INTERNATIONAL STANDARD

ISO/IEC 15938-14

First edition
2018-02

Information technology — Multimedia content description interface —

Part 14: Reference software, conformance and usage guidelines for compact descriptors for visual search

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

Partie 14: Logiciels de référence, conformité et lignes directrices pour l'utilisation des descripteurs compacts pour recherche visuelle

Reference number
ISO/IEC 15938-14:2018(E)

Contents

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

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

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

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

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

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

A list of all parts in the ISO/IEC 15938 series can be found on the ISO website.

Introduction

ISO/IEC 15938 (all parts) 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 ISO/IEC 15938 (all parts).

— **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.

ISO/IEC 15938 consists of 14 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 ISO/IEC 15938.

— **Part 7 – Conformance testing**: specifies the guidelines and procedures for testing conformance of implementations of ISO/IEC 15938.

— **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.

— **Part 14 – Reference software, conformance and usage guidelines for compact descriptors for visual search**: provides the reference software, specifies the conformance testing, and gives usage guidelines for compact descriptors for visual search.

The compact descriptors for visual search (CDVS) tool specified in ISO/IEC 15938-13 is 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.

ISO/IEC 15938-14:

— specifies the reference software for CDVS (Clause 5);

— specifies the conformance testing dataset, reference descriptors and conditions for CDVS (Clause 6);

— provides guidelines for the usage of CDVS (Clause 7).

The CDVS reference software is provided at http://standards.iso.org/iso-iec/15938/-14/ed-1/en.

Information technology — Multimedia content description interface —

Part 14: Reference software, conformance and usage guidelines for compact descriptors for visual search

1 Scope

This document provides the reference software, specifies the conformance testing, and gives usage guidelines for ISO/IEC 15938-13.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

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

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

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

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

3.1
image descriptor
descriptor extracted from one image

3.2
image descriptor length
size of an image descriptor in bytes

Note 1 to entry: ISO/IEC 15938-13 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.

3.3
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

3.4
local region
area in an image in the neighbourhood of an interest point, used to generate local feature descriptors

3.5
local feature descriptor
descriptor of a local region

3.6
global descriptor
aggregation of local feature descriptors into a compact representation of the image

3.7
compressed local feature descriptor
compressed representation of a local feature descriptor

4 Symbols and abbreviated terms

4.1 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.

4.2 Abbreviations

CDVS Compact Descriptors for Visual Search

MPEG Moving Picture Experts Group

MPEG-7 ISO/IEC 15938

4.3 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

4.4 Logical operators

|| logical OR

∨ logical OR

&& logical AND

∧ logical AND

! logical NOT

4.5 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
≠	not equal to

4.6 Bitwise operators

|	OR
&	AND

4.7 Assignment

=	assignment operator
←	assignment operator
→	assignment operator, except in $Q \rightarrow R$ and $R \rightarrow Q$, where it denotes matching direction between image descriptors Q and R

4.8 Set operators

∪	set union
∩	set intersection
⊆	subset
#	cardinality

4.9 Constants

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

4.10 Functions

ln() base-e logarithm

max() maximum value in argument list

min() minimum value in argument list

cos() cosine function

$| \, |$ 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

5 Reference software

5.1 Reference software location

The CDVS reference software is provided at http://standards.iso.org/iso-iec/15938/-14/ed-1/en.

5.2 Reference software license

Licensing information is provided in "CDVS_evaluation_framework/COPYING" and "CDVS_applications/COPYING". Each source code file also contains its own detailed licensing information in the header.

5.3 Reference software documentation

The CDVS reference software is documented by means of Doxygen. The documentation is provided in "CDVS_evaluation_framework/docs" of the CDVS reference software.

5.4 Reference software compilation

Instructions on how to compile the CDVS reference software is provided in "CDVS_evaluation_framework/docs/CDVS-build-run-instructions.pdf" of the CDVS reference software.

The CDVS reference software is entirely written in C and C++. On Windows, it has been compiled and tested on Windows 7 Enterprise 64-bit using Visual C++ 2010 (64-bit). On Linux, it has been compiled and tested on Ubuntu 14.04 LTS (64-bit).

