WO0034912A1: METHOD AND SYSTEM FOR PREDICTIVE ENCODING OF ARRAYS OF DATA

Video stream encoding method for storage and playback of movies or games in virtual reality by quantifying selected and predicted array value of image data

WO World Intellectual Property Organization (WIPO) Patent
A1 Publ.of the Int.Appl. with Int.search report

Julien, Signes M.; Rennes, France

France Telecom and Telediffusion de Franc, Paris, France

2000-06-15 / 1999-11-30

WO1999US0026032

1998-12-04 US1998000205191

A method and system for uniformly encoding arrays of values in a video stream. Using a data type of an element to be encoded, a predictive value is quantized and encoded as part of a video stream. The resulting video stream encodes rotations, normals, and vectors.

SPIVAK, Marvin, J.

ette date	Code	Description (remarks) List all possible codes for WO
2001-10-11	**REG** DE 8642 -	Reference to national code Impact abolished for de - i.e. pct appl. not ent. german phase
2001-06-01	**ENP**	Entry into the national phase in: (JP XX0000000587300)
2001-05-23	**ENP**	Entry into the national phase in: (CA XX0000002352292)
2000-09-08	**DFPE**	Request for preliminary examination filed prior to expiration of 19th month from priority date
2000-08-09	**121**	EP: the epo has been informed by wipo that ep was designated in this application
2000-06-15	**AK** +	Designated states cited in a published application with search report (AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CR CU CZ DE DK DM EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG US UZ VN YU ZA ZW)
2000-06-15	**AL** +	Designated countries for regional patents cited in a published application with search report (GH GM KE LS MW SD SL SZ TZ UG ZW AM AZ BY KG KZ MD RU TJ TM AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG)
2000-03-09	**ENP**	Entry into the national phase in: (AU XX0000000017132)

AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CR CU CZ DE DK DM EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG US UZ VN YU ZA ZW, **European patent:** AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE, **OAPI patent:** BF BJ CF CG CI CM GA GN GW ML MR NE

SN TD TG, **ARIPO patent:** GH GM KE LS MW SD SL SZ TZ UG ZW, **Eurasian patent:** AM AZ BY KG KZ MD RU TJ TM

15 familiy members

Publication	Pub. Date	Filed	Title
WO0034912A1	2000-06-15	1999-11-30	METHOD AND SYSTEM FOR PREDICTIVE ENCODING OF ARRAYS OF DATA
US6075901	2000-06-13	1998-12-04	Method and system for predictive encoding of arrays of data
NO20012723A0	2001-06-01	2001-06-01	FREMGANGSMAATE OG SYSTEM FOR PREDIKTIV KODING AV DATATABELLER
NO20012723A	2001-08-01	2001-06-01	FREMGANGSMAATE OG SYSTEM FOR FORUTSIGENDE KODING AV DATATABELLER
JP2002532923T2	2002-10-02	1999-11-30	
IL0143438A0	2002-04-21	1999-11-30	METHOD AND SYSTEM FOR PREDICTIVE ENCODING OF ARRAYS OF DATA
EP1138012A4	2005-01-19	1999-11-30	METHOD AND SYSTEM FOR PREDICTIVE ENCODING OF ARRAYS OF DATA
EP1138012A1	2001-10-04	1999-11-30	METHOD AND SYSTEM FOR PREDICTIVE ENCODING OF ARRAYS OF DATA
CN1329733T	2002-01-02	1999-11-30	Method and system for predictive encoding of arrays of data
CN1329733A	2002-01-02	1999-11-30	Method and system for predictive encoding of arrays of data
CN1124563C	2003-10-15	1999-11-30	Method and system for predictive encoding of arrays of data
CA2352292AA	2000-06-15	1999-11-30	METHOD AND SYSTEM FOR PREDICTIVE ENCODING OF ARRAYS OF DATA
BR9915947A	2001-08-21	1999-11-30	METODO E SISTEMA PARA CODIFICACAO DE PREDICAO DE ARRANJOS DE DADOS
AU0753876B2	2002-10-31	1999-11-30	METHOD AND SYSTEM FOR PREDICTIVE ENCODING OF ARRAYS OF DATA
AU0017132A5	2000-06-26	1999-11-30	METHOD AND SYSTEM FOR PREDICTIVE ENCODING OF ARRAYS OF DATA

TITLE OF THE INVENTIONMETHOD AND SYSTEM FOR PREDICTIVE ENCODING OFARRAYS OF DATACROSS-REFERENCE TO RELATED APPLICATIONS

The present application is related to the co-pending application entitled"Method and System for Encoding Rotations and Normals in 3D Generated Scenes,"serial number 09/205,190, attorney docket number 2167 2, filed on even dateherewith, also naming Julien Signes and Olivier Ondet as inventors. The contents ofthat co-pending application are incorporated herein by reference.

