INTERNATIONAL STANDARD

ISO/IEC 14496-2

First edition
1999-12-01

AMENDMENT 1
2000-07-15

Information technology — Coding of audio-visual objects —

Part 2: **Visual**

AMENDMENT 1: Visual extensions

iTeh STANDARD PREVIEW
(standards.iteh.ai)

Technologies de l'information — Codage des objets audiovisuels —

Partie 2: Codage visuel

ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-4b4b-4f5e-91b8-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

AMENDEMENT 1: Extensions visuelles

Reference number
ISO/IEC 14496-2:1999/Amd.1:2000(E)

© ISO/IEC 2000

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.

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-91b8-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

© ISO/IEC 2000

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

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

Printed in Switzerland

© ISO/IEC 2000 – All rights reserved

Contents

1 Scope 1
2 Normative references 1
3 Terms and definitions 2
4 Abbreviations and symbols 12
4.1 Arithmetic operators 12
4.2 Logical operators 13
4.3 Relational operators 13
4.4 Bitwise operators 13
4.5 Conditional operators 13
4.6 Assignment 13
4.7 Mnemonics 13
4.8 Constants 14
5 Conventions 14
5.1 Method of describing bitstream syntax 14
5.2 Definition of functions 15
5.2.1 Definition of next_bits() function 15
5.2.2 Definition of bytealigned() function 15
5.2.3 Definition of nextbits_bytealigned() function 15
5.2.4 Definition of next_start_code() function 16
5.2.5 Definition of next_resync_marker() function 16
5.2.6 Definition of transparent_mb() function 16
5.2.7 Definition of transparent_block() function 16
5.2.8 Definition of byte_align_for_upstream() function 16
5.3 Reserved, forbidden and marker_bit 16
5.4 Arithmetic precision 17
6 Visual bitstream syntax and semantics 17
6.1 Structure of coded visual data 17
6.1.1 Visual object sequence 18
6.1.2 Visual object 18
6.1.3 Video object 18
6.1.4 Mesh object 24
6.1.5 FBA object 25
6.1.6 3D Mesh Object 29
6.2 Visual bitstream syntax 30
6.2.1 Start codes 30
6.2.2 Visual Object Sequence and Visual Object 33
6.2.3 Video Object Layer 35
6.2.4 Group of Video Object Plane 40
6.2.5 Video Object Plane and Video Plane with Short Header 40
6.2.6 Macroblock 52
6.2.7 Block 58
6.2.8 Still Texture Object 59
6.2.9 Mesh Object 73
6.2.10 FBA Object 75
6.2.11 3D Mesh Object 85
6.2.12 Upstream message 103
6.3 Visual bitstream semantics 104
6.3.1 Semantic rules for higher syntactic structures 104
6.3.2 Visual Object Sequence and Visual Object 104

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5c-9168-4b2302f34d1f/iso-iec-14496-2-1999-amd-1-2000

6.3.3 Video Object Layer 110
6.3.4 Group of Video Object Plane 120
6.3.5 Video Object Plane and Video Plane with Short Header 120
6.3.6 Macroblock related 131
6.3.7 Block related 135
6.3.8 Still texture object 135
6.3.9 Mesh object 143
6.3.10 FBA object 145
6.3.11 3D Mesh Object 151
6.3.12 Upstream message 162

7 The visual decoding process 164
7.1 Video decoding process 165
7.2 Higher syntactic structures 166
7.3 VOP reconstruction 166
7.4 Texture decoding 167
7.4.1 Variable length decoding 167
7.4.2 Inverse scan 169
7.4.3 DC and AC prediction for intra macroblocks 170
7.4.4 Inverse quantisation 172
7.4.5 Inverse DCT 176
7.4.6 Upsampling of the Inverse DCT output for Reduced Resolution VOP 176
7.5 Shape decoding 178
7.5.1 Higher syntactic structures 178
7.5.2 Macroblock decoding 179
7.5.3 Arithmetic decoding 189
7.5.4 Spatial scalable binary shape decoding 191
7.5.5 Grayscale Shape Decoding 200
7.5.6 Multiple Auxiliary Component Decoding 203
7.6 Motion compensation decoding 203
7.6.1 Padding process 203
7.6.2 Sample interpolation for non-integer motion vectors 207
7.6.3 General motion vector decoding process 209
7.6.4 Unrestricted motion compensation 211
7.6.5 Vector decoding processing and motion-compensation in progressive P- and S(GMC)-VOP 212
7.6.6 Overlapped motion compensation 214
7.6.7 Temporal prediction structure 215
7.6.8 Vector decoding process of non-scalable progressive B-VOPs 216
7.6.9 Motion compensation in non-scalable progressive B-VOPs 216
7.6.10 Motion Compensation Decoding of Reduced Resolution VOP 221
7.7 Interlaced video decoding 226
7.7.1 Field DCT and DC and AC Prediction 226
7.7.2 Motion compensation 227
7.8 Sprite decoding 236
7.8.1 Higher syntactic structures 236
7.8.2 Sprite Reconstruction 237
7.8.3 Low-latency sprite reconstruction 237
7.8.4 Sprite reference point decoding 238
7.8.5 Warping 239
7.8.6 Sample reconstruction 241
7.8.7 GMC decoding 242
7.9 Generalized scalable decoding 243
7.9.1 Temporal scalability 245
7.9.2 Spatial scalability 248
7.10 Still texture object decoding 252
7.10.1 Decoding of the DC subband 253
7.10.2 ZeroTree Decoding of the Higher Bands 254
7.10.3 Inverse Quantisation 259
7.10.4 Still Texture Error Resilience 267
7.10.5 Wavelet Tiling 270
7.10.6 Scalable binary shape object decoding 271

