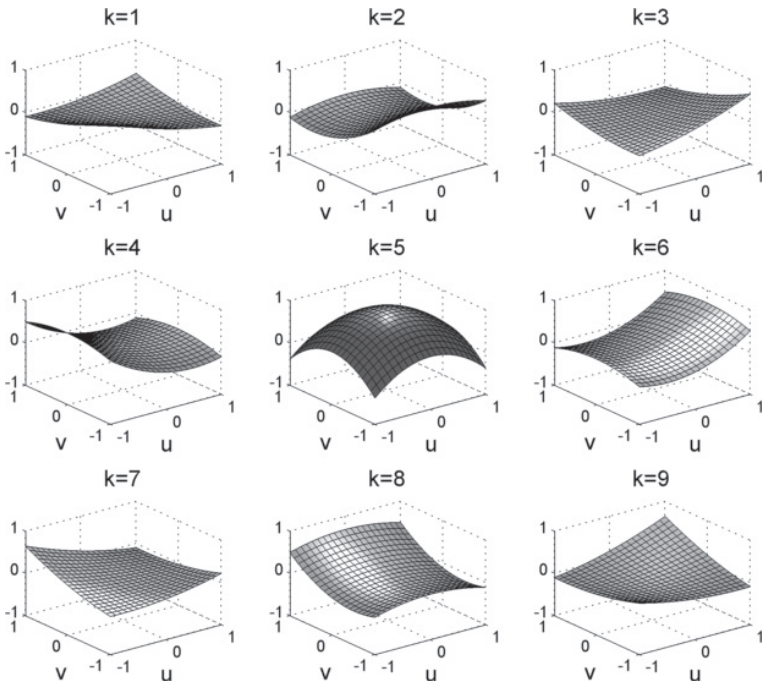


Figure 3 — Second-degree polynomials in the displacement variables

For each candidate $\{x, y, \sigma^*(x,y)\}$, the polynomial q is maximized or minimized by finding the zeros in the first derivatives,

$$\begin{aligned} &2\beta_5(x,y,\sigma^*)u + \beta_3(x,y,\sigma^*)v + \beta_2(x,y,\sigma^*) = 0 \\ &\beta_3(x,y,\sigma^*)u + 2\beta_4(x,y,\sigma^*)v + \beta_1(x,y,\sigma^*) = 0 \\ &\qquad\qquad\qquad\Downarrow \\ &u^*(x,y,\sigma^*) = u, \quad v^*(x,y,\sigma^*) = v \end{aligned} \tag{31}$$

A candidate shall be eliminated if one of the displacement parameters u^* or v^* has absolute value larger than 1. Saddle points almost never occur due to the elimination mechanism of Formula (22) and Formula (23).

The remaining candidates are updated as

$$\{x, y, \sigma^*(x,y)\} \leftarrow \{x + u^*(x,y,\sigma^*), y + v^*(x,y,\sigma^*), \sigma^*(x,y)\} \tag{32}$$

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

5.3.5 Transformation of coordinates and scale to the converted image resolution

Candidates are detected one octave at a time, and the analyzed images in each octave have half the size of those in the preceding octave. At this step in the processing the coordinates and scales are referred to the coordinate system of the octave in which they are detected. A further processing step is therefore necessary in order to map coordinates and scale to the resolution of the converted image.

The octaves are numbered with $q = 0,\ldots, Q - 1$ where 0 refers to the largest image size and $Q - 1$ refers to the smallest image size. For each octave with number $q \geq 1$, the coordinates and scale of all candidates shall be multiplied by a magnification factor

$$c(q) = 2^q \tag{33}$$

For all candidates $\{x, y, \sigma^*(x, y)\}$ detected in octave q, the coordinates shall be multiplied as

$${}_c x \leftarrow c(q) \cdot x \quad {}_c y \leftarrow c(q) \cdot y \tag{34}$$

and the scales shall be multiplied as

$${}_c\sigma^*({}_c x, {}_c y) \leftarrow c(q) \cdot \sigma^*(x, y). \tag{35}$$

Hereinafter, the left subscript c for symbols of coordinates and scale shall indicate that they refer to the resolution of the converted image.

These operations shall be performed for all octaves with number $q \geq 1$.

5.3.6 Elimination of duplicates

After completion of all octaves, a further step of processing shall eliminate duplicates of interest points due to independent processing of each octave. Candidates in any octave with number $q \geq 1$ shall therefore be compared to candidates of the preceding octave with number $q - 1$.

Each candidate $\{x, y, \sigma^*(x, y)\}$ detected in octave q shall be compared to each candidate $\{x', y', \sigma^*(x', y')\}$ detected in octave $q - 1$, and if the candidates are close both in the xy-plane and in the σ-dimension, as given by two thresholds θ_4, set to 2.0, and θ_5, set to 0.4, i.e. if

$$\sqrt{(2x - x')^2 + (2y - y')^2} < \theta_4 \text{ with } \theta_4 = 2.0 \wedge \left|2\sigma^*(x, y) - \sigma^*(x', y')\right| < \theta_5 \text{ with } \theta_5 = 0.4 \tag{36}$$

then both candidates are subject to an elimination process, as follows: if the polynomial values associated to the two candidates have opposite signs, both shall be kept. If the polynomial values are of the same sign, then the candidate having the smallest absolute value of said polynomial shall be eliminated, i.e. if $\left|p\left(x, y, \sigma^*(x, y)\right)\right| > \left|p\left(x', y', \sigma^*(x', y')\right)\right|$ then $\{x', y', \sigma^*(x', y')\}$ shall be eliminated, otherwise $\{x, y, \sigma^*(x, y)\}$ shall be eliminated.

5.3.7 Orientation Assignment

To allow rotation invariance for the subsequent feature description, each interest point shall be assigned a dominant orientation based on the distribution of quantized gradient directions of the pixels in an image circular patch with a radius equal to $3.96 \cdot \sigma^*(x, y)$. In the event that more than one dominant orientations are identified for a single interest point, that interest point shall be replicated in the set of interest points and a different dominant orientation among the identified ones shall be assigned to each replicated interest point.

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Given an interest point detected in a Gaussian-filtered image I_k, $k \in \{1,2\}$ [see Formula (5)], within octave q, with location (x_0,y_0) and detection scale $\sigma_0^*(x_0,y_0)$, the gradient magnitude and direction shall be computed for every pixel in the interest point's neighboring region. The value of k shall be chosen from {1.2} so that I_k is the closest Gaussian-filtered image to the detection scale $\sigma_0^*(x_0,y_0)$. For a pixel at location (x,y), the gradient magnitude $m_{I_k}(x,y)$ and orientation $\theta_{I_k}(x,y)$ shall be computed as

$$m_{I_k}(x,y)=\sqrt{(I_k(x+1,y)-I_k(x-1,y))^2+(I_k(x,y+1)-I_k(x,y-1))^2} \tag{37}$$

$$\theta_{I_k}(x,y)=\arctan\frac{I_k(x,y+1)-I_k(x,y-1)}{I_k(x+1,y)-I_k(x-1,y)} \tag{38}$$

Then, an orientation histogram Hist with N = 36 bins shall be formed from the computed gradient orientations, each bin covering $\frac{2\pi}{N}$ radians. The center of the ith bin Hist_i is $\frac{2\pi}{N}\times i$, $i \in \{0,\ldots,N-1\}$. Each pixel (x, y) within the circular patch shall be added to its nearest two histogram bins Hist_i and $\text{Hist}_{(i+1)\%N}$ based on its orientation $\theta_{I_k}(x,y)$, so that histogram bin values Hist_i and $\text{Hist}_{(i+1)\%N}$ are accumulated by the increment values of

$$\left(1-\frac{\left|\theta_{I_k}(x,y)-\frac{2\pi}{N}\times i+2\pi\right|\%2\pi}{\frac{2\pi}{N}}\right)\times m_{I_k}(x,y)\times e^{\frac{r^2}{2\times\left(1.32\times\sigma_0^*(x_0,y_0)\right)^2}} \tag{39}$$

and

$$\left(1-\frac{\left|\theta_{I_k}(x,y)-\frac{2\pi}{N}\times[(i+1)\%N]+2\pi\right|\%2\pi}{\frac{2\pi}{N}}\right)\times m_{I_k}(x,y)\times e^{\frac{r^2}{2\times\left(1.32\times\sigma_0^*(x_0,y_0)\right)^2}} \tag{40}$$

respectively, where

$$r^2=(x-x_0)^2+(y-y_0)^2 \tag{41}$$

and

$$e^{\frac{r^2}{2\times\left(1.32\times\sigma_0^*(x_0,y_0)\right)^2}} \tag{42}$$

is a Gaussian weighted window with a radius of 1.32 times the detection scale.

