INTERNATIONAL STANDARD

ISO/IEC 23000-10

First edition
2009-05-01

AMENDMENT 1
2010-06-01

Information technology — Multimedia application format (MPEG-A) —

Part 10: Video surveillance application format

AMENDMENT 1: Conformance and reference software

Technologie de l'information — Format pour application multimédia (MPEG-A) —

Partie 10: Format pour application à la vidéosurveillance

AMENDEMENT 1: Conformité et logiciel de référence

Reference number
ISO/IEC 23000-10:2009/Amd.1:2010(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

COPYRIGHT PROTECTED DOCUMENT

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Foreword

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

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Amendment 1 to ISO/IEC 23000-10:2009 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

This Amendment contains details of the conformance specifications and reference software for the Video surveillance application format (VSAF).

Information technology — Multimedia application format (MPEG-A) —

Part 10: **Video surveillance application format**

AMENDMENT 1: Conformance and reference software

Page 15, after 6.4.1.3 add the following two new clauses:

7 Conformance Points

7.1 Introduction

This Clause provides bitstreams and pointers to the software implementation that can be used to provide a deterministic answer to the following questions:

— Does a given bitstream conform to Video surveillance application format (VSAF) as defined in this part of ISO/IEC 23000, i.e. is it able to be decoded by the informative Video surveillance viewer (VSV) implementation?

— Does a given Video surveillance viewer (VSV) implementation conform to Video surveillance application format (VSAF) as defined in this part of ISO/IEC 23000, i.e. is it able to decode the reference bitstreams contained in this part of ISO/IEC 23000?

— The Video surveillance packer (VSP) software is normative and produces conformance VSAF bitstreams.

The term bitstream is used here to describe the VSAF file format together with its contents.

7.2 AVC Bitstream

MPEG-4 AVC (Advanced Video Coding) bitstreams stored inside a VSAF file shall comply with the specifications of this part of ISO/IEC 23000. An AVC bitstream that conforms to the VSV shall meet the following requirements:

— All AVC bitstreams shall have the same total duration, e.g. all video clips shall be the same length of time. Their physical lengths may vary, e.g. according to fps (frames per second) etc.

7.3 File Format

— VSAF files shall comply with the specifications of this part of ISO/IEC 23000. As an extension of the AVC File Format [ISO/IEC 14496-15:2004], it shall contain AVC file format structure with derivations as specified in 4.5 of this part of ISO/IEC 23000 as well as metadata boxes as specified in 6.2, 6.3, and 6.4 of this part of ISO/IEC 23000.

- The 'major brand' name shall contain 'vsf1' brand name, so the VSV can parse the boxes in the file format that conforms to the file format as specified in 4.5 of this part of ISO/IEC 23000.

- The AVC video track shall conform to the definition as described in Clause 5 of this part of ISO/IEC 23000.

7.4 Metadata

- All the XML metadata shall conform to the sets of ISO/IEC 15938 Parts 1 (Systems), 2 (DDL), 3 (Visual) and 5 (MDS) specified for VSAF. Every element shall comply with the specifications of this part of ISO/IEC 23000.

- The semantics of XML metadata shall conform to the specifications of this part of ISO/IEC 23000.

7.5 Measurement procedure

The procedure to check the conformance points is as follows:

- The VSV parses the VSAF file and displays its contents. The video tracks it contains are playable, the binary metadata and MPEG-7 metadata are processed and displayed.

- The VSV parses the SPS (sequence parameter sets) NAL (network abstraction layer) unit of the AVC bitstream inside the AVC video track.

There is no tolerance in measurement procedure. The diagnosis is pass or fail, however, it is possible to identify which parts of a VSV do and which do not comply.

7.6 Conformance files

A number of conformance files have been produced using the Video surveillance packer (VSP) software. The files are listed in Table AMD1.1.

The VSAF specification supports AVC bitstreams up to level 3.1; therefore different conformance files have been produced reflecting a sample of the appropriate range. The files are designed to test VSAF functionality and not the complete range of compliant AVC levels.

The conformance files are based upon the video bitstream "foreman.264". An external software module was used to generate the ISO/IEC 15938 meta-data included.

Table AMD1.1 — Use the VSV implementation.