The present invention relates to the encoding of computer-generated images,and more particularly to a unified coding method for arrays of data, independent ofthe function of the data. Accordingly, arrays of position data, rotations, and normalsare encoded consistently.

Discussion of the BackarounThe phrase "computer-generated images" encompasses an expanding area ofvideo technology. Originally the term was often equated with simple text images or2D images; however, the phrase now encompasses any type of digitally encodedvideo stream. The Motion Pictures Expert Group (MPEG) was formed to investigatethe technologies required for the encoding and decoding of image streams. Thesu@sSWtT- (RULE 26)additional MPEG standards: MPEG-2 and MPEG MPEG-4 is a standard that is "inprogress" and forms the basis for the present invention. The final committee drafts areISO/IEC FCD 14496-1 MPEG-4 Systems and -2 MPEG-4 Visual, the contents of thefinal committee drafts are incorporated herein by reference.

The draft standard departs from the single stream-based model of video andchanges the focus to a series of streams that act in concert. One portion of thestandard is the Binary Format for Scenes (also known as "BIFS"). This format allowsthe description of 3D objects and their motion and provides the ability for greaterinteraction with that portion of the video stream. The draft standard for MPEG-4BIFS proposes to encode multiple data fields by a linear quantization process, whichdoes not account for the correlation within those multiple fields. Thus, MPEG-4BIFS does not describe predictive encoding of arrays.

Another proposed graphics format is the Compressed Binary Fornnat forVirtual Reality Markup Language (hereinafter "CBF VRML") that includes anencoding method that is applicable only to a few fields of a few nodes and for alimited subset of data types. The fourth and fifth drafts of the CBF VRML standard,dated August 22, 1997 and October 15, 1997, respectively. are available fromwww.vrml.org and are incorporated herein by reference. Likewise, the RevisedRequirements Statement for CBF VRML97, dated April 3, 1998, is also incorporatedherein by reference.

Yet another graphics format is the Tag Image File format (TIFF). Thecontents of version 6.0 of the TIFF specification are incorporated herein by reference.

Each entry in a TIFF file is stored as four parts: (1) a tag, (2) a type identifier, (3) acount, and (4) one or more values. Page 16 of the version 6.0 specification states that2SUBSTITUTE SHEET (RULE 26)single value." Moreover. although page 16 also states that there may be multipleimages per file. the specification is designed for independently coded 2D imagesrather than streaming 3D images.

SUMMARY OF THE INVENTION

It is an object of the present invention to address the coding inefficiencies ofknown MPEG-4 BIFS data streams.

It is a further object of the present invention to provide a method and systemfor encoding arrays of data. representing a scene or object(s). using a temporalpredictive encoding process.

This and other objects of the present invention are addressed by one or more of(1) a computer-implemented method for uniformly encoding arrays of values, (2) asystem for uniforrnly encoding arrays of values, and (3) a computer program productfor uniforrnly encoding arrays of values. Such a system is applicable to improvedstorage and playback of games, virtual reality environments, and movies. Moreover,based on the improved efficiency, video streams encoded according to the presentinvention may be plaved back over lower bandwidth communications links than lessefficiently encoded streams.

BRIEF DECSRIPTION OF THE DRAWINGSA more complete appreciation of the invention and many of the attendantadvantages thereof will become readily apparent with reference to the followingdetailed description, particularly when considered in conjunction xvith theaccompanying drawings, in which.