The orientation histogram is subsequently smoothed by 6 iterations of mean filtering with filter window size 3.

Dominant orientations shall be determined by locating the peaks in the orientation histogram. The bin corresponding to the highest peak, as well as the bins with a bin value greater than 80% of the highest peak value, are selected as the dominant orientations of the interest point. Once the peak is selected, a quadratic interpolation between the peak and its two neighboring bins is performed to obtain more accurate orientation.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

5.3.8 Interest point characteristics

The above process produces a set of interest points with the characteristics: the location in converted image ${}_cx$, ${}_cy$; the scale in scale-space ${}_c\sigma^*$; the scale-normalized LoG response value p obtained with the polynomial value $p\left(x,y,\sigma^*(x,y)\right)$; the orientation θ; the ratio ρ of the squared trace of the Hessian to the determinant of the Hessian (see Formula 26), and the second derivative $p_{\sigma\sigma}$ of the scale space function with regard to σ.

5.4 Local feature selection

5.4.1 Operation

The interest point detection of 5.3 produces a number of interest points and hence local features (M) which may be greater than the number of local features (N) that is possible to store at a given image descriptor length. Therefore, a subset of N local features shall be selected from the M local features on the basis of a relevance probability measure r that is computed for each of the detected local features.

The relevance measure r is computed with the following five parameters as input (see 5.3.8):

a) the scale ${}_c\sigma^*$ of the interest point;

b) the scale-normalized LoG response value p obtained with the polynomial value $p\left(x,y,\sigma^*(x,y)\right)$;

c) the distance d from the interest point at coordinates (${}_cx$,${}_cy$) to the image center at the converted image resolution;

d) the ratio ρ of the squared trace of the Hessian to the determinant of the Hessian (see Formula 26);

e) the second derivative $p_{\sigma\sigma}$ of the scale space function with regard to σ. $p_{\sigma\sigma}$ is a shorthand for $\frac{\partial^2}{\partial\sigma^2}p(x,y,\sigma^*(x,y))$.

These parameters are quantized within the intervals shown in the tables of normative Annex C, and each quantization interval has an associated scalar value, also shown in the tables of normative Annex C. The five scalar values obtained with this procedure shall be multiplied in order to produce the final local feature relevance measure

$$r({}_c\sigma^*,p,d,\rho,p_{\sigma\sigma})=f_1({}_c\sigma^*)\cdot f_2(p)\cdot f_3(d)\cdot f_4(\rho)\cdot f_5(p_{\sigma\sigma}) \tag{43}$$

where the factors f_1,...f_5 shall be taken from the last column of the tables in normative Annex C. Any encoding step which requires fewer local features (N) than those detected (M) shall use the N local features with the highest values of the relevance measure.

The following example illustrates the working principle for determining the relevance values. Figure 4 depicts the interest points found in an image, while Figure 5 illustrates the local feature relevance measures.

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Figure 4 — Interest points found in an example image

Figure 5 — The interest points of the image of Figure 4 plotted with circles whose diameters are proportional to the relevance measures

5.4.2 Descriptor components

In the binary representation:

RelevanceBitsPresent specifies if a relevance bit for each compressed local feature descriptor is present in the bitstream or not. Its value shall be 1 if relevance bits are present in the bitstream, otherwise it shall be 0. The value of RelevanceBitsPresent shall be set according to the image descriptor length as shown in Table 5.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Table 5 — RelevanceBitsPresent values for the six image descriptor lengths

Image descriptor length	RelevanceBitsPresent
512 bytes	0
1024 bytes	0
2048 bytes	0
4096 bytes	0
8192 bytes	1
16384 bytes	1

RelevanceBits specify which compressed local feature descriptors correspond to the top 300 local features according to the sorting performed using the relevance measure. If the kth local feature is one of the top 300 local features, then RelevaceBits[k] is set to 1, otherwise it is set to 0. If NumberOfLocalDescriptor<300, then all the values in RelevanceBits are set to 1,

The relevance bits are ordered in the same order as the descriptors in LocalDescriptorElement, as described in 5.9.

5.5 Local feature description

The interest point detection of 5.3 produces a number of interest points, each characterized its position (x, y), its scale σ^*, and its orientation θ in I_k, where I_k, $k \in \{1,2\}$ is a Gaussian filtered image in the detection octave and k is chosen from {1,2} so that I_k is the closest Gaussian-filtered image to the detection scale σ^*. For a detected interest point, a local feature descriptor shall be extracted from a local image region around the interest point as described below.

The local region of the interest point shall be centered at its position (x, y), and shall be rotated by the interest point orientation θ so that its x axis is aligned to the orientation θ. The local region shall be divided into 4 horizontal and 4 vertical spatial subdivisions referred to as cells. The size of each side of each cell shall be $m\sigma^*$ pixels, where $m = 2.64$. From each cell, a histogram of gradients with 8 orientation bins, referred to as cell histogram, shall be generated. A local region histogram shall be formed by concatenating these cell histograms. This shall be referred to as a local feature descriptor, with 4x4x8 = 128 bins. Here, we denote the local feature descriptor as $h(t,i,j)$, where t (t = 0,...,7) represents the index of the orientation bin, i (i = 0,...,3) and j (j = 0,...3) represent the index of the horizontal and vertical spatial bins.

The procedure by which the local feature descriptor is constructed is described below, using a canonical local region representation, where the axis of the image and the local region coincide, and the size of each side of the cell is normalized to 1, depicted in Figure 6.

First, the gradient vector field for the scale space image shall be calculated.

$$G(x,y) = \nabla I_k(x,y) = \left[\frac{\partial I_k}{\partial x} \frac{\partial I_k}{\partial y} \right] \tag{44}$$

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Centers of each bin are given by,

$$\theta_t = \frac{\pi}{4} t, t = 0,\ldots,7$$
$$x_i = i - \frac{3}{2},\ i = 0,\ldots,3 \tag{45}$$
$$y_i = j - \frac{3}{2},\ j = 0,\ldots,3$$

The histogram shall be computed by using trilinear interpolation, i.e. by weighing contributions by the binning functions, expressed in the following equation.

$$h(t,i,j) = \int g_{\sigma_{win}}(x,y) \cdot w_{ang}(\angle G(x,y) - \theta_t) \cdot w(x - x_i) \cdot w(y - y_i) \cdot |G(x,y)| dxdy \tag{46}$$

Here $g_{\sigma_{win}}$ is a Gaussian window of standard deviation of σ_{win} = 2.0, and the weight functions are given by,

$$w(z) = \max(0, 1 - |z|)$$
$$w_{ang}(z) = \sum_{k=-\infty}^{\infty} w\left(\frac{4z}{\pi} + 8k\right) \tag{47}$$