File	Conformance test	Pass/fail (o/ x)
1.mp4	single camera (track), testing: AVC bitstream profile and level conformance correct mandatory binary metadata major brand the video tracks are the same length	
2.mp4	two cameras, testing: as above the ability for the format to contain more than one camera (i.e. video track) the file's MAFIdentification box Predecessor field references 1.mp4 the file's MAFIdentification box Successor field references 3.mp4	
3.mp4	two cameras, testing: as above ISO/IEC 15938 file- and track-level metadata elements are present	

8 Reference Software

8.1 Introduction

This Clause describes a cross-platform software implementation in the C++ programming language of Video surveillance application format (VSAF) as part of this part of ISO/IEC 23000.

- Video surveillance packer (VSP) – normative
- Video surveillance viewer (VSV) – informative

The VSAF reference software is normative in the sense that it correctly implements the normative clauses contained in this part of ISO/IEC 23000. Conforming this part of ISO/IEC 23000 implementations are not expected to follow the algorithms or the programming techniques used by the VSAF reference software. Although the packing software is considered normative, it is not expected to add anything normative to the VSAF textual clauses included in this part of ISO/IEC 23000. At present, the software compiles under Windows and Linux.

If present, ISO/IEC 15938 descriptors containing DominantColor are processed to provide a demonstration of content-based image retrieval.

8.2 Architecture

The architecture of the software as depicted in Figure AMD1.1 has three main components comprising a packer, with which to create conformance VSAF files, a viewer with which to inspect the conformance VSAF files and the various shared software libraries.

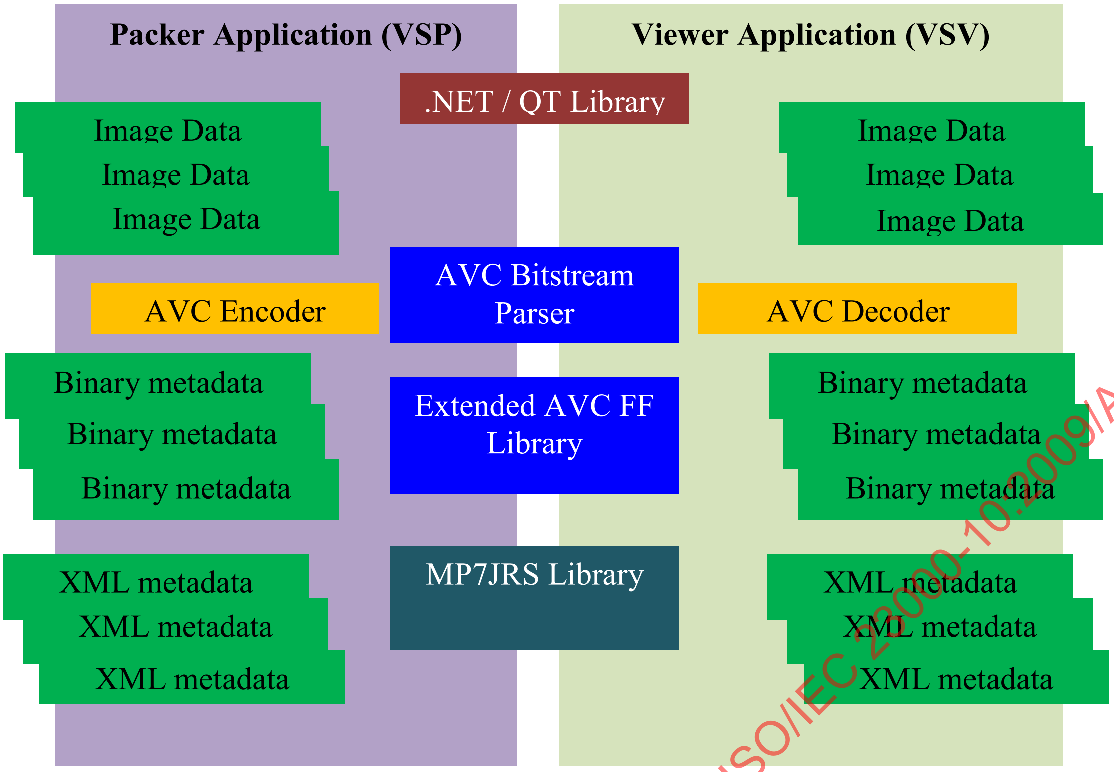


Figure AMD1.1 — Architecture of the VSAF reference software

8.3 Software components

Accordingly the VSAF reference software is organised according the following structure:

— **Video surveillance packager (VSP)**: a user-interface and set of classes for the generation of the AVC (Advanced Video Coding) File Format with Video surveillance application format extensions (VSE) otherwise known as a VSAF file.