5.5 Reference software architecture

The CDVS reference software comprises a library and several 64-bit applications using the library, which are functionally split into two groups:

a) CDVS evaluation framework, as used during the development of ISO/IEC 15938-13:

   1) "extract": extracts CDVS descriptor from a set of images;

   2) "match": examines pairs of CDVS descriptors and determines if the corresponding images are matching or not;

   3) "mkIndex": concatenates CDVS descriptors in a file structure that may be used as a reference database for the "retrieve" application;

4) “joinIndices”: concatenates reference databases previously produced by “mkIndex” to produce a larger reference database for the “retrieve” application;

5) “retrieve”: compares a query CDVS descriptor against a reference database and identifies the most similar entries in the database.

b) CDVS example applications:

1) “simpleextract”: extracts a CDVS descriptor from a single image;

2) “simplematch”: examines two CDVS descriptors and determines if the corresponding images are matching or not;

3) “simpleretrieve”: compares a set of query images with a set of reference images and identifies the most similar entries;

4) “conformance”: an application implementing the CDVS conformance testing procedure of Clause 6;

5) “HTTPServer”: a web service able to perform retrieval on a database of images, and to update its own database (this code has an additional dependency on libmicrohttpd);

6) “batchextract”: parallel extraction from an input directory of images to an output directory of CDVS descriptors;

7) “example1”, “example2”, “example3”: very simple examples using the CDVS library.

Information on how to build, install and run the CDVS reference software is provided in “CDVS_evaluation_framework/docs/CDVS-build-run-instructions.pdf” of the CDVS reference software.

6 Conformance testing

6.1 Conformance image data set and reference CDVS bitstreams

The conformance image dataset comprises a set of 1000 images. These images are named cvds000.jpg, cdvs001.jpg, ..., cdvs999.jpg and are provided in “CDVSConformanceDaset.zip” available at http://standards.iso.org/iso-iec/15938/-14/ed-1/en.

Reference CDVS bitstreams for the conformance image dataset are also provided. For each image cdvsxxx.jpg in the conformance image dataset, six reference CDVS bitstreams are provided corresponding to the six image descriptor lengths specified in the ISO/IEC 15938-13, i.e. 512 bytes, 1024 bytes, 2048 bytes, 4096 bytes, 8192 bytes and 16384 bytes. The reference CDVS bitstreams are provided in “CDVSReferenceBitstreams.zip”.

6.2 Conformance test conditions

6.2.1 General

To verify conformance of a CDVS encoder, test CDVS bitstreams shall be extracted from all the images in the conformance image dataset and for all six image descriptor lengths, and the test CDVS bitstreams shall be compared to the corresponding reference CDVS bitstreams.

A CDVS encoder shall be deemed conformant if all 6000 of the test CDVS bitstreams are decoded correctly by the CDVS reference software conformance application and no less than 5940 of the test CDVS bitstreams pass all of the conformance tests specified in 6.2.2 to 6.2.4.

6.2.2 Global descriptor test

For a pair or corresponding test and reference CDVS bitstreams, the bitwise (Hamming) distance between each 32-bit word of the test CDVS bitstream global descriptor and the corresponding 32-bit word of the reference CDVS bitstream global descriptor shall be computed. The number of distances not exceeding four bits shall be counted. The test CDVS bitstream shall pass the global descriptor test if the number of distances not exceeding four bits is greater than or equal to a threshold θ_G, where $\theta_G = 461$ for the image descriptor lengths of 512 bytes, 1024 bytes and 2048 bytes, and $\theta_G = 922$ for the image descriptor lengths of 4096 bytes, 8192 bytes and 16384 bytes.

6.2.3 Interest point test

For a pair or corresponding test and reference CDVS bitstreams, the number of interest points in the test CDVS bitstream shall be compared to the number of interest points in the reference CDVS bitstream. The test CDVS bitstream shall pass the interest point test if the number n_T of interest points in the test CDVS bitstream fulfils the condition $0,9 < \frac{n_R}{n_T} < 1,1$, where n_R is the number of interest points in the reference CDVS bitstream.

6.2.4 Local descriptor test

For a pair or corresponding test and reference CDVS bitstreams, the local descriptor test shall be conducted as follows.