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-91b8-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

7.11 Mesh object decoding 277
7.11.1 Mesh geometry decoding 278
7.11.2 Decoding of mesh motion vectors 281
7.12 FBA object decoding 283
7.12.1 Frame based face object decoding 283
7.12.2 DCT based face object decoding 284
7.12.3 Decoding of the viseme parameter fap 1 285
7.12.4 Decoding of the viseme parameter fap 2 286
7.12.5 Fap masking 286
7.12.6 Frame Based Body Decoding 286
7.12.7 DCT based body object decoding 287
7.13 3D Mesh Object Decoding 288
7.13.1 Start codes and bit stuffing 290
7.13.2 The Topological Surgery decoding process. 290
7.13.3 The Forest Split decoding process 292
7.13.4 Header decoder 293
7.13.5 partition type 294
7.13.6 Vertex Graph Decoder 296
7.13.7 Triangle Tree Decoder 299
7.13.8 Triangle Data Decoder 300
7.13.9 Forest Split decoder 305
7.13.10 Arithmetic decoder 312
7.14 NEWPRED mode decoding 317
7.14.1 Decoder Definition 317
7.14.2 Upstream message 317
7.15 Output of the decoding process 317
7.15.1 Video data 318
7.15.2 2D Mesh data 318
7.15.3 Face animation parameter data 318

8 Visual-Systems Composition Issues 318
8.1 Temporal Scalability Composition 318
8.2 Sprite Composition 319
8.3 Mesh Object Composition 320
8.4 Spatial Scalability composition 321

9 Profiles and Levels 321
9.1 Visual Object Types 321
9.2 Visual Profiles 324
9.3 Visual Profiles@Levels 325
9.3.1 Natural Visual 325
9.3.2 Synthetic Visual 325
9.3.3 Synthetic/Natural Hybrid Visual 327

Annex A (normative) Coding transforms 328
A.1 Discrete cosine transform for video texture 328
A.2 Discrete wavelet transform for still texture 329
A.2.1 Adding the mean 329
A.2.2 Wavelet filter 329
A.2.3 Symmetric extension 330
A.2.4 Decomposition level 330
A.2.5 Shape adaptive wavelet filtering and symmetric extension 331
A.3 Shape-Adaptive DCT (SA-DCT) 332
A.3.1 Definition of Forward SA-DCT 332
A.3.2 Definition of Inverse SA-DCT 334
A.4 SA-DCT with DC Separation and ΔDC Correction (ΔDC-SA-DCT) 335
A.4.1 Definition of Forward ΔDC-SA-DCT 336
A.4.2 Definition of Inverse ΔDC-SA-DCT 336

Annex B (normative) Variable length codes and arithmetic decoding 338
B.1 Variable length codes 338
B.1.1 Macroblock type 338

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-9108-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

B.1.2 Macroblock pattern 340
B.1.3 Motion vector 342
B.1.4 DCT coefficients 344
B.1.5 Shape Coding 354
B.1.6 Sprite Coding 359
B.1.7 DCT based facial object decoding 360
B.1.8 Shape decoding for still texture object 369
B.2 Arithmetic Decoding 370
B.2.1 Aritmetic decoding for still texture object 370
B.2.2 Arithmetic decoding for shape decoding 373
B.2.3 FBA Object Decoding 376

Annex C (normative) Face and body object decoding tables and definitions 378

Annex D (normative) Video buffering verifier 411
D.1 Introduction 411
D.2 Video Rate Buffer Model Definition 411
D.3 Comparison between ISO/IEC 14496-2 VBV and the ISO/IEC 13818-2 VBV (Informative) 414
D.4 Video Complexity Model Definition 415
D.5 Video Reference Memory Model Definition 417
D.6 Interaction between VBV, VCV and VMV (informative) 418
D.7 Video Presentation Model Definition (informative) 418

Annex E (informative) Features supported by the algorithm 420
E.1 Error resilience 420
E.1.1 Resynchronization 420
E.1.2 Data Partitioning 421
E.1.3 Reversible VLC 421
E.1.4 Decoder Operation 422
E.1.5 Adaptive Intra Refresh (AIR) Method 425
E.1.6 NEWPRED 427
E.2 Complexity Estimation 429
E.3 Resynchronization in Case of Unknown Video Header Format 429