— **Video surveillance viewer (VSV)**: a user-interface and set of classes to parse and display the AVC file format with VSE. The application allows the playback of the video tracks.

— **Extended AVC FF (File Format)**: a library of classes implementing the primitive functions defined in the Technical Specification. This software is normative as much as the Technical Specification and provides the functionalities needed to pack and unpack the VSAF files.

— **MP7JRS library**: non-normative library of classes encapsulating the functionalities to generate and parse MPEG-7 XML data which a device may have as a hardware or software implementation.

— **AVC Encoder**: non-normative library of classes encapsulating the functionalities to encode an AVC bitstream which a device may have as a hardware or software implementation.

— **AVC Decoder**: non-normative library of classes encapsulating the functionalities to decode an AVC bitstream which a device may have as a hardware or software implementation.

— **AVC Bitstream Parser**: non-normative library of classes encapsulating the functionalities to pack and unpack AVC bitstream to and from the AVC file format with VSE which a device may have as a hardware or software implementation.

— **Data**: The data input and output by the VSP and VSV applications. These data include folders of images, binary metadata and XML metadata.

8.3.1 Internal software tools

The VSAF reference software comprises the following software tools:

- **VSAF file format reference software**: ISO Base Media File Format amended with AVC extensions (see Table AMD1.2 for modifications to the library).
- **AVC bitstream parser**: NAL (network abstraction layer) conversion and file format insertion / extraction.
- **YUV to RGB conversion**: C++ class converts the output of the AVC decoder into red green blue (RGB) images.
- **VSAF MPEG-7 metadata library**: based upon MP7-JRS (see External software tools).
- **Video surveillance packer (VSP)**: command line program to create a new VSAF file based upon a configuration file.
- **Video surveillance viewer (VSV)**: opens the VSAF file, tests for conformance and plays back tracks.
- **DominantColor Description Scheme decoder**: restricted to aspects of the description scheme.

8.3.2 External software tools

Each individual module also requires a number of libraries in order to be compiled; these must be downloaded.

- The MP7JRS 2.2 library. The MP7-JRS performs the task of MPEG-7 Parser/ Marshaller/ Unmarshaller.
- The AVC reference decoding and encoding software: (JM v.15.1).
- Boost library for conversion from string to unsigned 64 bit number.

8.4 Video surveillance packer (VSP)

The Video surveillance packer (VSP) software forms the normative part of the reference software. It comprises a command line user-interface with which to produce conformance VSAF files. The software requires an initialisation file to provide the necessary VSAF configuration details. All the features of the specification are able to be produced.

NOTE The timed meta-data tracks are artificially produced.

8.4.1 Header files

- #include "CBitStream.h"
- #include "CPacket.h"
- #include "CAnalyse.h"
- #include "CommonDef.h"
- #include "MP4Atoms.h"
- #include "MP4Movies.h"
- #include "MP4Descriptors.h"

- #include "AVCSampleFunctions.h"
- #include "vsaf_meta_data.h"
- #include "boost/lexical_cast.hpp"

8.4.2 Functions

The following functions are used to generate the appropriate Boxes and functions to populate them:

- ISOSetMovieBrand()
- ISONewFileMeta()
- ISONewMAFIdentification()
- MP4NewHandle()
- ISOAddPrimaryData()
- MP4DisposeHandle()
- MP4NewMovieTrack()
- MP4AddTrackToMovieIOD()
- MP4CreateAVCConfigAtom()
- MP4SetHandleSize()
- MP4NewTrackMedia()
- MP4BeginMediaEdits()
- MP4SetMediaLanguage()
- ISONewAVCSampleDescription()
- ChunkInit()
- SampleInit()
- ChunkAddSample()
- MP4HandleCat()
- ChunkWrite()
- SampleAddNALUnit()
- MP4EndMediaEdits()
- MP4GetMediaDuration()
- MP4InsertMediaIntoTrack()
- ISONewTrackMeta()