Then, the histogram shall be L2 normalized, i.e. each bin of the histogram shall be divided by the L2 norm of the histogram. The resulting histogram shall then be clamped at 0.2, i.e. the bins with values greater than 0.2 shall take the value 0.2. Then, the histogram shall be L2 normalized again. Then, each bin in the histogram shall be mapped to an integer between 0 and 255 as

$$h(t,i,j) = \lfloor \min(255, 512 \cdot h(t,i,j)) \rfloor \tag{48}$$

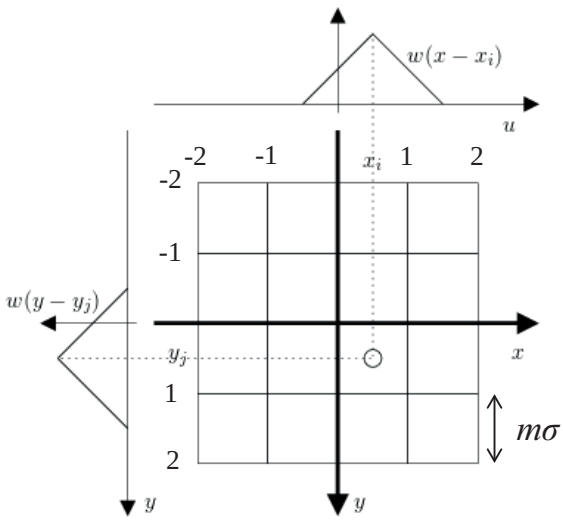


Figure 6 — Local feature descriptor construction

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

5.6 Local feature descriptor aggregation

5.6.1 Operation

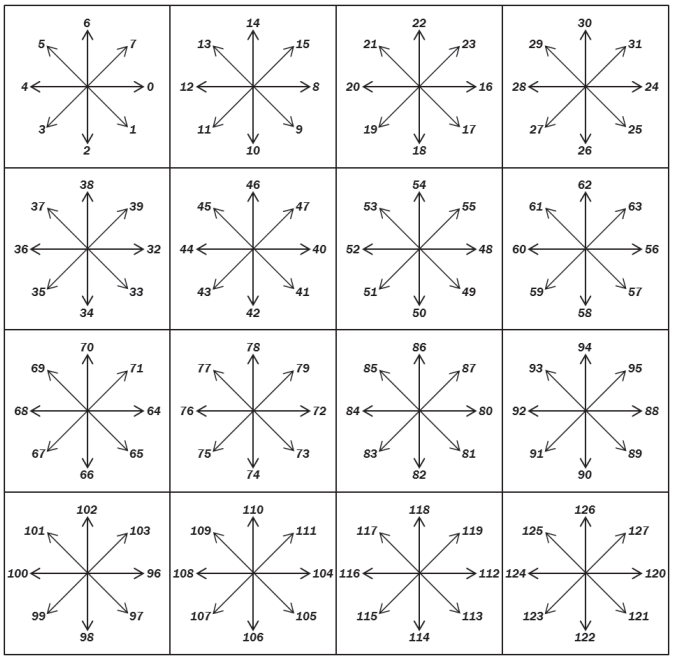


Figure 7 — Local feature descriptor histogram bin ordering in a 128-dimensional vector

The local feature descriptor aggregation process shall be applied to the top K local feature descriptors, selected and computed as described in 5.4 and 5.5, respectively, to produce a global descriptor. Denoting M the total number of available local feature descriptors in an image, the value of K shall be calculated as:

$$K = \begin{cases} M & \text{if } M \le 250 \\ 250 & \text{otherwise} \end{cases} \tag{49}$$

For the local feature descriptor aggregation, each 128-dimensional local feature descriptor h_t, $t = 0, \ldots, K\text{-}1$, with its histogram bins ordered as shown in Figure 7, shall be normalized and then transformed by Principal Component Analysis (PCA) to produce a 32-dimensional vector x_t, $t = 0,\ldots, K\text{-}1$. Specifically, the L1 normalization shall be performed to each dimension of h_t:

$$h'_{t,j} = h_{t,j} / |h_t|, j = 0,\ldots,127 \tag{50}$$

where $|h_t|$ denotes the L1 norm of h_t. This is followed by power normalization:

$$h'_{t,j} \leftarrow \operatorname{sgn}\left(h'_{t,j}\right)\left|h'_{t,j}\right|^{0.5} \tag{51}$$

where $\left|h'_{t,j}\right|$ denotes the absolute value of $h'_{t,j}$. Given the 128x32-dimensional PCA projection matrix P and 128-dimensional mean vector $\tilde{h}$, h'_t shall then be transformed as:

$$x_t = P^T (h'_t - \tilde{h}) \tag{52}$$

The projection matrix P and mean vector $\tilde{h}$ that shall be used are specified in normative Annex D.

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

The aggregation of the vectors x_t relies on Fisher Vector Aggregation. A Gaussian Mixture Model (GMM) comprising 512 Gaussian functions is used, denoted by $\lambda = \{w_i, \mu_i, \sigma_i^2, i = 0,...,511\}$ with w_i, μ_i and σ_i^2 denoting the mixture weight, mean vector, and variance vector of the ith Gaussian function. The 512-dimensional vector w, 512x32-dimensional matrix μ and 512x32-dimensional matrix σ^2 that shall be used are specified in normative Annex E.

For each vector x_t and for each Gaussian function i, the assignment probability $\gamma_t(i)$ shall be calculated as:

$$\gamma_t(i) = p(i \mid x_t, \lambda) = \frac{w_i p_i(x_t \mid \lambda)}{\sum_{j=0}^{511} w_j p_j(x_t \mid \lambda)} \tag{53}$$

where $p_i(x_t \mid \lambda)$ is given by

$$p_i(x_t \mid \lambda) = \frac{\exp\{-\frac{1}{2}(x_t - \mu_i)^T \sigma_i^{-2}(x_t - \mu_i)\}}{(2\pi)^{D/2}\sigma_i}, D = 32 \tag{54}$$

Denoting $X = \{x_t, t = 0,...,K-1\}$ the set of transformed local feature descriptors, the accumulated gradient vector with respect to the mean of the ith Gaussian function shall be calculated as

$$g_{\mu_i}^X = \frac{1}{K\sqrt{w_i}} \sum_{t=0}^{K-1} \gamma_t(i) \frac{x_t - \mu_i}{\sigma_i} \tag{55}$$

and the accumulated gradient vector with respect to the variance of the ith Gaussian function shall be calculated as

$$g_{\sigma_i}^X = \frac{1}{K\sqrt{2w_i}} \sum_{t=0}^{K-1} \gamma_t(i)((\frac{x_t - \mu_i}{\sigma_i})^2 - 1) \tag{56}$$

For each Gaussian function i, the standard deviation $\delta(i)$ of the 32-dimensional accumulated gradient vector $g_{\mu_i}^X = \left[g_{\mu_i,0}^X, g_{\mu_i,1}^X, ..., g_{\mu_i,31}^X\right]$ with respect to the mean of that function shall be calculated as

$$\delta(i) = \sqrt{\frac{1}{32}\sum_{j=0}^{31}\left(g_{\mu_i,j}^X - \frac{1}{32}\sum_{k=0}^{31} g_{\mu_i,k}^X\right)^2} \tag{57}$$

For the local feature descriptor aggregation, a number of Gaussian functions shall be selected to be used in the aggregated local feature descriptor as described below.

First, the Gaussian functions are ranked in descending order according to $\delta(i)$.

For the three lower descriptor lengths of 512 bytes, 1024 bytes and 2048 bytes, the top k Gaussian functions shall be used in the aggregated local feature descriptor. The values that shall be used for the parameter k for different descriptor lengths are specified in normative Annex F.