3.UBSTITUTE SHEET (RULE 26)and/or decoding services of the present invention-,Figure 2 is a schematic illustration of an encoder/decoder system. andFigure 3A is an illustration of an original bit-length used to encode a rotationin 128 bits;Figure 3B is an illustration of the result of using a known quantization andencoding method that uses 8-bit quantization to encode a series of rotations in 27 bitseach;andFigure 3C is an illustration of the result of using the quantization andpredictive encoding method according to the present invention - i.e., using 8-bitquantization and 4-bit predictive encoding (1) to encode a first rotation in 27 bits and(2) to encode a series of subsequent rotations in 12 bits each.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring now to the drawings, wherein like reference numerals designateidentical or corresponding parts throughout the several views, Figure I is a schematicillustration of a computer system for uniformly encoding and/or decoding arrays ofvalues. A computer I 00 implements the method of the present invention, wherein thecomputer housing 102 houses a motherboard. 104 which contains a CPU 106, memory108 (e.g., DRAM, ROM, EPROM, EEPROM, SRAM, SDRAM. and Flash RAM),and other optional special purpose logic devices (e.g., ASICs) or configurable logicdevices (e.g., GAL and reprogrammable FPGA). The computer I 00 also includesplural input devices, (e. a keyboard 122 and mouse 124), and a display card I IO forcontrolling monitor 120. In addition, the computer system I 00 further includes afloppy disk drive I 1 4. other removable media devices (e.g., compact disc 1 19, tape,4SUBSTITUTE SHEET (RULE 26)Gand removable matneto-optical media (not shown)): and a hard disk I I or otherfixed, high density media drives, connected using an appropriate device bus (e.g., aSCSI bus, an Enhanced IDE bus. or a Ultra DMA bus). Also connected to the samedevice bus or another device bus, the computer I 00 may additionally include acompact disc reader I 1 8. a compact disc reader/writer unit (not shown) or a compactdisc (not shown). Although compact disc II 9 is shown in a CD caddy, thecompact disc 1 19 can be inserted directly into CD-ROM drives that do not requirecaddies. In addition. a printer (not shown) also provides printed listings of encodedand/or decoded arrays.

As stated above, the system includes at least one computer readable medium.

Examples of computer readable media are compact discs I 1 9, hard disks 1 12, floppydisks, tape, magneto-optical disks, PROMs (EPROM, EEPROM, Flash EPROM),DRAM, SRAM, SDRAM, etc. Stored on any one or on a combination of computerreadable media, the present invention includes software for controlling both thehardware of the computer I 00 and for enabling the computer I 00 to interact with ahuman user. Such software may include, but is not limited to. device drivers,operating systems and user applications, such as development tools. Such computerreadable media further includes the computer program product of the presentinvention for uniformly encoding and/or decoding arrays in scene descriptions and/oranimations. The computer code mechanisms/devices of the present invention can beany interpreted or executable code mechanism, including but not limited to scripts,interpreters, dynamic link libraries, Java classes. and complete executable programs.

As shown in Figure 2. an authoring tool 200 is used in the first step of creatinga scene 240. Once the authoring tool 200 has created a series of images, the series ofimages are sent to a quantizer 205. After quantization. the quantized images are sent5SUBSTITUTE SHEET (RULE 26)to more compactly represent the contents of the later frame through differentialencoding. The predictor may include linear or non-linear (i.e., multiple order)interpolation. The differences between frames are encoded by the entropyencoder/decoder 215. Encoder/decoder 215 mav be anv one of a number of entropysystems (e.g., an adaptive arithmetic encoder). The results of the encoder/decoder215 are formatted by the embedding protocol 220 and either transmitted or storedusing a transmitter or storage device 225.

To decode the formatted, stored/transmitted images, the encoding process inundone" in reverse. First, the formatting is decoded by the embedding protocol 220to produce the original entropy encoded data. The entropy-encoded data are fed to adecoder 230 that includes a compensator. The compensator utilizes earlier frames andthe differential encodings to produce the frames of quantized data. The quantizeddata is sent to an inverse quantizer 235 which inverse quantizes the data into thescene. Due to the number of bits used in the quantization process, the scene may notbe a perfect representation of the original scene authored, but, by using the presentinvention, arrays are more efficiently encoded so that additional bits may be used forother purposes (e.g., higher resolution quantization).

For the purpose of clarity, authoring is described hereinafter as beingperformed to create a series of animation parameters, v. An arbitrary parameter, vi, isthe i" parameter in the series and may itself contain an array of real values. Duringquantization, the parameters, vi are converted to quantized parameters, vqi. Oneexample of a quantization method is linear uniform quantization as is described indescribed in MPEG-4 BIFS version 1.

6.UBSTITUTE SHEET (RULE 26)hereinafter in which prediction of values is extended to all values types. One or moreprediction functions are defined for each data type so that values of that data type canbe predicted. In an alternate embodiment, one prediction function (e.g.. addition) isused for all data types. As a result of prediction. during decoding. generally fewerbits are needed to recreate the intended scene.

The coding method described hereinafter is a generalized and uniformpredictive coding method that applies to all types of arrays of data in a scene. Thus,the method is applicable to intra-frame encodings (i.e., I-mode) and predictiveencodings (i.e., P-mode). In an alternate embodiment. in P-mode. a variable lengthencoderisusedtofurtherreducethedatasize. AsshowninFigures')A-')C,whenapplied to rotations, the encoding of data can be accomplished in fewer bits thanwithout quantization, and can be accomplished with even fewer bits yet whencombined with predictive encoding.