- ISONewCameraIdentification()
- ISOAddTrackReference()
- ISONewVideoSurveillanceMetadataSampleDescription()
- ISOAddMediaSamples()
- ISODisposeHandle()

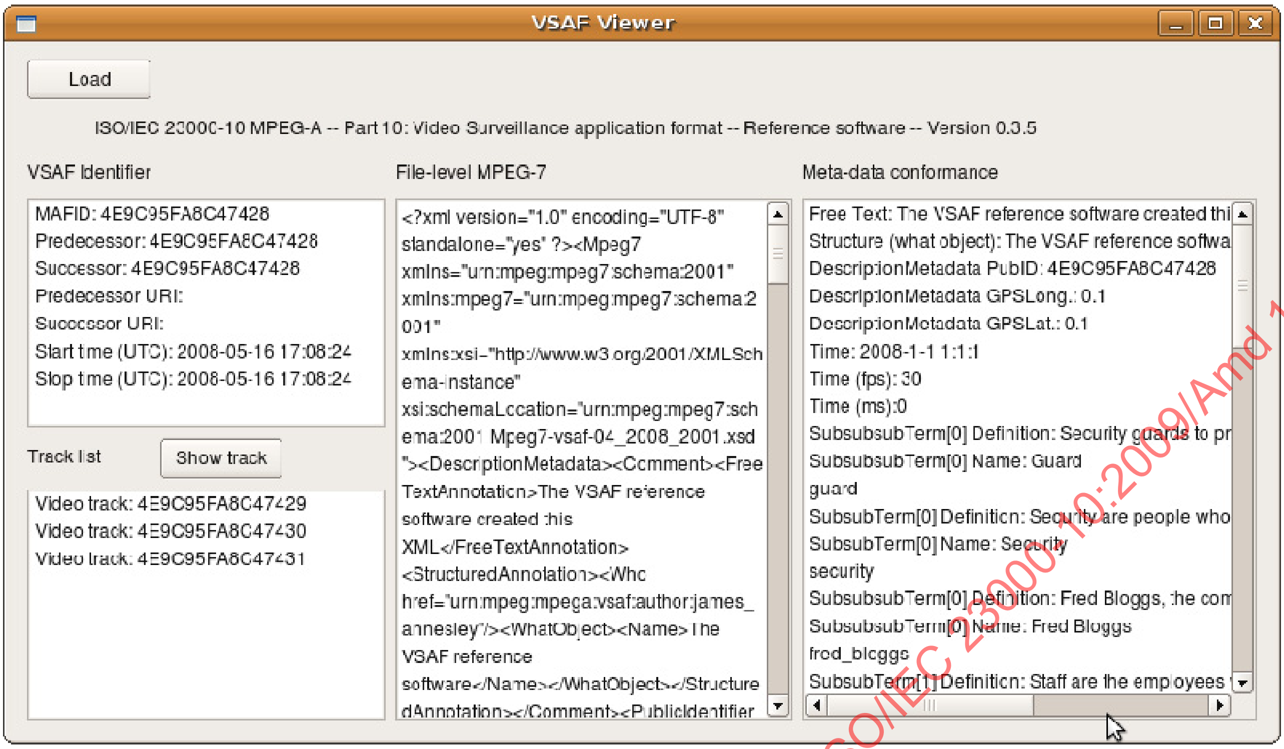
8.5 Video surveillance viewer (VSV)

The Video surveillance viewer (VSV) software forms the informative parts of the reference software. The VSAF files are able to be loaded and inspected and their conformance can be verified. Screenshots of the Main application window are shown in Figure AMD1.2 and a view of the track playback window is shown in Figure AMD1.3.

The VSV parses the metadata using sets of “get” functions of the VSAF file using the VSAF API (Application Programming Interface). These API calls conform to the metadata schema as defined in Annex B of this part of ISO/IEC 23000.