For the three higher descriptor lengths of 4096 bytes, 8192 bytes and 16384 bytes, the ith Gaussian function shall be selected to be used in the aggregated local feature descriptor if and only if

$$\delta(i) > \tau_\delta \tag{58}$$

where τ_δ denotes the Gaussian function selection threshold. The values that shall be used for the threshold τ_δ for different descriptor lengths are specified in normative Annex F.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

The aggregated gradient vectors $g_{\mu_i}^X$ and $g_{\sigma_i}^X$ for the Gaussian functions selected as described above shall be binarized using sign binarization function $b(z)$ to each dimension of them, which converts any positive value to binary '1' and any non-positive value to binary '0'.

$$b(z) = \begin{cases} 1 & z > 0 \\ 0 & z \le 0 \end{cases} \tag{59}$$

Denoting $\tilde{g}_{\mu_i}^X$ the binarized aggregated gradient vector with respect to mean for the ith Gaussian function and $\tilde{g}_{\sigma_i}^X$ the binarized aggregated gradient vector with respect to variance for the ith Gaussian function, the aggregated local feature descriptor, also referred to as the global descriptor, shall be formed as follows.

For the lowest descriptor length of 512 bytes, the global descriptor $\tilde{g}^X$ shall be generated by concatenating subsets of bits of the mean vectors $\tilde{g}_{\mu_i}^X$ corresponding to the selected Gaussian functions, i.e.

$$\tilde{g}^X = \tilde{g}_{\mu}^X \tag{60}$$

Where $\tilde{g}_{\mu}^X$ denotes the concatenated mean vector calculated as

$$\tilde{g}_{\mu}^X = \{\tilde{g}_{\mu_i}^X \otimes M_i, (i = 0, \ldots, 511) \wedge (i \in A_k^X)\} \tag{61}$$

where A_k^X denotes the top k Gaussian functions ranked in descending order according to $\delta(i)$ and $\tilde{g}_{\mu_i}^X \otimes M_i$ denotes bits selected from $\tilde{g}_{\mu_i}^X$, the binarized aggregated gradient vector with respect to mean for the ith Gaussian function, using 32-bit binary selection mask M_i where a value of 1 indicates that a bit shall be selected and a value of 0 indicates that a bit shall not be selected. The selection mask that shall be used for each Gaussian cluster is specified in normative Annex G.

For the descriptor lengths of 1024 bytes and 2048 bytes, the global descriptor $\tilde{g}^X$ shall be generated by concatenating the mean vectors $\tilde{g}_{\mu_i}^X$ corresponding to the selected Gaussian functions, i.e.

$$\tilde{g}^X = \tilde{g}_{\mu}^X \tag{62}$$

Where $\tilde{g}_{\mu}^X$ denotes the concatenated mean vector calculated as

$$\tilde{g}_{\mu}^X = \{\tilde{g}_{\mu_i}^X, (i = 0, \ldots, 511) \wedge (i \in A_k^X)\} \tag{63}$$

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

For the three higher descriptor lengths of 4096 bytes, 8192 bytes and 16384 bytes, the global descriptor $\tilde{g}^X$ shall be generated by concatenating the mean vectors $\tilde{g}^X_{\mu_i}$ corresponding to the selected Gaussian functions, followed by the variance vectors $\tilde{g}^X_{\sigma_i}$ corresponding to the selected Gaussian functions, i.e.

$$\tilde{g}^X = \{\tilde{g}^X_\mu, \tilde{g}^X_\sigma\} \tag{64}$$

where

$$\tilde{g}^X_\mu = \{\tilde{g}^X_{\mu_i}, (i = 0,...,511) \wedge (\delta(i) \geq \tau_\delta)\} \tag{65}$$

and

$$\tilde{g}^X_\sigma = \left\{\tilde{g}^X_{\sigma_i}, \left(i = 0,...,511\right) \wedge \left(\delta\left(i\right) \geq \tau_\delta\right)\right\} \tag{66}$$

are the concatenated mean and variance vectors, respectively.

5.6.2 Descriptor components

In the binary representation:

GlobalHasBitSelection specifies whether the binary selection masks *M* are applied to the binarized aggregated gradient vectors with respect to mean or not. The value of GlobalHasBitSelection value shall be 1 if bit selection is applied, otherwise it shall be 0. The value of GlobalHasBitSelection shall be set according to the image descriptor length as shown in Table 6.

Table 6 — GlobalHasBitSelection values for the six image descriptor lengths

Image descriptor length	GlobalHasBitSelection
512 bytes	1
1024 bytes	0
2048 bytes	0
4096 bytes	0
8192 bytes	0
16384 bytes	0

GlobalHasVariance specifies whether the image descriptor contains binarized aggregated gradient vectors with respect to variance or not. The value of GlobalHasVariance shall be 1 if the vectors with respect to variance are used, otherwise it shall be 0. The value of GlobalHasVariance shall be set according to the image descriptor length as shown in Table 7.

Table 7 — GlobalHasVariance values for the six image descriptor lengths

Image descriptor length	GlobalHasVariance
512 bytes	0
1024 bytes	0
2048 bytes	0
4096 bytes	1
8192 bytes	1
16384 bytes	1

NumberOfGlobalFunctions corresponds to the total number of Gaussian functions of the global descriptor and NumberOfGlobalFunctions=512.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

GlobalFunctionPresent represents which Gaussian functions are selected and, for the ith Gaussian function,

$$\text{GlobalFunctionPresent} = \begin{cases} 1 & \text{if Gaussian function selected} \\ 0 & \text{otherwise} \end{cases}$$

GlobalFunctionMeanVector represents the concatenated mean vector $\tilde{g}_{\mu}^{X}$.

GlobalFunctionVarianceVector represents the concatenated variance vector $\tilde{g}_{\sigma}^{X}$.

5.7 Local feature descriptor compression

5.7.1 Operation

H_0	H_1	H_2	H_3
H_4	H_5	H_6	H_7
H_8	H_9	H_{10}	H_{11}
H_{12}	H_{13}	H_{14}	H_{15}

Figure 8 — Local feature descriptor comprising 16 cell histograms

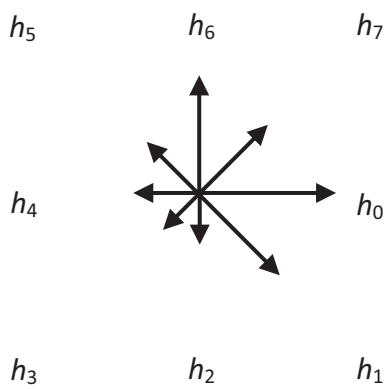


Figure 9 — Cell histogram comprising eight bins

For a local feature descriptor, each of the cell histograms H_0, ..., H_{15} as shown in Figure 8, each with bins h_0, ..., h_7 as shown in Figure 9, shall be independently transformed. More specifically, as shown in Figure 10, each of cell histograms H_0, H_2, H_5, H_7, H_8, H_{10}, H_{13}, and H_{15} shall be transformed using Transform A (see Formula (67)), comprising eight transform functions, and each of cell histograms H_1, H_3, H_4, H_6, H_9, H_{11}, H_{12}, and H_{14} shall be transformed using Transform B (see Formula (68)), also comprising eight transform functions which are different from those of Transform A.

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

The transformation of each of the cell histograms $H_0, \ldots, H_{15}$ using all the transform functions of Transform A or Transform B according to Figure 10 produces 128 transformed descriptor elements. However, depending on the specified image descriptor length, a transformed local feature descriptor shall contain all or a subset of those 128 transformed descriptor elements, as explained in more detail next.

Transform A

$$\begin{aligned}
v_0 &= (h_2 - h_6)/2 \\
v_1 &= (h_3 - h_7)/2 \\
v_2 &= (h_0 - h_1)/2 \\
v_3 &= (h_2 - h_3)/2 \\
v_4 &= (h_4 - h_5)/2 \\
v_5 &= (h_6 - h_7)/2 \\
v_6 &= ((h_0 + h_4) - (h_2 + h_6))/4 \\
v_7 &= ((h_0 + h_2 + h_4 + h_6) - (h_1 + h_3 + h_5 + h_7))/8
\end{aligned} \tag{67}$$

Transform B

$$\begin{aligned}
v_0 &= (h_0 - h_4)/2 \\
v_1 &= (h_1 - h_5)/2 \\
v_2 &= (h_7 - h_0)/2 \\
v_3 &= (h_1 - h_2)/2 \\
v_4 &= (h_3 - h_4)/2 \\
v_5 &= (h_5 - h_6)/2 \\
v_6 &= ((h_1 + h_5) - (h_3 + h_7))/4 \\
v_7 &= ((h_0 + h_1 + h_2 + h_3) - (h_4 + h_5 + h_6 + h_7))/8
\end{aligned} \tag{68}$$

where the symbol ' / ' denotes 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).