Additional types of data also can be addressed. Any 2D or 3D parameter, v,can be seen as a vector of real values of dimension D given by.

v = (v [0], v[I v[D- I]).

The corresponding quantized value of this parameter is an array of integers ofdimension D' given by.

q= (q[O]@ q[I q[D'- I]),where the dimension D' is not necessarily equal to D. Vectors, unit vectors, andIintegers comprise three specific categories of parameters that are encoded accordingto the predictive, quantized encoding of the present invention.

7.UBSTITUTE SHEET (RULE 26)and one-dimensional vectors. Three-dimensional vectors are used to encode values oftypes: Position3l), Size 3D. and Texture Coordinate. Two-dimensional vectors areused to encode values of types: Position2D. Size 21), and Texture Coordinate. One-dimensional vectors are floats that encode any single, non-integer value. That is, afloating point array of length I is used to encode values of types Angle, Scale,Interpolator keys. and BoundFloat. Colors are also a special form of floating pointthree-dimensional vectors with the red, green and blue components each beingrepresented on a scale of 0.0 - I.O. Based on the MPEG-4 BIFS v I specification,vectors are used for categories where QuantTypes and AnimationTypes are 1, 2, 4, 5,6, 7, 8, 1 1 and 12.

As a result of the special properties of unit vectors, unit vectors are separatelyaddressed and are used for normals and rotations (QuantTypes and AnimationTypes 9and 10). Additional details of the use of rotations and normals are disclosed in theabove-identified co-pending patent application.

The last category, integers. is also subject to predictive, quantized, encodings.

Based on the MPEG4 BIFS v I specifications, this category matches QuantTypes andAnimationTypes 3, 13 and 14. For reference, a more complete description of thecategories are set forth below in Tables I and 2.

8.UBSTITUTE SHEET (RULE 26)Table 1: QuantTypes (Quantization Categories)Category Description0 NoneI 3D Position2 2D positions3 drawOrder4 SFColor5 Texture Coordinate6 Angle7 Scale8 Interpolator keys9 NormalsI 0 RotationsI I Object Size 3D12 Object Size 2D13 Linear Scalar Quantization14

Coordindex15 ReservedTable 2: AnirnTypesCategory Description0 NoneI Position 3D2 Positions 2D3 Reserved4 Color5 Reserved6 Angle7 Float8 BoundFloat9 Normais10 Rotation11 Size 3D12 Size 2D13 Integer14 Reserved15 ReservedThe Quantization Inverse Quantization process is specified by several parameters.

These parameters depend on the general category of the coded parameter. Forvectors, the Quantizer / Inverse Quantizer should be supplied with.

9.UBSTITUTE SHEET (RULE 26)array of floating point numbers ("floats") of the same dimension as thedimension D of the parameter.

QuantMax: The upper bounds of quantization for the vector - coded as anarray of floats of the same dimension as the dimension D of the parameter.

QuantNbBits: The number of bits used in the vector quantization.

For certain categories (such as Colors or naturally Bound Floats). the QuantMin andQuantMax value may be obtained from the default (finite) bounds of the parameter. Inan alternate embodiment, a quantization step is specified instead of QuantMax; suchthat.

QuantStep = (QuantMax-QuantMin)/((2^(QuantNbBits))-1)Similarly, for unit vectors, the encoder/decoder is only supplied with oneencoding parameter, QuantNbBits - i.e., the number of bits used in the vectorquantization. For consistency, the floats QuantMin and QuantMax are defined as 0and 1, respectively.

For integers, the Quantizer / Inverse Quantizer is supplied with.

QuantMin: The lower bounds of quantization for the vector - coded as an integer.

QuantNbBits: The number of bits used in the vector quantization.

The elementary quantization process quantizes real values according to threeparameters.

QuantMin : The minimum quantization valueQuantMax: The maximum quantization value10SUBSTITUTE SHEET (RULE 26)A quantization step then can be defined according to.

QuantNbBits <= 0 QuantStep has no real meaning. and the quantized value issimply set to 0.

QuantNbBits > 0 OuantAllay - OuantMinQuantStep = -2 (humuNbIhisThe elementary quantization process is then defined by the following function(in the C programming language style).

Int quantize (float QuantMin, float QuantMay., float v, int QuantNbBits)returns the rounded value of (v-QuantMin)/QuantStepAccordingly, the elementary inverse quantization process is defined by the followingfunction.

float invQuantize (float QuantMin,float QuantMax, float vq,int QuantNbBits)retums (QuantMin + vq*QuantStep)The quantization process described in MPEG 4 BIFS v I utilises quantization andinverse quantization functions such as those described above and is applicable to thepresent invention. Other quantization and inverse quantization functions may be usedin alternate embodiments as long as the quantization process supports a compensationprocess as well.