Annex F (informative) Preprocessing and postprocessing 431
F.1 VOP Generation Tools: Automatic and Semi-automatic Segmentations 431
F.1.1 Automatic Segmentation 431
F.1.2 Semi-automatic Segmentation 441
F.1.3 References 449
F.2 Bounding Rectangle of VOP Formation 450
F.3 Postprocessing for Coding Noise Reduction 451
F.3.1 Deblocking filter 451
F.3.2 Deringing filter 453
F.3.3 Further issues 455
F.4 Chrominance Decimation and Interpolation Filtering for Interlaced Object Coding 455

Annex G (normative) Profile and level indication and restrictions 457

Annex H (informative) Patent statements 460
H.1 Patent statements for ISO/IEC 14496 Version 1 460
H.2 Patent statements for the extensions provided in ISO/IEC 14496 Version 2 461

Annex I (informative) Encoder Complexity Reduction Based on Intelligent Pre-Quantisation 463
I.1 Introduction 463
I.2 Feature Selection and Pre-quantisation 463
I.3 Model Verification and Threshold Setting 465
I.3.1 H.263 Quantiser 465
I.3.2 MPEG-4 Quantiser 465

Annex J (normative) View dependent object scalability 467
J.1 Introduction 467
J.2 Decoding Process of a View-Dependent Object 467
J.2.1 General Decoding Scheme 467
J.2.2 Computation of the View-Dependent Scalability parameters 469
J.2.3 VD mask computation 471

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/b6c4f7f1-1b4b-4b5e-9168-f9329b215d01/iso-iec-14496-2-1999-amd-1-2000

J.2.4 Differential mask computation ... 472
J.2.5 DCT coefficients decoding ... 472
J.2.6 Texture update ... 472
J.2.7 IDCT ... 473

Annex K (normative) Decoder Configuration Information ... 474
K.1 Introduction ... 474
K.2 Description of the set up of a visual decoder (informative) ... 474
K.2.1 Processing of decoder configuration information ... 475
K.3 Specification of decoder configuration information ... 476
K.3.1 VideoObject ... 476
K.3.2 StillTextureObject ... 476
K.3.3 MeshObject ... 477
K.3.4 FaceObject ... 477
K.3.5 3DMeshObject ... 477

Annex L (informative) Rate control ... 478
L.1 Frame Rate Control ... 478
L.1.1 Introduction ... 478
L.1.2 Description ... 478
L.1.3 Summary ... 482
L.2 Multiple Video Object Rate Control ... 482
L.2.1 Initialization ... 482
L.2.2 Quantisation Level Calculation for I-frame and first P-frame ... 482
L.2.3 Update Rate-Distortion Model ... 485
L.2.4 Post-Frameskip Control ... 485
L.3 Macroblock Rate Control ... 487
L.3.1 Rate-Distortion Model ... 487
L.3.2 Target Number of Bits for Each Macroblock ... 488
L.3.3 Macroblock Rate Control ... 488

Annex M (informative) Binary shape coding ... 491
M.1 Introduction ... 491
M.2 Context-Based Arithmetic Shape Coding ... 491
M.2.1 Intra Mode ... 491
M.2.2 Inter Mode ... 492
M.3 Texture Coding of Boundary Blocks ... 493
M.4 Encoder Architecture ... 493
M.5 Encoding Guidelines ... 494
M.5.1 Lossy Shape Coding ... 494
M.5.2 Coding Mode Selection ... 495
M.6 Conclusions ... 495
M.7 References ... 495

Annex N (normative) Visual profiles@levels ... 497

Annex O (informative) 3D Mesh Coding ... 501
O.1 Introduction ... 501
O.2 Topological Surgery Representation ... 501
O.2.1 Simple Polygon Representation ... 502
O.2.2 Vertex Graph representation ... 503
O.3 Encoding guidelines for 3D Mesh Coding ... 504
O.3.1 Topological Surgery Encoding ... 504
O.3.2 Support for non-manifolds and Non-orientable manifolds ... 505
O.3.3 Support for Error Resilience ... 507
O.4 Encoder considerations for efficient compression of Vertex Properties ... 511
O.5 Progressive Forest Split Representation ... 512
O.5.1 Encoding the Forest ... 512
O.5.2 Support for meshes with polygonal faces ... 513
O.5.3 Method for generating of a PFS Representation of a Triangular 3D Mesh ... 513
O.5.4 Topological Tests ... 514
O.5.5 Geometric Tests ... 516
O.6 Complexity estimation for Computational Graceful Degradation ... 516

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-9168-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

O.7 QoS for SNHC through upstream 518
O.8 References 519

Bibliography 521

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-91b8-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. 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 Amendment 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 International Standard ISO/IEC 14496-2:1999 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

Annexes A, B, C, D, G, J, K and N form a normative part of this Amendment. Annexes E, F, H, I, L, M and O are for information only.

ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-91b8-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

© ISO/IEC 2000 – All rights reserved

Introduction

Purpose

This part of ISO/IEC 14496 was developed in response to the growing need for a coding method that can facilitate access to visual objects in natural and synthetic moving pictures and associated natural or synthetic sound for various applications such as digital storage media, internet, various forms of wired or wireless communication etc. The use of ISO/IEC 14496 means that motion video can be manipulated as a form of computer data and can be stored on various storage media, transmitted and received over existing and future networks and distributed on existing and future broadcast channels.