8.5.1 Header files

- #include "vsv_qt.h"
- #include "trackview.h"
- #include "ui_vsv_qt.h"
- #include "ui_trackview.h"
- #include "vsvunpack.h"
- #include "mp4movies.h"
- #include "mp4osmacros.h"
- #include "yuv2rgb.h"
- #include "vsaf_meta_data.h"
- #include "MP4Descriptors.h"
- #include "AVCSampleFunctions.h"



ISO/IEC 23000-10 MPEG-A -- Part 10: Video Surveillance application format -- Reference software -- Version 0.3.5

VSAF Identifier

MAFID: 4E9C95FA8C47428
Predecessor: 4E9C95FA8C47428
Successor: 4E9C95FA8C47428
Predecessor URI:
Successor URI:
Start time (UTC): 2008-05-16 17:08:24
Stop time (UTC): 2008-05-16 17:08:24

Track list

Video track: 4E9C95FA8C47429
Video track: 4E9C95FA8C47430
Video track: 4E9C95FA8C47431

File-level MPEG-7

```
<?xml version="1.0" encoding="UTF-8"
standalone="yes" ?><Mpeg7
xmlns="urn:mpeg:mpeg7:schema:2001"
xmlns:mpeg7="urn:mpeg:mpeg7:schema:2
001"
xmlns:xsi="http://www.w3.org/2001/XMLSch
ema-instance"
xsi:schemaLocation="urn:mpeg:mpeg7:sch
ema:2001 Mpeg7-vsaf-04_2008_2001.xsd
"><DescriptionMetadata><Comment><Free
TextAnnotation>The VSAF reference
software created this
XML</FreeTextAnnotation>
<StructuredAnnotation><Who
href="urn:mpeg:mpega:vsaf:author:james_
annesley"/><WhatObject><Name>The
VSAF reference
software</Name></WhatObject></Structure
dAnnotation></Comment><PublicIdentifier
```

Meta-data conformance

Free Text: The VSAF reference software created thi
Structure (what object): The VSAF reference softwa
DescriptionMetadata PubID: 4E9C95FA8C47428
DescriptionMetadata GPSLong.: 0.1
DescriptionMetadata GPSLat.: 0.1
Time: 2008-1-1 1:1:1
Time (fps): 30
Time (ms):0
SubsubsubTerm[0] Definition: Security guards to pr
SubsubsubTerm[0] Name: Guard
guard
SubsubTerm[0] Definition: Security are people who
SubsubTerm[0] Name: Security
security
SubsubsubTerm[0] Definition: Fred Bloggs, the com
SubsubsubTerm[0] Name: Fred Bloggs
fred_bloggs
SubsubTerm[1] Definition: Staff are the employees

Figure AMD1.2 — VSV main screen showing track and metadata contents.

