(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 December 2015 (23.12.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/193147 A1

(51) International Patent Classification:
H04N 19/50 (2014.01) *H04N 19/463* (2014.01)
H04N 19/30 (2014.01) *H04N 19/186* (2014.01)

(21) International Application Number:
PCT/EP2015/062909

(22) International Filing Date:
10 June 2015 (10.06.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
14305948.3 19 June 2014 (19.06.2014) EP
14306167.9 18 July 2014 (18.07.2014) EP

(71) Applicant: **THOMSON LICENSING** [FR/FR]; 1-5 rue Jeanne d'Arc, 92130 Issy-les-Moulineaux (FR).

(72) Inventors: **BORDES, Philippe**; c/o Technicolor R&D France, 975, avenue des Champs Blancs, CS17616, F-35576 Cesson Sévigné Cedex (FR). **RACAPE, Fabien**; c/o Technicolor R&D France, 975, avenue des Champs Blancs, CS17616, F-35576 Cesson Sévigné Cedex (FR). **ANDRIVON, Pierre**; c/o Technicolor R&D France, 975, avenue des Champs Blancs, CS17616, F-35576 Cesson Sévigné Cedex (FR).

(74) Agents: **LORETTE, Anne** et al.; 1-5 rue Jeanne d'Arc, 92130 Issy-Les-Moulineaux (FR).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ,

[Continued on next page]

(54) Title: METHOD FOR DECODING A COLOR MAPPING TABLE AND CORRESPONDING DEVICE

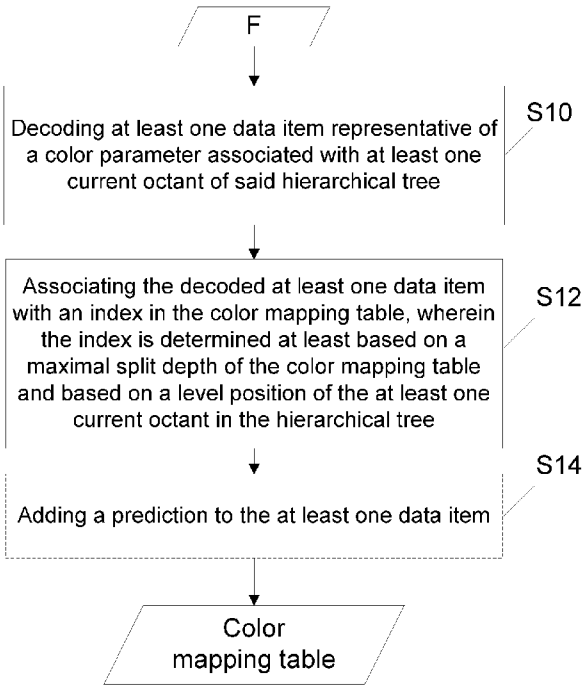


FIGURE 7

(57) Abstract: A method for decoding a color mapping table coded as a hierarchical tree of octants arranged in levels is disclosed. The method comprises for at least one current octant of said hierarchical tree: a) decoding (S10) at least one data item representative of a color parameter associated with said at least one current octant; b) associating (S12) said decoded at least one data item with an index in the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a level position of said at least one current octant in the hierarchical tree.

WO 2015/193147 A1

TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:

— *with international search report (Art. 21(3))*

METHOD FOR DECODING A COLOR MAPPING TABLE AND CORRESPONDING DEVICE

1. TECHNICAL FIELD

In the following, a method of decoding a color mapping table coded as a hierarchical tree of octants/nodes arranged in levels is disclosed. A corresponding decoding device is disclosed. Corresponding encoding method and encoding device are also disclosed. Such encoding/decoding methods are used in the field of Color Gamut Scalability (CGS) coding/decoding.

2. BACKGROUND ART

In Scalable Video Coding (e.g. SHVC as defined in the document entitled *Preliminary version of High efficiency video coding (HEVC) scalable extension Draft 6, JCTVC_Q1008_v3* published on June 7, 2014), in the case where the color spaces and/or the color gamuts of the Base Layer (BL) and of the Enhancement Layer (EL) are different and/or have been color graded differently, one can use a color mapping process to transform the color samples of the BL when performing the prediction of the EL samples from BL samples as depicted in **Figure 1** (inter-layer prediction process). This tool known as Color Gamut Scalability (CGS) or color mapping in SHVC. In Figure 1, the up-scaling is optional and is only applied in the case where BL pictures and EL pictures have different spatial resolutions.

In the current SHVC specification (JCTVC_Q1008_v3), the color mapping is performed using a color mapping table and more precisely a 3D LUT (LUT is an English acronym of Look-Up Table) containing color parameters. The 3D LUT is composed of a lattice of nodes/octants, each node/octant being associated with color parameters (e.g. 4 or 8 color vertices or 9 matrix coefficients with 3 additional offsets). Each node represents an octant, i.e. a portion of a 3D color space, e.g. YUV or RGB color spaces. EP application PCT/EP14/055333 "*Method for encoding and method for decoding a LUT and corresponding devices*" describes a method of encoding a color 3D LUT as a hierarchical tree of octants, wherein an octant may be represented by a node. The encoding method comprises the prediction of some color parameters from previously

reconstructed neighboring color parameters. The color 3D LUT is also called color mapping table.