Application

The applications of ISO/IEC 14496 cover, but are not limited to, such areas as listed below:

IMM Internet Multimedia

IVG Interactive Video Games

IPC Interpersonal Communications (videoconferencing, videophone, etc.)

ISM Interactive Storage Media (optical disks, etc.)

MMM Multimedia Mailing

NDB Networked Database Services (via ATM, etc.)

RES Remote Emergency Systems

RVS Remote Video Surveillance

WMM Wireless Multimedia

Multimedia

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-91b8-
[illegible]5401/iso-iec-14496-2-1999-amd-1-2000

Profiles and levels

ISO/IEC 14496 is intended to be generic in the sense that it serves a wide range of applications, bitrates, resolutions, qualities and services. Furthermore, it allows a number of modes of coding of both natural and synthetic video in a manner facilitating access to individual objects in images or video, referred to as content based access. Applications should cover, among other things, digital storage media, content based image and video databases, internet video, interpersonal video communications, wireless video etc. In the course of creating ISO/IEC 14496, various requirements from typical applications have been considered, necessary algorithmic elements have been developed, and they have been integrated into a single syntax. Hence ISO/IEC 14496 will facilitate the bitstream interchange among different applications.

This part of ISO/IEC 14496 includes one or more complete decoding algorithms as well as a set of decoding tools. Moreover, the various tools of this part of ISO/IEC 14496 as well as that derived from ISO/IEC 13818-2:1996 can be combined to form other decoding algorithms. Considering the practicality of implementing the full syntax of this part of ISO/IEC 14496, however, a limited number of subsets of the syntax are also stipulated by means of “profile” and “level”.

A “profile” is a defined subset of the entire bitstream syntax that is defined by this part of ISO/IEC 14496. Within the bounds imposed by the syntax of a given profile it is still possible to require a very large variation in the performance of encoders and decoders depending upon the values taken by parameters in the bitstream.

In order to deal with this problem "levels" are defined within each profile. A level is a defined set of constraints imposed on parameters in the bitstream. These constraints may be simple limits on numbers. Alternatively they may take the form of constraints on arithmetic combinations of the parameters.

Object based coding syntax

Video object

A *video object* in a scene is an entity that a user is allowed to access (seek, browse) and manipulate (cut and paste). The instances of video objects at a given time are called *video object planes* (VOPs). The encoding process generates a coded representation of a VOP as well as composition information necessary for display. Further, at the decoder, a user may interact with and modify the composition process as needed.

The full syntax allows coding of rectangular as well as arbitrarily shaped video objects in a scene. Furthermore, the syntax supports both nonscalable coding and scalable coding. Thus it becomes possible to handle normal scalabilities as well as object based scalabilities. The scalability syntax enables the reconstruction of useful video from pieces of a total bitstream. This is achieved by structuring the total bitstream in two or more layers, starting from a standalone base layer and adding a number of enhancement layers. The base layer can be coded using a non-scalable syntax, or in the case of picture based coding, even using a syntax of a different video coding standard.

To ensure the ability to access individual objects, it is necessary to achieve a coded representation of its shape. A natural video object consists of a sequence of 2D representations (at different points in time) referred to here as VOPs. For efficient coding of VOPs, both temporal redundancies as well as spatial redundancies are exploited. Thus a coded representation of a VOP includes representation of its shape, its motion and its texture.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

FBA object

The FBA object is a collection of nodes in a scene graph which are animated by the FBA (Face and Body Animation) object bitstream. The FBA object is controlled by two separate bitstreams. The first bitstream, called BIFS, contains instances of Body Definition Parameters (BDPs) in addition to Facial Definition Parameters (FDPs) , and the second bitstream, FBA bitstream, contains Body Animation Parameters (BAPs) together with Facial Animation Parameters (FAPs).

ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-91b8-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

A 3D (or 2D) *face object* is a representation of the human face that is structured for portraying the visual manifestations of speech and facial expressions adequate to achieve visual speech intelligibility and the recognition of the mood of the speaker. A face object is animated by a stream of *face animation parameters (FAP)* encoded for low-bandwidth transmission in broadcast (one-to-many) or dedicated interactive (point-to-point) communications. The FAPs manipulate key feature control points in a mesh model of the face to produce animated visemes for the mouth (lips, tongue, teeth), as well as animation of the head and facial features like the eyes. FAPs are quantized with careful consideration for the limited movements of facial features, and then prediction errors are calculated and coded arithmetically. The remote manipulation of a face model in a terminal with FAPs can accomplish lifelike visual scenes of the speaker in real-time without sending pictorial or video details of face imagery every frame.