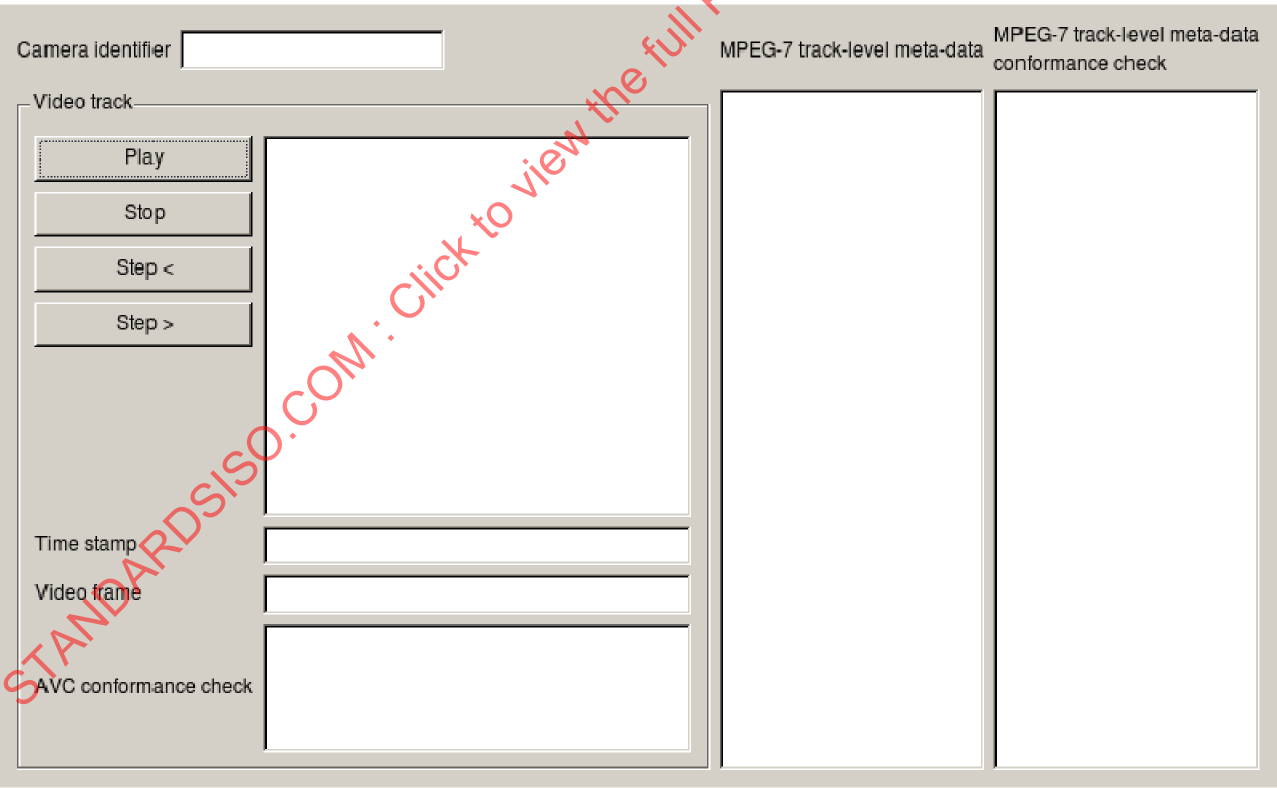


Figure AMD1.3 — VSV track screen allows video and metadata playback.

These are the functions used to read the appropriate Boxes, functions to read the metadata and GUI functions.

8.5.2 File-level

- unpackVSAF()
- MP4OpenMovieFile()
- ISOGetFileMeta()
- ISOGetPrimaryItemData()
- getTracks()
- MP4GetMovieIndTrack()
- MP4GetTrackMedia()
- MP4GetMediaHandlerDescription()
- ISOGetMediaDuration()
- ISOGetMediaTimeScale()
- ISOGetTrackMeta()
- getVideo()
- MP4GetTrackReference()
- MP4GetTrackMedia()
- MP4GetMediaHandlerDescription()
- getTimedMetadata()
- MP4CreateTrackReader()
- MP4TrackReaderGetCurrentSampleDescription()
- ISOGetAVCSampleDescription()
- ISOGetSampleDescriptionDimensions()
- ISOGetAVCSampleDescriptionPS()
- MP4SetHandleSize()
- MP4NewHandle()
- MP4GetHandleSize()
- MP4TrackReaderGetNextAccessUnit()
- SampleGetNALUnit()
- ISOGetSampleDescriptionType()
- ISOGetVideoSurveillanceMetadataSampleDescription()
- MP4DisposeHandle()

8.5.3 Track-level

- getAVCProfile()
- getmp7()
- metadatacheck()
- SetBGR()
- SetYUVInfo()
- SetYUVImage()
- Convert()
- GetRGBImage()
- GetRGBSize()
- vsaftrackinit()
- getFreeText()
- getNText()
- getStructuredAnnotation()
- getDescriptionMetadataPublicIdentifier()
- getDescriptionMetadataGPSLongitude()
- getDescriptionMetadataGPSLatitude()
- getDescriptionMetadataCreationTime()
- getInstrumentToolName()
- getNInstrumentSetting()
- getInstrumentSettingName()
- getInstrumentSettingValue()
- getNVideo2DSpatialCoordinates()
- getVideo2DSpatialCoordinatesIndices()
- getVideo2DSpatialCoordinatesID()
- getVideo2DSpatialCoordinatesName()
- getNVideo2DSpatialCoordinatesValues()
- getVideo2DSpatialCoordinatesValues()
- getVideoMediaInformationMediaProfileMediaInstanceInstanceIdentifier()