As part of the compensation process, the following parameters are defined.

CornpMin: An array of integers that define the lower bound of thecompensation vector., vDelta.

ComT)NbBits: An integer that defines the number of bits used to code thecomponents of the compensation vector.

11.UBSTITUTE SHEET (RULE 26)encoding of the components of the compensation vector vDelta in the binary file.

Each component vDelta[i] is translated by CompMin[i] and coded usingCompNbBits. The actual coded value is then: vDelta[i] - CompMin[i].

The general prediction process will now be described in terms of successivequantized values vq 1, vq2 and the delta value between them vDelta.

The encoding process will first be described. For vectors and integers, thecalculation is straightforward and given by.

vDelta[i] = vq2[i] - vq I [i].

For unit vectors, a more detailed calculation is utilized (and is discussed ingreater detail in the above-identified co-pending application). As discussed above, ifQuantNbBits=O, then there is no actual coding since no bits are available for use. Inthat case, all values are assumed to be zero and need not be written to a video stream.

However, when QuantNbBits>O, the coding process is performed as follows.

First the variable inv is initially set to one (but this variable may change during theprocess). Then, the number, N, of reduced components is determined, where N=2 fornormals and N=3 for rotations. Then, the differential orientations and directionsbetween vq I and vq2 are computed based on.

dOri = (ori2-ori I) mod (N+ 1)dDir = dirl *dir2scale =2 QuantNbffits- IThen there are two cases to consider.

12.UBSTITUTE SHEET (RULE 26)dOri 0 vDelta is defined byvDelta[i] =vq2[i]-vqlfi]dOri 0 Let dist = vq2[N-dOrl] + vq I [dOri- Iif dist<O inv = -IvDelta is then defined byO<i<dO vDelta[il=inv*vq2[i-dOri-l]-vq[i]i=dOri-I vDelta[i]=inv*2*scale-distdori<i< vDelta[i]=Inv*vq2[1-dOri]-vqlfi]The variable J nve r s e calculated according to i nve r s e = inv dDir. As a resultof the marginal non-injectivity of the quantization process, vq2', the compensation ofvq I by vDelta may not yield exactly vq2. However, vq2 and vq2' will alwaysrepresent nearly the same normal or rotation.

Each of the three basic categories of values (i.e., integers, vectors, and unitvectors) can be predicted. Integers are predicted using simple subtraction. Vectorsutilize a quantized-component-based subtraction for prediction. Unit vectors use aspecialized subtraction based on the topological structure of the mapping cube.

Having predicted values for each data type. compensation is used duringdecoding to correct for differences between the predicted and actual values. Let vq Ibe the previously computed quantized parameter, vDelta the delta value and vq2 bethe quantized value resulting from the addition. It then follows that a quantized valuevq contains an array of integers vq[]. AdditionalINI, for normals and rotations, vq I1 3SUBSTITUTE SHEET (RULE 26)vDelta contains an array of integers vDelta[]. Additionally. for normals. it contains aninteger inverse whose value is -I or 1. The result vq2 is then computed depending onthe type of value being calculated. For vectors and integers, the components of vq2are given by.

vq2 [i] = vq I [i] + vDelta[i]For unit vectors. however, additional processing is performed because of the unit cubestructure.

As described above, the following method is performed. First a component-by-component addition is performed, and the results are stored in a temporary arraygiven by.

vqTemp[i] = vq I [i] + vDelta[i].

If QuantNbBits is zero or 1, scale is set to 1/2, otherwise, scale is set to 2Q,anINbBits-1As described in greater detail in the co-pending application, the use ofQuantNbBits=O is essentially equivalent to the QuantNbBits=l for the presentinvention. By using the direction and orientation information alone (i.e., whenQuantNbBits=O), movement from one face to every other can be described.

However, in the more common cases described below, QuantNbBits >=2.

Then, N, the number of reduced components, is set (where N=2 for normals and N=3for rotations). Once those initial steps are performed. additional processing isperformed as described below, depending on which of the three conditions are met.