A simple streaming connection can be made to a decoding terminal that animates a default face model. A more complex session can initialize a custom face in a more capable terminal by downloading *face definition parameters (FDP)* from the encoder. Thus specific background images, facial textures, and head geometry can be portrayed. The composition of specific backgrounds, face 2D/3D meshes, texture attribution of the mesh, etc. is described in ISO/IEC 14496-1:1999. The FAP stream for a given user can be generated at the user's terminal from video/audio, or from text-to-speech. FAPs can be encoded at bitrates up to 2-3kbit/s at necessary speech rates. Optional temporal DCT coding provides further compression efficiency in exchange for delay. Using the facilities of ISO/IEC 14496-1:1999, a composition of the animated face model and synchronized, coded speech audio (low-bitrate speech coder or text-to-speech) can provide an integrated low-bandwidth audio/visual speaker for broadcast applications or interactive conversation.

Limited scalability is supported. Face animation achieves its efficiency by employing very concise motion animation controls in the channel, while relying on a suitably equipped terminal for rendering of moving 2D/3D faces with non-normative models held in local memory. Models stored and updated for rendering in the terminal can be simple or complex. To support speech intelligibility, the normative specification of FAPs intends for their selective or complete use as signaled by the encoder. A masking scheme provides for selective transmission of

FAPs according to what parts of the face are naturally active from moment to moment. A further control in the FAP stream allows face animation to be suspended while leaving face features in the terminal in a defined quiescent state for higher overall efficiency during multi-point connections.

A body model is a representation of a virtual human or human-like character that allows portraying body movements adequate to achieve nonverbal communication and general actions. A body model is animated by a stream of *body animation parameters* (BAP) encoded for low-bitrate transmission in broadcast and dedicated interactive communications. The BAPs manipulate independent degrees of freedom in the skeleton model of the body to produce animation of the body parts. The BAPs are quantized considering the joint limitations, and prediction errors are calculated and coded arithmetically. Similar to the face, the remote manipulation of a body model in a terminal with BAPs can accomplish lifelike visual scenes of the body in real-time without sending pictorial and video details of the body every frame.

The BAPs, if correctly interpreted, will produce reasonably similar high level results in terms of body posture and animation on different body models, also without the need to initialize or calibrate the model.The BDP set defines the set of parameters to transform the default body to a customized body optionally with its body surface, body dimensions, and texture.

The *body definition parameters (BDP)* allow the encoder to replace the local model of a more capable terminal. BDP parameters include body geometry, calibration of body parts, degrees of freedom, and optionally deformation information.

The FBA Animation specification is defined in ISO/IEC 14496-1:1999/Amd.1:2000 and this part of ISO/IEC 14496. This clause is intended to facilitate finding various parts of specification. As a rule of thumb, FAP and BAP specification is found in the part 2, and FDP and BDP specification in the part 1. However, this is not a strict rule. For an overview of FAPs/BAPs and their interpretation, read subclauses "6.1.5.2 Facial animation parameter set", "6.1.5.3 Facial animation parameter units", "6.1.5.4 Description of a neutral face" as well as the Table C-1. The viseme parameter is documented in subclause "7.12.3 Decoding of the viseme parameter fap 1" and the Table C-5 in annex C. The expression parameter is documented in subclause "7.12.4 Decoding of the expression parameter fap 2" and the Table C-3. FBA bitstream syntax is found in subclauses "6.2.10 FBA Object", semantics in "6.3.10 FBA Object", and subclause "7.12 FBA object decoding" explains in more detail the FAP/BAP decoding process. FAP/BAP masking and interpolation is explained in subclauses "6.3.11.1 FBA Object Plane", "7.12.1.1 Decoding of FBA", "7.12.5 FBA masking" . The FIT interpolation scheme is documented in subclause "7.2.5.3.2.4 FIT" of ISO/IEC 14496-1:1999. The FDPs and BDPs and their interpretation are documented in subclause "7.2.5.3.2.6 FDP" of ISO/IEC 14496-1:1999. In particular, the FDP feature points are documented in Figure C-1. Details on body models are documented in Annex C.

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbcd1711-1b4b-4f5e-91b8-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

Mesh object

A 2D *mesh object* is a representation of a 2D deformable geometric shape, with which synthetic video objects may be created during a composition process at the decoder, by spatially piece-wise warping of existing video object planes or still texture objects. The instances of mesh objects at a given time are called *mesh object planes* (mops). The geometry of mesh object planes is coded losslessly. Temporally and spatially predictive techniques and variable length coding are used to compress 2D mesh geometry. The coded representation of a 2D mesh object includes representation of its geometry and motion.