- getVideoMediaInformationMediaProfileMediaInstanceMediaLocatorMediaUri()
- getVideoMediaInformationMediaProfileMediaInstanceMediaLocatorStreamID()
- getVideoMediaInformationMediaIdentificationEntityIdentifier()
- getNVideoMediaInformationMediaIdentificationEntityIdentifierVideoDomain()
- getVideoMediaInformationMediaIdentificationEntityIdentifierVideoDomain()
- getVideoMediaTimeMediaTimePoint()
- getNVideoStillRegions()
- getVideoStillRegionID()
- getVideoTemporalDecompositionStillRegionMediaTimePoint()
- getVideoTemporalDecompositionStillRegionSpatialLocatorBox()
- getVideoTemporalDecompositionStillRegionDominantColor()
- getVideoTemporalDecompositionStillRegionScalableColor()
- getVideoTemporalDecompositionStillRegionGridLayoutDescriptorsGrid()
- getVideoTemporalDecompositionStillRegionGridLayoutDescriptorsMask()
- getNVideoTemporalDecompositionStillRegionGridLayoutDescriptors()
- getVideoTemporalDecompositionStillRegionGridLayoutDescriptorsDominantColor()
- getVideoTemporalDecompositionStillRegionGridLayoutDescriptorsScalableColor()
- getNVideoRelationships()
- getVideoRelationshipsID()
- getNVideoRelationshipsRelation()
- getVideoRelationshipsRelationID()
- getVideoRelationshipsRelationType()
- getVideoRelationshipsRelationStrength()

8.6 Modifications to the ISO Base Media File Format reference software

The ISO Base Media File Format reference was modified and extended to support the functionality required by the VSAF specification. These modifications include the MAFIdentification and the CAMIdentification Boxes. The modifications are listed below in Table AMD1.2.

Table AMD1.2 — Modifications the ISO Base Media File Format reference software

a) ISOSampleDescriptions.c

1) Functions

i) ISONewVideoSurveillanceMetadataSampleDescription

b) ISOMovies.h

1) MP4_FOUR_CHAR_CODE

i) ISOVSAFBrand

2) Function prototypes

i) ISONewMAFIdentification

ii) ISONewCameraIdentification

iii) ISONewVSAFMetadataSampleDescription

iv) ISONewVideoSurveillanceMetadataSampleDescription

c) MP4Atoms.h

1) MP4_FOUR_CHAR_CODE

i) ISOVideoSurveillanceMetadataSampleEntryBoxType

ii) ISOVideoSurveillanceMetadataSampleConfigBoxType

iii) ISOMAFIdentificationBoxType

iv) ISOCameraIdentificationBoxType

2) Structs

i) ISOVideoSurveillanceMetadataSampleEntryBox

ii) ISOVideoSurveillanceMetadataSampleConfigBox

iii) ISOCameraIdentificationBox

iv) ISOMAFIdentificationBox

3) Function prototypes

i) ISOCreateVideoSurveillanceMetadataSampleConfigBox

ii) ISOCreateVideoSurveillanceMetadataSampleEntryBox

iii) ISOCreateCameraIdentificationBox

iv) ISOCreateMAFIdentificationBox

d) ISOMeta.c

    1) Function declarations

        i) ISONewMAFIdentification

        ii) ISONewCameraIdentification

e) MetaAtom.c

    1) Added to addAtom function

        i) ISOMAFIdentificationBoxType

        ii) ISOCameraIdentificationBoxType

f) MP4Atoms.c

    1) Added to MP4CreateAtom function

        i) ISOVideoSurveillanceMetadataSampleConfigBoxType

        ii) ISOVideoSurveillanceMetadataSampleEntryBoxType

        iii) ISOCameraIdentificationBoxType

        iv) ISOMAFIdentificationBoxType

g) MP4Movies.h

    1) MP4_FOUR_CHAR_CODE

        i) ISOVSMDTrackReferenceType

h) New Files

    1) ISOVideoSurveillanceMetadataSampleConfigBox.c

    2) ISOVideoSurveillanceMetadataSampleEntryBox.c

    3) ISOMAFIdentificationBox.c

    4) ISOCameraIdentificationBox.c

8.7 VSAF MPEG-7 API details

The VSAF MPEG-7 API (Application Programming Interface) provides the mechanism with which to populate (pack) or extract (unpack) MPEG-7 metadata. The use of this mechanism helps to establish the conformance of the MPEG-7 metadata to the VSAF specification.

8.7.1 Metadata related functions in VSAF MPEG-7 API

Tables AMD1.3 and AMD1.4 show the "get" functions used in the informative reference software (VSV). The tables are split into **Required** and **Optional** parts. If the Required elements are not present, the conformance test fails. The metadata will be seen as conformant if it validates against the VSAF XML schema. Certain further elements, if included by virtue of their parent elements, might be required to be included, dictated by the rules of the schema.