First, p_1 shall be computed as $p_1 = \min(n_R, n_T)$ where n_T and n_R are the number of interest points in the test and reference CDVS bitstreams, respectively.

Then, a one-to-one correspondence between the interest points of the test and reference CDVS bitstream local descriptors shall be established. For each interest point in the reference CDVS bitstream, a one-to-one correspondence with an interest point in the test CDVS bitstream shall be established if there is at least one interest point in the test CDVS bitstream such that:

a) the L1 distance between the test and reference CDVS bitstream local descriptors is less than or equal to a threshold θ_L^e, where θ_L^e is equal to 8, 8, 16, 26, 32 and 51 for the image descriptor lengths of 512 bytes, 1024 bytes, 2048 bytes, 4096 bytes, 8192 bytes and 16384 bytes, respectively; and

b) the L2 distance between the (x, y) coordinates of the test and reference interest points is less than a threshold θ_L^c, where $\theta_L^c = 6,5$ pixels.

Let p_2 denote the number of interest points in the reference CDVS bitstream for which a correspondence is established. The test CDVS bitstream shall pass the local descriptor test if $\frac{p_2}{p_1} > 90\ \%$.

7 Usage guidelines

7.1 Overview

This Clause is informative, and does not form an integral part of this document.

This Clause describes how compact descriptors for visual search can be used in pairwise matching, i.e. comparison of two image descriptors to determine the similarity between the images, and retrieval, i.e. projecting a query descriptor to a database of descriptors so as to retrieve the most similar database images to the query image.

7.2 Pairwise matching

7.2.1 Overview

Given two image descriptors Q and R, where Q denotes a query and R a reference descriptor, the pairwise matching procedure described here performs a comparison between the global descriptors of Q and R, as well as the matching of the local descriptors in Q and R. Figure 1 illustrates this pairwise matching procedure. This procedure determines whether the query and the reference images depict the same objects or scene (a match), or not (a non-match). In case of a match, it also produces localization information, i.e. the position of the matching objects in the image.

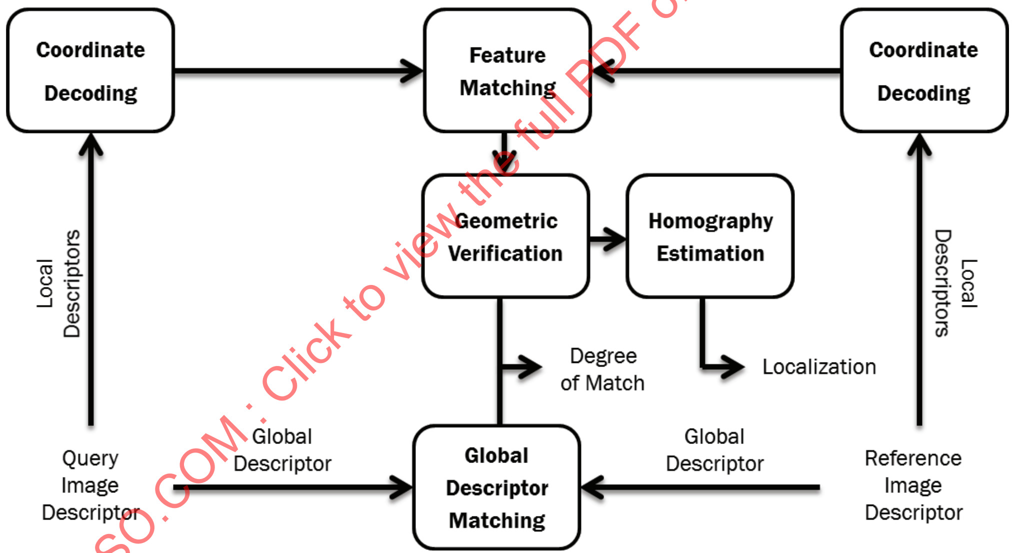

Figure 1 — Pairwise matching pipeline

7.2.2 Global descriptor matching

For the matching of the global descriptors, the similarity score is a weighted correlation given by

$$s_{Q,R} = \frac{\sum_{i=0}^{511} b_i^Q b_i^R W_{HA\left(u_i^Q, u_i^R\right)} \left(D - 2HA\left(u_i^Q, u_i^R\right)\right)}{\left(\sum_{i=0}^{511} b_i^Q\right)^{0.3} \left(\sum_{i=0}^{511} b_i^R\right)^{0.3}}$$