A H_0	B H_1	A H_2	B H_3
B H_4	A H_5	B H_6	A H_7
A H_8	B H_9	A H_{10}	B H_{11}
B H_{12}	A H_{13}	B H_{14}	A H_{15}

Figure 10 — Cell histogram transform utilization

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Transformed descriptor elements shall be added to a transformed local feature descriptor in groups of four elements according to the priority list of Table 8, where the element index column specifies one of elements v_0–v_7 and the group index column specifies one of four groups g_0–g_3 of four cell histograms each. Each group g_0–g_3 is an ordered set of four cell histograms, and the four groups shall be defined as

$$g_0=\{H_0, H_{15}, H_3, H_{12}\},\ g_1=\{H_7, H_8, H_1, H_{14}\},\ g_2=\{H_2, H_{13}, H_4, H_{11}\} \text{ and } g_3=\{H_5, H_{10}, H_6, H_9\} \tag{69}$$

For example, denoting ${}^{i}v_j$ a transformed descriptor element with $i\in\{0,...,15\}$ the cell histogram index and $j\in\{0,...,7\}$ the element index, the first row of the priority list of Table 8 specifies that the first four transformed descriptor elements to be added to the transformed local feature descriptor shall be the elements v_0 from the four cells of group g_3, i.e. ${}^{5}v_0$, then ${}^{10}v_0$, then ${}^{6}v_0$, then ${}^{9}v_0$. The second row of the priority list of Table 8 specifies that the next four transformed descriptor elements to be added to the transformed local feature descriptor shall be ${}^{7}v_0$, ${}^{8}v_0$, ${}^{1}v_0$, ${}^{14}v_0$, in that order, and so on.

As shown in Table 8, a transformed local feature descriptor shall comprise 5, 5, 10, 16, 20 and 32 groups of elements for the image descriptor lengths of 512, 1024, 2048, 4096, 8192 and 16384 bytes, respectively.

Since each group has four elements, this means that a transformed local feature descriptor shall comprise 20, 20, 40, 64, 80 and 128 descriptor elements for the image descriptor lengths of 512, 1024, 2048, 4096, 8192 and 16384 bytes, respectively.

Following the transformation step, transformed descriptor elements ${}^{i}v_j$ shall be quantized to three values according to the following formula.

$${}^{i}\tilde{v}_j = \begin{cases} -1 & \text{if } {}^{i}v_j \le {}^{i}QL_j \\ 0 & \text{if } {}^{i}v_j > {}^{i}QL_j \text{ and } {}^{i}v_j \le {}^{i}QH_j \\ +1 & \text{if } {}^{i}v_j > {}^{i}QH_j \end{cases} \tag{70}$$

The quantization thresholds ${}^{i}QL_j$ and ${}^{i}QH_j$ that shall be used are specified in normative Annex H. Following the quantization step, the quantized descriptor elements ${}^{i}\tilde{v}_j$ shall be encoded in the bitstream as specified in Table 9, giving a compressed local feature descriptor $\tilde{V}$ of L elements where L is equal to 20, 20, 40, 64, 80 and 128 for the image descriptor lengths of 512, 1024, 2048, 4096, 8192 and 16384 bytes, respectively.

Table 8 — Priority list for encoding of transformed descriptor elements at different target image descriptor lengths

Priority	Group g_x index	Element v_i index	Number of Groups / Number of elements / Image descriptor length
1	3	0	5 groups / 20 elements / 512 and 1024 bytes
2	1	0	
3	2	0	
4	0	0	
5	3	6	
6	3	1	10 groups / 40 elements / 2048 bytes
7	1	1	
8	2	1	
9	0	1	
10	3	2	
11	1	2	
12	2	2	

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Table 8 *(continued)*

Priority	Group g_x index	Element v_i index	Number of Groups / Number of elements / Image descriptor length
13	0	2	
14	1	6	16 groups /
15	2	6	64 elements /
16	0	6	4096 bytes
17	3	7	
18	1	7	20 groups /
19	2	7	80 elements /
20	0	7	8192 bytes
21	3	3	
22	1	3	
23	2	3	
24	0	3	
25	3	4	
26	1	4	
27	2	4	
28	0	4	
29	3	5	
30	1	5	32 groups /
31	2	5	128 elements /
32	0	5	16384 bytes

Table 9 — Compressed local feature descriptor ternary value encoding

Ternary value	Bitstream encoding
–1	10
0	0
+1	11

5.7.2 Descriptor components

In the binary representation:

NumberOfElementGroups specifies the number of element groups in a compressed local feature descriptor. As shown in Table 10, the number of element groups is dependent on the target image descriptor length.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Table 10 — Values of NumberOfElementGroups for the specified image descriptor lengths

Image descriptor length	NumberOfElement-Groups
512 bytes	5
1024 bytes	5
2048 bytes	10
4096 bytes	16
8192 bytes	20
16384 bytes	32

Each element group contains four elements and the number of elements in each compressed local feature descriptor is given by 4×NumberOfElementGroups.

LocalDescriptorElements represents the compressed local feature descriptors for an image, i.e. **LocalDescriptorElements[k][n]** corresponds to the nth element of the kth compressed local feature descriptor in the image, where each compressed local feature descriptor has 4×NumberOfElementGroups elements.

5.8 Local feature location compression

5.8.1 Operation

5.8.1.1 Overview

The interest point coordinates $({}_cx, {}_cy)$ for the compressed local feature descriptors to be added to the image descriptor shall be encoded at the resolution of the converted image $J(x, y)$ as specified in 5.2 and shall proceed as follows. The converted image shall be subdivided into a matrix LB of non-overlapping blocks of size 3x3 denoted LB(i,j), where i represents the block index in the horizontal x direction and j the block index in the vertical y direction. Out of this representation, a histogram map binary matrix HM with fixed size, equal to the LB matrix size, and a histogram count vector HC with variable size shall be generated as shown in Figure 11 and described below.

For the histogram map matrix HM, each histogram map element HM(i,j) shall take a value of 0 when no interest points are present in the corresponding LB(i,j) and shall take a value of 1 otherwise.

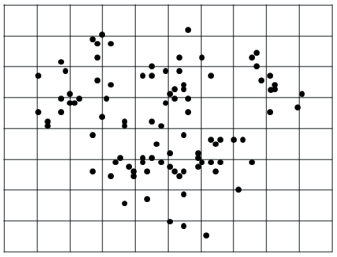
If the vertical resolution of the histogram map matrix HM is greater than its horizontal resolution, then the histogram map matrix HM shall be transposed and the transposed matrix shall be used as the histogram map matrix in all subsequent local feature location compression operations. The following description refers to a histogram map matrix HM with a horizontal resolution greater or equal to its vertical resolution.

For the histogram count vector HC, each histogram count element HC(g), where g denotes the gth non-zero element of HM(i,j) in the scanning order described below, shall take a value equal to the number of interest points present in the corresponding block LB(i,j). Therefore, the total size of the histogram count vector shall be equal to the number of non-zero elements of HM(i,j).