14.UBSTITUTE SHEET (RULE 26)index 1. vq2[i] vqTemp[i]ivjemp[i) < scale orientation2= orientation Idirection2 =direction I * inverse(2) There is one vq2 is resealed as if gliding on the faces of the mapping cube.

and onlv one Let inv I if vqTemp[k]>=O and -1 elseindex k such that Let dOri = k+11vqTemp[kj > scale The components of vq2 are computed as follows0 < i < N -dOri vq2[i] = inv*vqTemp[(I+dOrll) mod N]i = N -dOri vq2[i] = inv*1.*scale - vqTemp[dOri-ljN - dOri < i < N vq2[i] = inv*vqTemp[(i+dOrl- 1) mod N]orientation2 (orientationl + dOri) mod (N+I)direction2 = direetionl * inverse * invThere are1 several indices k The result is undefinedsuch thatlvqTemp[kl > scaleFor each of the three type of values (i.e., integers. vectors. and unit vectors),the compensation process reverses the predictive process. Thus. for integers a simpleaddition process undoes the predictive subtraction. Likewise, for vectors a quantized-component-based addition is used. Lastly, a topologically coherent addition is usedfor unit vectors, potentially requiring computation of a new orientation and direction.

As described Nvith reference to Figure 2, once the encoder 215 has completedentropy encoding, the resulting values must be stored, along with header information,in the format specified by the protocol. Thus. one embodiment utilizes the following1 5SUBSTITUTE SHEET (RULE 26)given a parameter category. that describes which QuantMin, QuantMax.

QuantNbBits. CompMin. CompNbBits are to be used, according to the previouslydescribed process. Then a header for each array is coded. The header includes.

(1) a length of the field. and(2) a policy for I-mode (i.e., simple quantization) and P-mode given one of thefollowing three categories: (a) a single Intra value is followed by onlyPredictive coded values, (b) an I-value is provided for each n p-values(thereby limiting the length of the propagation of an error), and (c) a bitprecedes (or follows) each value to signify whether the value is an I-value or aP-value.

Then the quantization and compensation parameters for coding predictivevalues are added to the stream. Intra quantization parameters (QuantMin, QuantMax,QuantNbBits) can be retrieved in BITS by the corresponding QuantizationParameterin the scope of the encoded field. The compensation parameters (CompMin,CompNbBits) are added directly to the stream.

Lastly, the array of quantized values is coded, following the Intra/predictivepolicy. I-values are coded as quantized values. P-values are encoded through anarithmetic adaptive encoder whose session is opened between any two Intras.

The bitstrearn syntax below is used to code the array of values. The bitstreamsyntax uses conventions of the MSDL language as specified by MPEG-4 FDIS14496 1 6SUBSTITUTE SHEET (RULE 26). 1 Syntaxclass Array0fData(ArrayHeader header;Array0fValues values;1.2 SemanticThe arrav of data is composed of a Header, and an ArrayOfValues2.0 ArrayHeader2. 1 Syntaxclass ArrayHeaderfint(5) NbBits;int(NbBits) numberOfFields;bit(2) Ippoiicy;if (IPP0iiCY == 1)int(NbBits) intraInterval;InitialArrayQP qp;2.2 SemanticThe arrav header contains first information to specify the number of fields(NbBits is the number of bits used to code the numberOfFields). Then theIntra/Predictive policy (IPPolicy) is specified as follows.

If is is 0Only one Intra value at the beginning and then only predictive codedvaluesIf is is IAn Intra every given number of predictive valuesIf is is 2An bit for each value to determine whether the value is an Intra orpredictive valueLastly, the InitialArrayQP is coded.

1.7SUBSTITUTE SHEET (RULE 26). 1 Syntaxciass !--'ziia1ArrayQP(switc'- (Ippolicy)case 1 .

int NbBits;int(NbBits) intraIntervai;// no breakcase 0case 2-int(5) CompNbBits;for (int i=O;i<NbComp(quantType))int(QuantNbBits+1) CompMin[ijno breakcase 3.

break;3.2 SemanticIf IPPolicy is 1. the size of the interval between two intras is first specified.

Independent of the IPPolicy, the number of Bits used in Predictive mode CompNbBitsand the CompMins are coded. (The function NbComp(is a function that returns thenumber of components of the quantizing bounds, and depends on the object. Forinstance it returns 3 for 3)D positions, 2 for 2D positions, and 3 for rotations.)4.0 ArrayQP4.1 Syntaxciass ArrayQP(switch (intraMode)case 1 .

int NbBits;int(NbBits) intraIntervai;// no breakcase 0case 2boolean(l) hasCompNbBitsif (hasComQNbBits)int(5) CompNbBits;boolean(l) hasCompMinif (hasCompMin)for (int i=O;i<NbComp(quantType))18SUBSTITUTE SHEET (RULE 26)int(QuantUtEits+."; ComoMin[i]no breakcase 3.

break;4.2 SemanticArrayQP fullfills tile same purpose as InitialArrayQP, but in this case, theparameters are optionnaly set. If they are not set in the stream, they are set by default.

in reference to the InitialArrayQP or the latest received value of the parameter.