where

u_i^Q	denotes the binarized Fisher sub-vector $g_{\mu_i}^Q$ or $g_{\sigma_i}^Q$ of the i^{th} Gaussian function in GMM. $b_i^Q = 1$ if the i^{th} Gaussian function is present in the descriptor for Q, and $b_i^Q = 0$ otherwise;
u_i^R	denotes the binarized Fisher sub-vector $g_{\mu_i}^R$ or $g_{\sigma_i}^R$ of the i^{th} Gaussian function in GMM. $b_i^R = 1$ if the i^{th} Gaussian function is present in the descriptor for R, and $b_i^R = 0$ otherwise;
$HA\left(u_i^Q, u_i^R\right)$	represents the Hamming distance of the Fisher sub-vector of the i^{th} Gaussian function from Q and R, ranging from 0 to D (D=24 for the image descriptor length of 512 bytes and D=32 for the image descriptor lengths of 1024 bytes, 2048 bytes, 4096, bytes, 8192 bytes and 16384 bytes);
$W_{HA\left(u_i^Q, u_i^R\right)}$	denotes the penalty to the Hamming distance and the weights are learned from a set of matching/non-matching image pairs.

A single threshold τ is used to declare a match/non-match. If $s_{Q,R} > \tau$, a match according to the global descriptors is declared, otherwise a non-match.

7.2.3 Local descriptors matching

7.2.3.1 Two-way local feature matching

A two-way local descriptor matching process is used, matching descriptors from the query Q to the reference R and from the reference R to the query Q.

For local descriptor matching, the compressed local feature descriptors and their coordinates are decoded for both Q and R. Then, for each of the two matching directions, $Q \rightarrow R$ and $R \rightarrow Q$, the matching proceeds as follows.

The compressed local feature descriptors are compared in the compressed domain using the L1 distance, which is calculated for corresponding groups of four ternary elements. When compressed local feature descriptors of different lengths are matched, the scalability of the descriptors is exploited by reducing the higher length descriptors to the subset of elements which appear in the lower length descriptors.

The ratio r of closest distance to the next closest distance is used as a criterion for distinctiveness to determine the interest point matches (correspondences) between Q and R. Each matching interest point pair is then assigned a matching score calculated as

$$\beta = \cos\left(\frac{\pi r}{2}\right)$$

Whenever two or more different interest points in one image match the same interest point in the other image, the match that has the largest score is maintained. This improves the chance of having correct matches among those submitted to the geometric consistency check stage.

The above process is performed in each matching direction, $Q \rightarrow R$ and $R \rightarrow Q$. This process generally produces partially overlapping sets of matching interest point pairs. Denote X the set of interest point pairs found in the $Q \rightarrow R$ matching direction, and Y the set of interest point pairs found in the $R \rightarrow Q$ matching direction. Similarly, denote $\beta_{Q \rightarrow R}$ a matching score in the $Q \rightarrow R$ matching direction, and $\beta_{R \rightarrow Q}$ a matching score in the $R \rightarrow Q$ matching direction. For a matching interest point pair, a match weight w is then calculated as follows:

— For matching interest point pairs which are present in only one of the two directions, i.e. in $\left(X \cup Y - X \cap Y\right)$, w is calculated as

$$w = \gamma \cdot \beta \text{ with } \gamma = \min\left(1, \frac{\#\left(X \cap Y\right)}{\#\left(X \cup Y - X \cap Y\right)}\right),$$ β the matching score $\beta_{Q \rightarrow R}$ or $\beta_{R \rightarrow Q}$, and # denoting the cardinality of the set.

— For matching interest point pairs which are present in both directions, i.e. in $X \cap Y$, w is calculated as

$$w = \frac{\beta_{Q \rightarrow R} + \beta_{R \rightarrow Q}}{2}$$

7.2.3.2 Geometric verification

The geometric verification estimates the correctly matched interest points for the two images, provided that a preliminary test declares a probable match between the images.

To this end, the histogram of logarithmic distance ratios (LDR) for pairs of matches is used.