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

a) Example set of identified interest points



b) Subdivision of the scaled resolution image into a block based representation (LB matrix)

0	0	1	0	0	1	0	0	0	0
0	1	1	1	0	1	1	1	0	0
0	1	1	1	1	1	1	1	1	1
0	1	1	1	1	1	0	0	1	0
0	0	1	1	1	1	1	1	0	0
0	0	1	1	1	1	1	1	0	0
0	0	0	1	1	1	0	0	0	0
0	0	0	0	0	1	1	0	0	0

c) Resulting histogram map (non-empty elements marked as 1, empty elements marked as 0)

Figure 11 — Histogram map generation process

Both the histogram count and the histogram map shall be encoded using static arithmetic coding.

The histogram count shall be encoded using a 64-symbol, single model, static arithmetic coding scheme. The arithmetic coder symbols are the histogram count elements HC(g), processed in the order g=0,1,2,..., and the model probabilities are shown in normative Annex I. The operation of the arithmetic encoder is described in 5.8.1.3 below.

The histogram map shall be encoded using a binary context-based static arithmetic coding scheme. The input to the arithmetic coder is each histogram map element HM(i,j). The histogram map elements shall be encoded following a specific order defined by a clockwise circular scanning of the histogram map, beginning with elements located at the center of the image towards elements located at a periphery of the image, as illustrated in Figure 12a. The scanning procedure shall stop when the last non-zero element of the histogram map has been encoded.

Defining Hx and Hy as the horizontal resolution and vertical resolution of the histogram map HM, respectively, and pTot as the number of elements of the histogram count vector HC, the circular scanning shall be carried out according to the following procedure, described in pseudo-code form.

```
isOdd = Hy % 2;
iTotalSteps = (Hy - isOdd) / 2;
```


© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

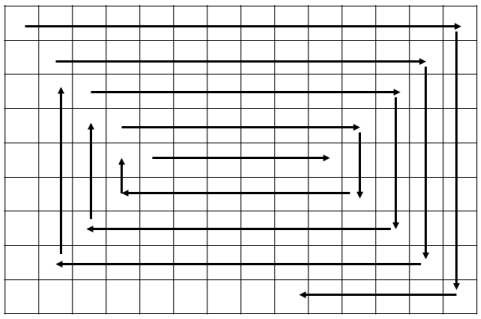
```
iMinX = iTotalSteps - 1;
iMaxX = Hx - 1 - (iTotalSteps - 1);
iMinY = iTotalSteps - 1;
iMaxY = Hy - 1 - (iTotalSteps - 1);
iEncodedCount = 0;

if (isOdd)
{
     iY = iMinY + 1;
     for (iX = iMinX + 1; iX < iMaxX; iX++)
     {
          Encode(HM[iX,iY]);
          if (HM[iX,iY] != 0 )
               if (++iEncodedCount >= pTot)
                    terminate encoding;
     }
}
for(iSteps = 0; iSteps < iTotalSteps; iSteps++)
{
     // Left side of the rectangle
     For (iX = iMinX, iY = iMinY; iY <= iMaxY; iY++)
     {
          Encode (HM[iX,iY]);
          if (HM[iX,iY] != 0)
               if (++iEncodedCount >= pTot)
                    terminate encoding;
     }
     // Bottom side of the rectangle
     For (iX = iMinX + 1, iY = iMaxY; iX < iMaxX; iX++)
     {
          Encode (HM[iX,iY]);
          if (HM[iX,iY] != 0)
               if (++iEncodedCount >= pTot)
                    terminate encoding;
     }
     // Right side of the rectangle
     For (iX = iMaxX, iY = iMaxY; iY >= iMinY; iY--)
     {
          Encode (HM[iX,iY]);
          if (HM[iX,iY] != 0)
               if (++iEncodedCount >= pTot)
                    terminate encoding;
     }
     // Upper side of the rectangle
     For (iX = iMaxX - 1, iY = iMinY; iX > iMinX; iX--)
     {
          Encode (HM[iX,iY]);
          if (HM[iX,iY] != 0)
               if (++iEncodedCount >= pTot)
                    terminate encoding;
     }
     iMinX--; iMinY--; iMaxX++; iMaxY++;
}
```

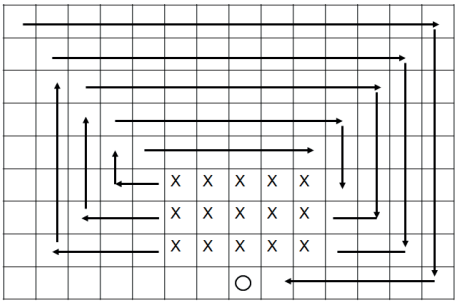
In the pseudo-code shown above, the procedure Encode refers to arithmetic encoding of input elements. The operation of the arithmetic encoder is described in 5.8.1.3. The context adopted at each encoding step shall be determined according to 5.8.1.2.

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015



a) Circular Scanning



b) Circular scan with the reference region used for generating the context. The blocks marked with 'x' represent an example reference region for the to-be-coded block (marked with 'o')

Figure 12 — Context generation for arithmetic coding

5.8.1.2 Histogram map coding

5.8.1.2.1 First stage of the coding of the histogram map

The first histogram map elements processed in the defined scanning order of the histogram map shall be encoded using a binary single model static arithmetic coding scheme. The model probabilities $p(q)$ with $q \in \{0,1\}$ shall be computed as described in normative Annex J. This first stage of the arithmetic coding of the histogram map shall terminate when a number of loops of the circular scanning equal to 5 has been scanned.

5.8.1.2.2 Second stage of the coding of the histogram map

Following the first stage, the subsequent histogram map elements processed in the defined scanning order of the histogram map shall be encoded using a binary context-based static arithmetic coding scheme.

The context information for encoding the HM(i,j) element shall be computed as follows: a reference region in spatial proximity of HM(i,j) shall be selected, and a context value k shall be computed.

The reference region shall have a maximum size of 55 elements in an 11x5 rectangle. For element HM(i,j). the vertices of the reference region shall be positioned as follows: HM(i-5, j-5; i+5, j-5; i+5, j-1; i-5, j-1) when j>(vertical scaled resolution/2), or HM(i-5, j+5; i+5, j+5; i+5, j+1; i-5, j+1) otherwise. This is illustrated in Figure 12b. During the circular scanning and according to the position of the element to be encoded, it is possible that some of the elements of the reference region are not yet encoded or that the vertex positions calculated as shown above fall outside the boundaries of the histogram map matrix. In those cases, only those elements of the reference region already visited by the scanning

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

process and falling within the boundaries of the histogram map shall be considered for the context value computation.

The context value k shall be computed as

$$k = \left\lfloor \frac{n \cdot 55}{s} \right\rfloor \tag{71}$$

where n is the number of non-zero elements falling into the reference region, $s \leq 55$ is the total number of encoded elements in the reference region, and 55 is the maximum size of a reference region.

The model probabilities $p(q|k)$ with $q \in \{0,1\}$ and $k \in \{0,\ldots,55\}$ and shall be computed as described in normative Annex J.

5.8.1.3 Arithmetic encoding process

5.8.1.3.1 Arithmetic encoding overview

This subclause describes the normative arithmetic encoding process that matches the arithmetic decoding process described in the informative Annex K.

The histogram count and the histogram map shall be encoded using the static arithmetic coding process described in this subclause. In both cases, the encoding process shall follow exactly the same steps; only the number of symbols and their probabilities differ, as described in normative Annex I and normative Annex J. The elements of histogram count and histogram map shall be encoded in the sequential order described earlier in this subclause.

Inputs to the arithmetic encoding process are the elements of the vector to be encoded, hereby referred as *val*, the probabilities of the different symbols to occur *ctx* and the number of values to be encoded *nVal*. Outputs of this process are sequential bits of the encoded bitstream.