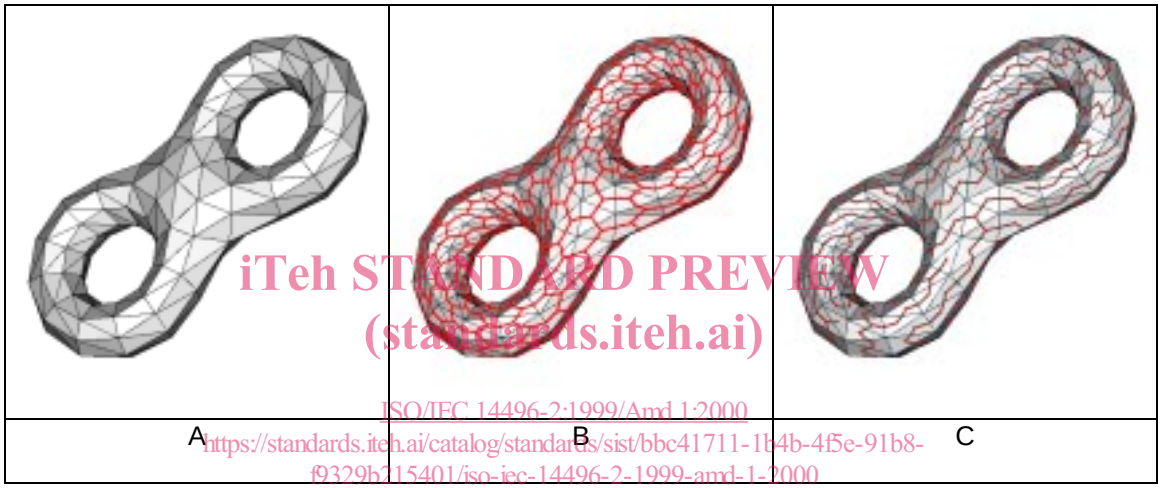
3D Mesh Object

The 3D Mesh Object is a 3D polygonal model that can be represented as an IndexedFaceSet or Hierarchical 3D Mesh node in BIFS. It is defined by the position of its vertices (geometry), by the association between each face and its sustaining vertices (connectivity), and optionally by colours, normals, and texture coordinates (properties). Properties do not affect the 3D geometry, but influence the way the model is shaded. 3D mesh coding (3DMC) addresses the efficient coding of 3D mesh object. It comprises a basic method and several options. The basic 3DMC method operates on manifold model and features incremental representation of single resolution 3D model. The model may be triangular or polygonal – the latter are triangulated for coding purposes and are fully recovered in the decoder. Options include: (a) support for computational graceful degradation control; (b) support for non-manifold model; (c) support for error resilience; and (d) quality scalability via hierarchical transmission of levels of detail with implicit support for smooth transition between consecutive levels. The compression of application-specific geometry streams (Face Animation Parameters) and generalized animation parameters (BIFS Anim) are currently addressed elsewhere in this part of ISO/IEC 14496.

In 3DMC, the compression of the connectivity of the 3D mesh (e.g. how edges, faces, and vertices relate) is lossless, whereas the compression of the other attributes (such as vertex coordinates, normals, colours, and texture coordinates) may be lossy.

Single Resolution Mode

The incremental representation of a single resolution 3D model is based on the Topological Surgery scheme. For manifold triangular 3D meshes, the Topological Surgery representation decomposes the connectivity of each *connected component* into a *simple polygon* and a *vertex graph*. All the triangular faces of the 3D mesh are connected in the simple polygon forming a *triangle tree*, which is a spanning tree in the dual graph of the 3D mesh. Figure V2 - 1 shows an example of a triangular 3D mesh, its dual graph, and a triangle tree. The vertex graph identifies which pairs of boundary edges of the simple polygon are associated with each other to reconstruct the connectivity of the 3D mesh. The triangle tree does not fully describe the triangulation of the simple polygon. The missing information is recorded as a *marching edge.*



iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-91b8-f9329b215401/iso-iec-14496-2-1999-amd-1-2000

Figure V2 - 1 -- A triangular 3D mesh (A), its dual graph (B), and a triangle tree (C)

For manifold 3D meshes, the connectivity is represented in a similar fashion. The polygonal faces of the 3D mesh are connected in a simple polygon forming a *face tree*. The faces are triangulated, and which edges of the resulting triangular 3D mesh are edges of the original 3D mesh is recorded as a sequence of *polygon_edge* bits. The face tree is also a spanning tree in the dual graph of the 3D mesh, and the vertex graph is always composed of edges of the original 3D mesh.

The vertex coordinates and optional properties of the 3D mesh (normals, colours, and texture coordinates) are quantized, predicted as a function of decoded ancestors with respect to the order of traversal, and the errors are entropy encoded.

Incremental Representation

When a 3D mesh is downloaded over networks with limited bandwidth (e.g. PSTN), it may be desired to begin decoding and rendering the 3D mesh before it has all been received. Moreover, content providers may wish to control such incremental representation to present the most important data first. The basic 3DMC method supports this by interleaving the data such that each triangle may be reconstructed as it is received. Incremental representation is also facilitated by the options of hierarchical transmission for quality scalability and partitioning for error resilience.

Hierarchical Mode

An example of a 3D mesh represented in hierarchical mode is illustrated in Figure V2 - 2. The hierarchical mode allows the decoder to show progressively better approximations of the model as data are received. The hierarchical 3D mesh decomposition can also be organized in the decoder as layered detail, and view-dependent expansion of this detail can be subsequently accomplished during a viewer's interaction with the 3D model.

Downloadable scalability of 3D model allows decoders with widely varied rendering performance to use the content, without necessarily making repeated requests to the encoder for specific versions of the content and without the latencies of updating view-dependent model from the encoder. This quality scalability of 3D meshes is complementary to scaleable still texture and supports bitstream scalability and downloading of quality hierarchies of hybrid imagery to the decoder.