Let the interest point pairs in $X \cup Y$, i.e. all the interest point pairs detected in either one or both matching directions as described in 7.2.3.1, be

$$(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$$

where x_n denotes the coordinates of an interest point in Q and y_n denotes the coordinates of the matched interest point in R.

The LDR for these matches are

$$z_{ij} = \ln \left\| \frac{x_i - x_j}{y_i - y_j} \right\|,$$

forming the set

$$Z = \{ z_{ij} \mid i \neq j \}$$

The LDR for pairs of inliers is distributed in a manner that is distinctively different to how the LDR for pairs of outliers and (inlier/outlier) pairs are distributed. The distribution for all pairs is represented by a histogram h for these values, by counting the occurrences over each bin,

$$h(k) = \#(Z \cap \zeta_k)$$

The bins $\zeta_1, ..., \zeta_k$ are adjacent intervals, each of width 0.2, subdividing the interval [−2.5, 2.5].

The set of correctly matched interest points in one image is assumed to undergo squeezing and stretching within certain limits with respect to the set of correctly matched interest points in the other image, such that the inliers contribute to bins contained in a limited portion of the interval [−2.5, 2.5].

The outlier behaviour is expressed through a discrete probability density

$$f(k), k = 1, ..., K$$

This function is called the outlier pdf. The uses a nonparametric version of $f(k)$, obtained by a convolution of two histograms. These two histograms represent the frequencies of the two terms of the LDR

$$\ln \| x_i - x_j \|, \quad -\ln \| y_i - y_j \|$$

The method proceeds in two stages: one is a hypothesis test used to exclude rapidly most non-matching pairs of images, the second estimates the inliers, used for ranking, decision and localization purposes.

In the first stage, the hypothesis test compares the histogram h to the outlier pdf f through Pearson's goodness-of-fit test. This test has the probability of false positives as a parameter. This probability is set a little higher than the target, leaving the final decision to the second step.

In the second stage, the inliers are estimated by solving the eigenvalue problem outlined below.

a) Compute the factor β

$$\beta = \frac{\Sigma_{k=1}^{K} h(k) f(k)}{\Sigma_{k=1}^{K} (f(k))^2}$$

b) Create the outlier normal of the histogram

$$d(k) = h(k) - \beta f(k)$$

c) Let q be the quantizer that assigns a bin to any LDR value

$$z \in \zeta_k \Rightarrow z \xrightarrow{q} k$$

Make the inlier evidence matrix D

$$D_{ij} = \begin{cases} d_q(z_{ij}) & i \neq j \\ 0 & i = j \end{cases}$$

where $d_q = d \circ q$, and z_{ij} are the LDR values

d) Find the dominant eigenvector r or D with eigenvalue μ

$$Dr = \mu r$$

e) Estimate the number of inliers

$$\hat{m} = 1 + \frac{\mu}{\max_{k=1,\ldots,K} d(k)}$$

f) The inliers correspond to (the indices of) the $\hat{m}$ largest elements in the eigenvector r.

The expected inlier evidence matrix (i.e. D modelled as a random matrix) has the exact inlier indicating vector as its dominant eigenvector. The elements of this vector assume only two values: zero in correspondence to the outliers and a non-zero constant in correspondence to the inliers.

For illustrative purposes, Figure 2 shows two images depicting the same building, photographed from different angles and distances. Figure 2 also indicates the interest points and matches. As can be seen, the matches are predominantly incorrect, due to both the repeated patterns that are found on building facades and the occlusion by trees. Figure 3 shows the inliers identified by the above process. Figure 4 shows the elements of the dominant eigenvector, used to identify the inliers.

7.2.3.3 Weighted inliers computation

A match between Q and R is quantified by accumulating the weights w of the local feature matches, computed in 7.2.3.1, that are found to be inliers by the geometric verification of 7.2.3.2. If the sum of these weights exceeds a given threshold, i.e.

$$\left(\sum_{n:\text{match } n \text{ is correct}} w_n \right) > \theta$$

then an image match according to the local descriptors is declared, otherwise a non-match.

7.2.4 Image match decision

The pairwise matching declares that Q and R depict the same objects or scene if a match is declared according to the global descriptor or the local descriptors. In the case of a match, localization information is calculated as described 7.2.5.