In the arithmetic coding process each element of *val* vector is sequentially encoded, according to the following pseudo-code. The Encode procedure is described in deeper details later in this subclause.

```
for (i = 0; i < nVal; i++)
     Encode (val[i], ctx);
```

At each encoding stage, the operational interval needs to be updated. Two variables are used hereby to exhaustively describe such interval, namely *codILow*, pointing to the lower end of the interval, and *codIHigh*, specifying the upper end of the interval. At the beginning of the process, *codILow* is set equal to 0, and *codIHigh*, set equal to the maximum value of the interval. This initial interval is subdivided in four identical quarters, with thresholds *firstQtr*, *half*, and *thirdQtr,* respectively. The variable *fbits*, storing the number of following bits, is set to 0.

5.8.1.3.2 Arithmetic encoding process of one input value

This subclause describes the behavior of the Encode procedure of the previous pseudo-code, for the arithmetic encoding of one input value.

Inputs to this procedure are the element to be encoded, hereby referred as *val* and the probabilities vector *ctx*. Outputs of this procedure are the updated variables *codIHigh* and *codILow*, and optionally new encoded bits.

The following pseudo-code describes the procedure behavior:

```
nVal = sizeof (ctx);
val = nVal - val -1;
RangeAdaptation (&codLow, &codRange, ctx[], val);
for (;;) {
   if (codIHigh <half) {
      OutputLeadingBitPlusFollow (0);
   } else if (codILow >=half) {
      OutputLeadingBitPlusFollow (1);
```

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

```
      codILow -= half;
      codIHigh -= half;
   } else if (codILow >= firstQtr && codIHigh < thirdQtr) {
      fbits += 1;
      codILow -= firstQtr;
      codIHigh -= firstQtr;
   } else
      break;
   codILow = 2* codILow;
   codIHigh = 2* codIHigh +1;
}
```

RangeAdaptation and OutputLeadingBitPlusFollow procedures are described in the following paragraphs.

The encoder outputs the leading bits as soon as this can be known with certainty. This happens in two cases: when the upper bound of the interval is smaller than *half* (output 0), or when the lower bound is bigger than *half* (output 1). If *codILow* falls into the second quarter of the interval, and *codIHigh* falls into the third quarter of the interval, therefore when the current interval straddles 0.5, it is not possible to predict the output bit; however, it is known that the following bit will have opposite value. Therefore, the occurrence of this case is tracked, increasing the value of the variable *fbits:* the bits tracked by *fbits* are hereby referred as following bits. The following bits shall be assigned values opposite to the next leading bit being flushed to the encoded bitstream. In order to maintain fixed precision, a re-normalization of the interval length shall be applied when one leading bit is outputted or a following bit is tracked. The following pseudo-code describes the RangeAdaptation procedure where, at each encoder step, the current operational interval is updated. The whole interval is divided into sub-intervals proportional to the probability of occurrence of the different symbols. Such probabilities, are described as non-normalized integer cumulative probabilities in normative Annex I and normative Annex J.

Input of the procedure are *codILow*, *codIRange*, the context *ctx* and the value *val* to be encoded. Output of the procedure are the updated values *codILow* and *codIRange,* set to the boundaries of the sub-interval corresponding to the actual symbol to be encoded. Such sub-interval becomes the interval used in the next step.

```
RangeAdaptation (&codILow, &codIHigh, ctx [], val)
{
   codIRange = codIHigh – codILow + 1;
   codIHigh = codILow + (codIRange *ctx[val])/ctx[0]-1;
   codILow = codILow + (codIRange *ctx[val+1])/ctx[0];
}
```

The following pseudo-code describes the procedure of writing new bits on the bitstream, when necessary. If there is any bit tracked by *fbits* variable, such bits shall be assigned value opposite to the current leading bit.

```
OutputLeadingBitPlusFollow (int bit)
{
   output_bit (bit);
   while (fbits > 0) {
      output_bit (!bit);
      fbits -= 1;
}
```

5.8.2 Descriptor components

In the binary representation:

HistogramCountSize

Size of the histogram count vector.

HistogramMapSizeX

Specifies the horizontal X resolution of the histogram map for location coding.

HistogramMapSizeY

Specifies the vertical Y resolution of the histogram map for location coding.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

HistogramCount

Binary representation of the histogram count.

HistogramMap

Binary representation of the histogram map.

5.9 Encoding order of compressed local feature descriptors and relevance bits

Compressed local feature descriptors, corresponding to the local features with the highest values of the relevance measure, shall be encoded in the following order. Starting from the histogram map matrix representation, as described in 5.8, the histogram map matrix shall be scanned column-by-column, scanning each column top-to-bottom, and the compressed local feature descriptors in each non-null block of the histogram map matrix encountered during the scanning process shall be encoded. When a non-null block of the histogram map matrix contains more than one descriptor, the descriptors within the non-null block shall be ordered in descending order of their corresponding local feature relevance measure, i.e. starting from the descriptor with the highest relevance measure.

When used, the relevance bits shall be encoded in the order as the compressed local feature descriptors.

5.10 Computation of the number of compressed local feature descriptors at different image descriptor lengths

This subclause is informative, and does not form an integral part of this International Standard.

This process outlined below may be used to determine the number of compressed local feature descriptors that can be added in an image descriptor, given a specified image descriptor length. It is applied after the computation of the global descriptor.

a) The number of bits **TargetLength** is computed multiplying the target image descriptor length by 8;

b) The number **AvailableBits** is computed subtracting the size in bits of the global descriptor from **TargetLength**;

c) The value **AverageLocalLength** is computed summing the number of bits **BitsPerCoordinate** used on average to encode the interest point coordinate and 1.666×4×**NumberOfElementGroups**. **BitsPerCoordinate** is equal to 7.5, 7.0, 5.0, 4.8, 4.7, and 4.6 for the image descriptor lengths of 512, 1024, 2048, 4096, 8192 and 16384 bytes, respectively;

d) The value **ApproxNumber,** an estimation of the number of compressed local feature descriptors that can be added to the image descriptor, is computed truncating to integer the value **TargetLength** / **AverageLocalLength** + 0.5**;**

e) The number **UsedBits** is determined by compressing **ApproxNumber** local feature descriptors and corresponding interest point coordinates;

f) The final number of compressed local feature descriptors, **NumberOfLocalDescriptors**, is determined truncating to integer the value (**AvailableBits** × **ApproxNumber**) / **UsedBits** + 0.25.

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Annex A
(informative)

CDVS encoder organization

This annex is informative, and does not form an integral part of this International Standard.