If IPPolicy is 1. the size of the interval between two intras is first specified. Inam,, case. the number of Bits used in Predictive mode (CompNhBits) and theCompMins are coded. The function NbComp(is a function that returns the numberof components of the quantizing bounds, is simply depends on the object. For instanceis is 3 for 31) positions, 2 for 2D positons. 3 for rotations, etc.

5.0 ArrayMalues5.1 Syntaxclass ArrayOfValues fArrayIValue vaiue[O];for (int i=1; i < nurnberOfFields;i++)Switch (intraMode) fcase 0.

ArrayPVaiue value;break;case 1.

bit (1) isIntra;if (isIntra) fbit(l) hasQP;if (hasQP)ArrayQP qp;ArrayIVaiue value;else (ArrayPvalue value;break;case 2.

if ((i 4ntraIntervai) 0)bit(l) hasQP;if (hasQP)ArrayQP qp;19SUBSTITUTE SHEET (RULE 26)ArrayIValue value;else fArrayPvaiue value;break;5.2 SemanticThe array of values first codes a first intra value. and then according to theIPPolicy, codes Intra and Predictive values. In P only mode, no more intra values arecoded. In the second mode, a bit decides of the P or I mode at each value. In thatcase, a QP can be sent for Intra values. If a QP is sent, the statistics of the arithmeticencoder are reset. In the third mode, an Intra is sent every intraInterval values.

6.0 ArraylValue6. 1 Syntaxclass AnimationIvaiue(FieldData field) fswitch (field.animType)case 9: Normalint(l) directioncase 10: Rotationint(2) orientationbreak;default.

break;for (j=O;j<getNbComp(field);j++)int(QuantNbBits) vq[j];6.2 SemanticThe ArrayIValue represents the quantized intra value of a field. The value iscoded following the quantization process described in the quantization section, andaccording to the type of the field. For normals the direction and orientation valuesspecified in the quantization process are first coded. For rotations only the orientationvalue is coded. If the bit representing the direction is 0. the normal's direction is set to1, if the bit is 1. the non-nal's direction is set to The value of the orientation is20SUBSTITUTE SHEET (RULE 26)field's value are then coded as a sequence of unsigned integers using the number ofbits specified in the field data structure. The decoding process in intra modecomputes the animation values by applying the inverse quantization process.

7.0 ArrayPValue7. 1 Syntaxclass ArrayPValue(FieldData field)switch (fieid.animType)case 9: // Normalint(l) inversebreak;default.

break;for (j=O;j<getNbComp(field);j++)int(aacNbBits) vqDelta[j];7.2 SemanticThe ArrayPValue represents the difference between the previously received quantizedvalue and the current quantized value of a field. The value is coded using thecompensation process as described above.

The values are decoded from the adaptive arithmetic coder bitstream with theprocedure v-aac = aa-decode(model). The model is updated with the proceduremodel-update(model, v_aac). For normals the inverse value is decoded through theadaptive arithmetic coder with a uniform, non-updated model. If the bit is 0, theninverse is set to 1, the bit it is 1, inverse is set to The compensation valuesvqDelta[i] arethendecodedonebyone.Letvq(t-l)thequantizedvaluedecodedat the previous frame and v-aac(t) the value decoded by the frame's AdaptiveArithmetic Decoder at instant t with the field's models. The value a time t is obtainedfrom the previous value as follows .

2.1SUBSTITUTE SHEET (RULE 26)vq(t) = AddDelta(vq(t- 1), vDelta(t))v(t) = InvQuant(vq(t))The field's models are updated each time a value is decoded through the adaptivearithmetic coder. If the animType is I (Position3D) or 2 (Position2D), eachcomponent of the field's value is using its own model and offset PMin[i]. In all othercases the same model and offset PMin[O] is used for all the components.

As shown in Table 3 below, the compression of arrays of values according tothe present invention can provide significant compression of scenes as compared toVRML ASCII files representing substantially similar video streams.

Table 3File Vrml Scene BIFS-Scene BIFS-SceneFile size (kB) No Compensation with Compensation(estimation)File size (kB) ratio File size ratio(kB)Intro 347 45 7.7 25 13.T-Prolog 621 1 81 7.7 48 12.9Moviel 968 103 9.4 65 14.9Finale 412 58 7.1 35 11.8Movie2 117 13.2 8.9 9 13.0Skeleton 34 2.9 11.7Floops 1324 95 13.9 65 20.3Fishswim 37 4.4 8.4Store 155 14 11.1 9 17 Tree 80 4.9 16.3Channe110 100 2.9 34.5Meteol 1 4.0 0.285 14.0Flight 1 62 4.7 13.2As would be evident to one of ordinary skill in the art, numerousmodifications can be made based upon the teachings of the present invention.