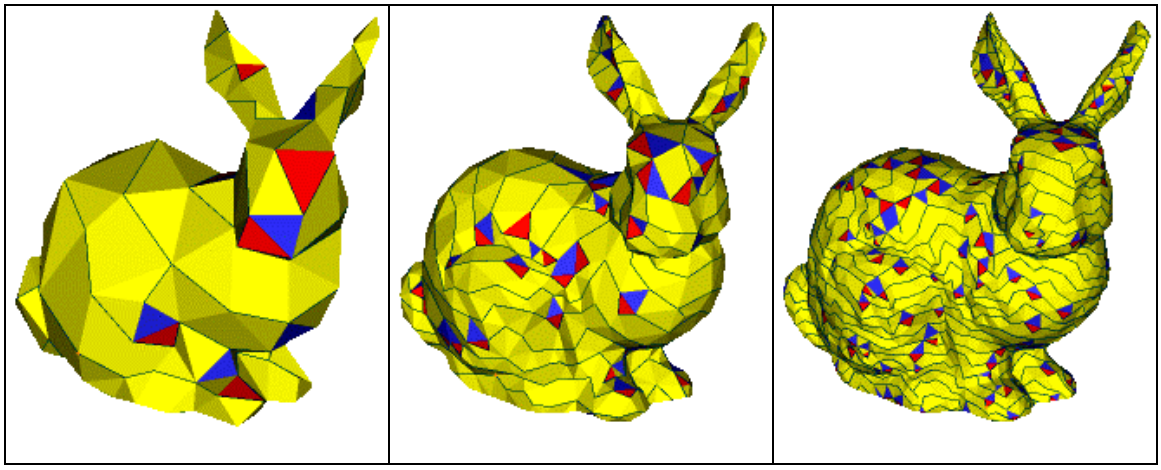
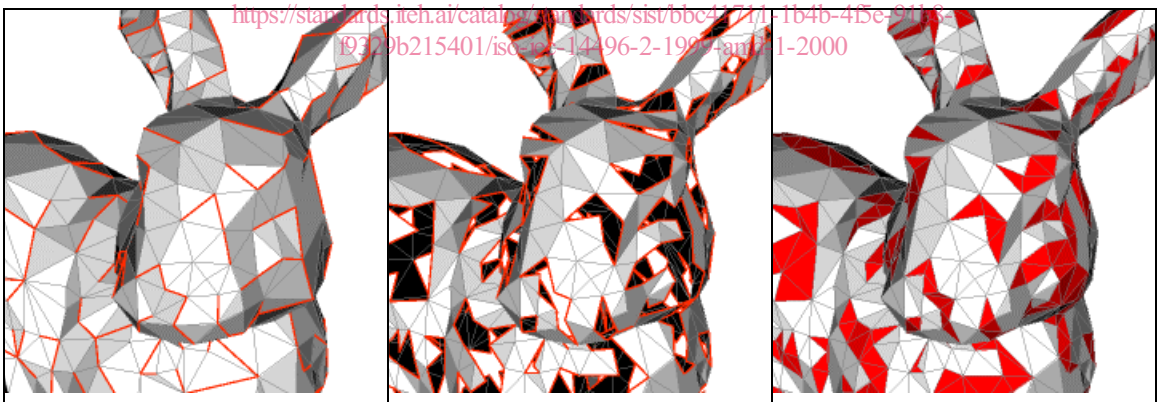


Figure V2 - 2 -- A hierarchy of levels of detail

The hierarchical representation of a 3D model is based on the Progressive Forest Split scheme. In the Progressive Forest Split representation, a manifold 3D mesh is represented as a *base 3D mesh* followed by a sequence of *forest split* operations. The base 3D mesh is encoded as a single resolution 3D mesh using the Topological Surgery scheme. Each forest split operation is composed of a *forest* in the graph of the current 3D mesh, and a *sequence of simple polygons*. Figure V2 - 3 illustrates the method. To prevent visual artifacts which may occur while switching from a level of detail to the next one, the data structure supports smooth transition between consecutive levels of detail in the form of linear interpolation of vertex positions.

iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-91b8-f9329b215401/iso-iec-14496-2-1999-amd-1-2000



A	B	C

Figure V2 - 3 -- The Forest Split operation. A 3D mesh with a forest of edges marked (A). The tree loops resulting from cutting the 3D mesh through the forest edges (B). Each tree loop is filled by a simple polygon composed of polygonal faces (C).

Error Resilience for 3D mesh object

If the 3D mesh is partitioned into independent parts, it may be possible to perform more efficient data transmission in an error-prone environment, e.g., an IP network or datacasting service in a broadcast TV network. It must be possible to resynchronize after a channel error, and continue data transmission and rendering from that point instead of starting over from scratch. Even with the presence of channel errors, the decoder can start decoding and rendering from the next partition that is received intact from the channel.