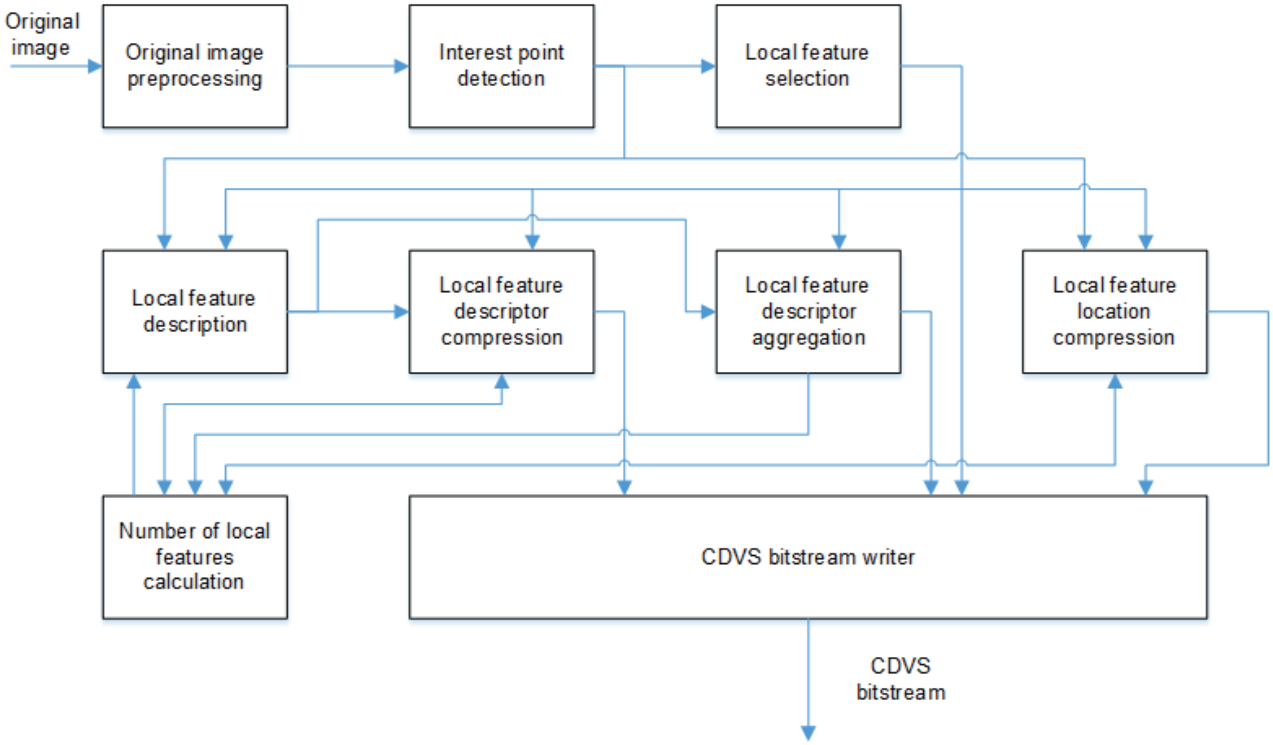


Figure A.1 — CDVS encoder organisation

Figure A.1 shows a simplified diagram of a complete CDVS encoder implementing the encoding operations of Clause 5. More specifically, the "Original image preprocessing" block performs the operations specified in 5.2. Then, the "Interest point detection" block identifies interest points in the converted image as specified in 5.3. The characteristics of the interest points are then used in the "Local feature selection" block, which performs the operations specified in 5.4 to rank the local features according to their relevance. For a number of the most relevant features, the "Local feature description" and "Local feature descriptor compression" blocks compute uncompressed local descriptors as specified in 5.5 and compressed local descriptors as specified in 5.7, respectively. The coordinates of those features are encoded by the "Local feature location compression" block, as specified in 5.8, while the "Local feature descriptor aggregation" block computes a global descriptor as specified in 5.6. The number of local feature descriptors to be computed/compressed, and corresponding coordinates to be encoded, is determined by the "Number of local features calculation" block, according to 5.10. As can be seen in 5.10, this operates by first computing a coarse estimate according to prior statistics, and then a refined estimate based on a trial CDVS encoding. The "CDVS bitstream writer" block then encodes CDVS bitstreams comprising the global descriptor, compressed local feature descriptors, encoded coordinates, and local feature relevance information, according to the binary representation syntax specified in 4.1 and based on the ordering specified in 5.9.

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Annex B
(normative)

Coefficients for coordinate refinement

Table B.1 — Coefficients for polynomial approximation to scale-space function in the displacement parameters, $\sigma = \sigma_0$

	a_k	b_k	c_k	d_k	e_k	f_k
k=1	0.098214	0.098214	0.278622	-0.113412	-0.113412	-0.027805
k=2	0.310104	-0.19549	0	-0.31171	0	0.06846
k=3	0.098214	0.098214	-0.27862	-0.11341	0.113412	-0.02781
k=4	-0.19549	0.310104	0	0	-0.31171	0.06846
k=5	-0.6204	-0.6204	0	0	0	0.84439
k=6	-0.19549	0.310104	0	0	0.311714	0.06846
k=7	0.098214	0.098214	-0.27862	0.113412	-0.11341	-0.02781
k=8	0.310104	-0.19549	0	0.311714	0	0.06846
k=9	0.098214	0.098214	0.278622	0.113412	0.113412	-0.02781

Table B.2 — Coefficients for polynomial approximation to scale-space function in the displacement parameters, $\sigma = \sigma_1$

	a_k	b_k	c_k	d_k	e_k	f_k
k=1	0.095647	0.095647	0.264138	-0.107186	-0.107186	-0.028542
k=2	0.311924	-0.19079	0	-0.30458	0	0.063217
k=3	0.095647	0.095647	-0.26414	-0.10719	0.107186	-0.02854
k=4	-0.19079	0.311924	0	0	-0.30458	0.063217
k=5	-0.62438	-0.62438	0	0	0	0.863039
k=6	-0.19079	0.311924	0	0	0.304575	0.063217
k=7	0.095647	0.095647	-0.26414	0.107186	-0.10719	-0.02854
k=8	0.311924	-0.19079	0	0.304575	0	0.063217
k=9	0.095647	0.095647	0.264138	0.107186	0.107186	-0.02854

Table B.3 — Coefficients for polynomial approximation to scale-space function in the displacement parameters, $\sigma = \sigma_2$

	a_k	b_k	c_k	d_k	e_k	f_k
k=1	0.094511	0.094511	0.257040	-0.104562	-0.104562	-0.028356
k=2	0.312611	-0.18873	0	-0.30036	0	0.059829
k=3	0.094511	0.094511	-0.25704	-0.10456	0.104562	-0.02836
k=4	-0.18873	0.312611	0	0	-0.30036	0.059829
k=5	-0.62567	-0.62567	0	0	0	0.874571
k=6	-0.18873	0.312611	0	0	0.300359	0.059829
k=7	0.094511	0.094511	-0.25704	0.104562	-0.10456	-0.02836

AVA Documents
Document Sharing Hub

© ISO/IEC 2015 – All rights reserved

Full document details: ISO IEC 15938-13-2015

Table B.3 *(continued)*

	a_k	b_k	c_k	d_k	e_k	f_k
k=8	0.312611	-0.18873	0	0.300359	0	0.059829
k=9	0.094511	0.094511	0.25704	0.104562	0.104562	-0.02836

Table B.4 — Coefficients for polynomial approximation to scale-space function in the displacement parameters, $\sigma = \sigma_3$

	a_k	b_k	c_k	d_k	e_k	f_k
k=1	0.093707	0.093707	0.253510	-0.102943	-0.102943	-0.029147
k=2	0.313394	-0.187276	0	-0.298825	0	0.059807
k=3	0.093707	0.093707	-0.25351	-0.102943	0.102943	-0.029147
k=4	-0.187276	0.313394	0	0	-0.298825	0.059807
k=5	-0.62703	-0.62703	0	0	0	0.877473
k=6	-0.187276	0.313394	0	0	0.298825	0.059807
k=7	0.093707	0.093707	-0.25351	0.102943	-0.102943	-0.029147
k=8	0.313394	-0.187276	0	0.298825	0	0.059807
k=9	0.093707	0.093707	0.25351	0.102943	0.102943	-0.029147

© ISO/IEC 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015

Annex C
(normative)

Probability values for the feature selection

Table C.1 — Probability of the scale

From scale	To scale	Probability
0	2.3556	0.12954
2.3556	3.1955	0.20912
3.1955	4.4075	0.25351
4.4075	6.2719	0.30088
6.2719	9.1889	0.30965
9.1889	14.591	0.32856
14.591	28.0947	0.28868
28.0947	Inf	0.23013

Table C.2 — Probability of the scale normalized LoG response value

From response value	To response value	Probability
0	2.1369	0.048659
2.1369	4.1829	0.10754
4.1829	6.6621	0.15476
6.6621	9.5024	0.18724
9.5024	12.6649	0.21148
12.6649	16.1452	0.23032
16.1452	19.9594	0.24637
19.9594	24.1346	0.26105
24.1346	28.7362	0.27355
28.7362	33.8848	0.28806
33.8848	39.7636	0.3026
39.7636	46.7056	0.31695
46.7056	55.3583	0.33558
55.3583	67.3264	0.35803
67.3264	87.2124	0.38619
87.2124	Inf	0.39905

© ISO/IEC 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO IEC 15938-13-2015