Accordingly, the examples and embodiments described above are not intended to belimiting but rather exemplary. Only the appended claims define the scope ofprotection to which applicants are entitled.

22.UBSTITUTE SHEET (RULE 26) †

CLAIMS

1.A computer-implemented method of uniformly encoding a video stream.the method comprising the steps of-(a) determining, for substantially each data type used in a video stream, acorresponding, type-specific quantization parameters;(b) selecting a first value from a first array of one of the data types used in thevideo stream;(c) quantizing the first value using the corresponding, type-specificquantization parameters;(d) selecting a second value from the first array of one of the data types usedin the video stream;(e) determining a predicted value from the first value and the second valueusing the corresponding, type-specific quantization parameters;(f) quantizing the predicted value using the corresponding, type-specificquantization parameters;(g) encoding the quantized first value and the quantized predicted value intothe video stream; and(h) repeating steps (b)-(g) for substantially all arrays to be encoded in thevideo stream.

2.The method according to claim 1, further comprising the step of encodingheader information into the video stream.

3.The method according to claim 1, further comprising the step of decodingthe video stream encoded in steps (g) and (h).

23SUBSTITUTE SHEET (RULE 26)1201 10NZ100 lo,@CD121 k,@ 11 4 106124 ------ ------------------- ----------------A_j@f I I f A 1 7 1 A f-T-r-T=%22FIGO 1240Authoring Tool Scene23Vi V205Quantizer Inverse Quantizerqi /210 q9i A 230Prediction Compensationq. qj., q'idi d9idi d9i215 IFEntropy coding/decoding220Embedding Protocol225Transmission, StorageFigure 22 / 3SUBSTITUTE SHEET (RULE 26)Figure 3AInitial data: 128 bits for each rotationFigure 3B27 bits 27 bits 27 bits 27 bitsFigure 3C27 bits 12 b 12 b 12 b3 / 3SUBSTITUTE SHEET (RULE 26)INTERNATIONAL SEARCH REPORT International application No.PCT/IJS99/26032A. CLASSIFICATION OF SUBJECT MATTERIPC(6) :GO6K 9/00US CL :Please See Extra SheetAccording to International Patent Classification (lPC) or to both national classification and IPCB. FIELDS SEARCHEDMinimum documentation searched (classification system followed by classification symbols)U.S. 382t2329 233, 234, 235, 236, 238, 239, 240o 241t 242, 243v 244, 248, 251Documentation searched other than minimum documentation to the extent that such documents are included in the fields searchedElectronic data base consulted during the international search (name of data base and, where practicable, search terms used)C. DOCUMENTS CONSIDERED TO @'L RELEVANTCategory* Citation of document, with indication, where appropriate, of the relevant passages Relevant to claim No.y US 59227,878 A (PURI et al) 13 July 1993, column 2, lines 48 1-3y US 41751,742 A (MEEKER) 14 June 1988, column 5, lines 1 1-3Further documents are listed in t:.. continuation of Box C. See patent family annex.Special categories of cited documents: IT' later document published after the international filing date or prioritydate and not in conflict with the application but cited to understand'Aa document defining the general state of the art which in not considered die principle or theory underlying the inventionto be of particular relevance,XI document of particular relevance; the claimed invention cannot be'El earlier document published on or after the international filing date considered novel or cannot be considered to involve an inventive step*L' document which may throw doubts on priority claim(s) or which is when the document is taken alonecited to establish the publication date r(another citation or other IVspecial reason (as specified) document c articular relevance; the claimed invention cannot beconsidered to involve an inventive stop when the document is

10.document referring to an oral disclosure, use. exhibition or other combined with one or more other such documents, such combinationMae= being obvious to a person &killed in the art,PI document published prior to the international filing date but later than o&n document member of the same patent familythe priority date claimedDate of the actual completion of the international search Date of mailing of the international search report12 FEBRUARY 2000 07 MAR 2000Name and mailing address of the ISA/US Authorized officerCommissioner of Patents and TrademarksBox PCT BIJAN TADAYONWashington, D.C. 20231Facsimile No. (703) 305-3230 Telephone No. (703) 3 7 2@@7@Form PCT/ISAt2lO (second sheetXJuly 1992)*WTERNATIONAL SEARCH REPORT International application No.PCT/IJS99t26032A. CLASSIFICATION OF SUBJECT MATTER:US CL :382/232, 233, 234, 235, 236, 238, 239, 240, 241, 242, 243, 244, 248, 251Form PCT/ISA/210 (extra shcctXJuly 1992)* †