Flexible partitioning methods can be used to organize the data, such that it fits the underlying network packet structure more closely, and overhead is reduced to the minimum. To allow flexible partitioning, several connected

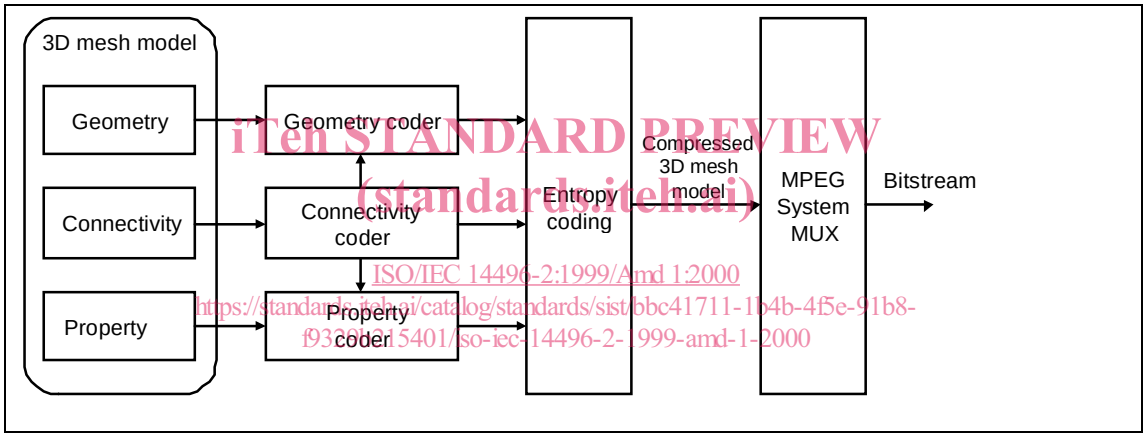
components may be merged into one partition, where as a large connected component may be divided into several independent partitions. Merging and dividing of connected components using different partition types can be done at any point in the 3D mesh object.

Stitching for Non-Manifold and Non-Orientable Meshes

The connectivity of a non-manifold and non-orientable 3D mesh is represented as a manifold 3D mesh and a sequence of *stitches*. Each stitch describes how to identify one or more pairs of vertices along two linear paths of edges, and each one of these paths is contained in one of the vertex graphs that span the connected components of the manifold 3D mesh.

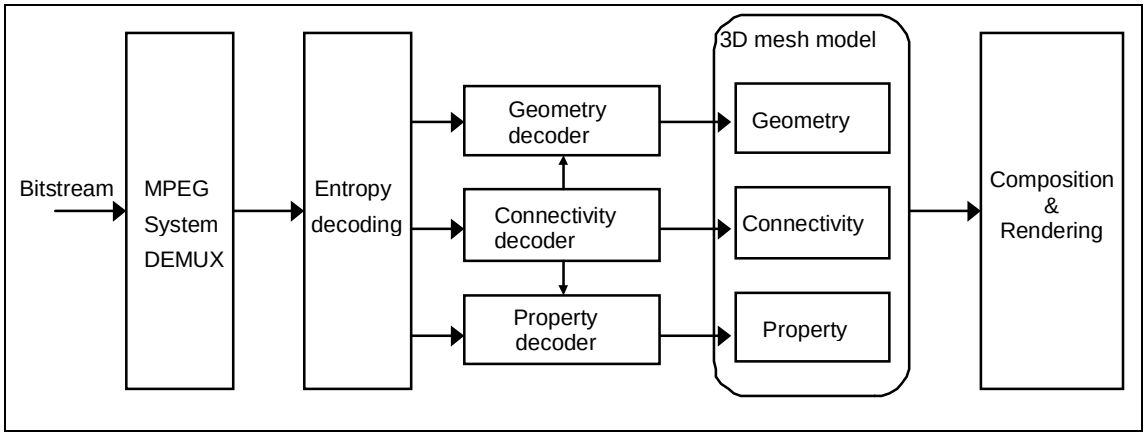
Encoder and Decoder Block Diagrams

High level block diagrams of a general 3D polygonal model encoder and decoder are shown in Figure V2 - 4. They consist of a 3D mesh connectivity (de)coder, geometry (de)coder, property (de)coder, and entropy (de)coding blocks. Connectivity, vertex position, and property information are extracted from 3D mesh model described in VRML or MPEG-4 BIFS format. The connectivity (de)coder is used for an efficient representation of the association between each face and its sustaining vertices. The geometry (de)coder is used for a lossy or lossless compression of vertex coordinates. The property (de)coder is used for a lossy or lossless compression of colour, normal, and texture coordinate data.



iTeh STANDARD PREVIEW
(standards.iteh.ai)
ISO/IEC 14496-2:1999/Amd 1:2000
https://standards.iteh.ai/catalog/standards/sist/bbc41711-1b4b-4f5e-91b8-f93388d15401/iso-iec-14496-2-1999-amd-1-2000

(a) 3D mesh encoder



(b) 3D mesh decoder

Figure V2 - 4 -- General block diagram of the 3D mesh compres

Overview of the object based nonscalable syntax

The coded representation defined in the non-scalable syntax achieves a high compression ratio while preserving good image quality. Further, when access to individual objects is desired, the shape of objects also needs to be