In SHVC, the 3D LUT is described using a hierarchical tree of octants of maximal depth cm_octant_depth, where the final nodes/octants are further split into YPartNum leaves as depicted in **Figures 2** and **3**. In this case the splitting is done in one dimension only, namely the Y dimension (in the YUV color space). For each of these YPartNum leaves, the color parameters are encoded. The encoding/decoding of one color parameter comprises coding/decoding of residual values, e.g. res_y, res_u, res_v as defined in JCTVC_Q1008_v3. The residual values and the reconstructed/decoded color parameters may be stored in 3D arrays (e.g. res_y[yIdx][uIdx][vIdx], res_u[yIdx][uIdx][vIdx], res_v[yIdx][uIdx][vIdx] and LutY [yIdx][uIdx][vIdx], LutU [yIdx][uIdx][vIdx], LutV [yIdx][uIdx][vIdx] respectively). In the following, the word “decoded” and “reconstructed” can be used interchangeably.

The current coding syntax (as defined in JCTVC_Q1008_v3) thus makes it possible to recursively split the tree nodes into 8 octants as depicted in Figures 2 and 3. The splitting is signaled by the syntax element “split_octant_flag”. In figures 2 and 3, the value of split_octant_flag is indicated inside the tree leaves.

The corresponding coding syntax is depicted in the following table 1 extracted from sections F.7.3.5.1 and F.7.3.5.2 of the document entitled *Preliminary version of High efficiency video coding (HEVC) scalable extension Draft 6, JCTVC-Q1008-V3*.

color_mapping_table( ) {	**Descriptor**
cm_octant_depth	u(2)
cm_y_part_num_log2	u(2)
cm_input_luma_bit_depth_minus8	u(3)
cm_input_chroma_bit_depth_delta	se(v)
cm_output_luma_bit_depth_minus8	u(3)
cm_output_chroma_bit_depth_delta	se(v)
cm_res_quant_bits	u(2)
color_mapping_octants(0, 0, 0, 0, 1 << cm_octant_depth)	
}	

color_mapping_octants(depth, yIdx, uIdx, vIdx, length) {	**Descriptor**
if (depth < cm_octant_depth)	
split_octant_flag	u(1)
if (split_octant_flag)	
for(k = 0; k < 2; k++)	
for(m = 0; m < 2 ; m++)	
for(n = 0; n < 2; n++)	
color_mapping_octants(depth + 1, yIdx + YPartNum * k * length / 2, uIdx + m * length / 2, vIdx + n * length / 2, length / 2)	
else	
for(i = 0; i < YPartNum; i++)	
for(vertex = 0; vertex < 4; vertex++) {	
coded_vertex_flag[yIdx + i][uIdx][vIdx][vertex]	u(1)
if(coded_vertex_flag[yIdx + i][uIdx][vIdx][vertex]) {	
res_y[yIdx + i][uIdx][vIdx][vertex]	se(v)
res_u[yIdx + i][uIdx][vIdx][vertex]	se(v)
res_v[yIdx + i][uIdx][vIdx][vertex]	se(v)
}	
}	
}	

Table 1: Current Color LUT syntax.

The color mapping table semantics are defined in section F.7.4.6.

Figure 4 represents an example of a 3D LUT (Y and U dimensions are represented while the V dimension is not) obtained with cm_octant_depth=2, YPartNum=4, where the octant(y=0,u=0,v=0, depth=0) is split into 8 octants, in which first octant octant(y=0,u=0,v=0, depth=1) is split again into 8 octants but the second octant octant(y=1,u=0,v=0, depth=1) (depicted with a bold line) is not split again.

The solution as currently disclosed in JCTVC_Q1008_v3 has several drawbacks.

First, the reconstructed explicitly coded color parameters in JCTVC_Q1008_v3 are consecutively stored in the LUT (circle inside the octant delimited with the bold line). Therefore, the color space sampling of this octant is not optimal because not uniformly distributed inside the octant in the case where its depth is strictly inferior to cm_octant_depth and split_octant_flag=0. Second, in JCTVC_Q1008_v3, some of the residuals representative of color parameters may be indicated as “not explicitly encoded” (they are identified by a question mark in Figure 4). In this case, JCTVC_Q1008_v3 teaches to derive these “non-explicitly-encoded” color parameters by inferring they have zero residuals. Then, they are (recursively) derived from the last reconstructed neighboring (in the color table) color parameters in decoding order (i.e. previously decoded color parameters), as specified by the prediction process of the color parameters LUT values described in JCTVC_Q1008_v3 and as depicted by the arrows in the example of Figure 4. In this example, the relative distance of color parameters derivation is up to R/2 (with R = range of the component, typically 256 for a 8-bit full range signal) as depicted in Figure 4. Specifically, the distance between a non-explicitly-encoded color parameter and the corresponding explicitly-encoded color parameter is up to R/2 (as depicted in the second row of the Figure 4).

3. BRIEF SUMMARY

A method for decoding a color mapping table coded as a hierarchical tree of octants arranged in levels is disclosed. The method comprises for at least one current octant of said hierarchical tree:

a) decoding at least one data item representative of a color parameter associated with said at least one current octant; and

b) associating said decoded at least one data item with an index in the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a level position of said at least one current octant in the hierarchical tree.

Specifically, the color parameters are almost uniformly arranged in the color mapping table. Consequently, a better color space sampling of the color mapping table is obtained in case of non-explicitly-encoded color parameters.

In addition, a lower distance derivation between explicitly-encoded color parameters and non-explicitly-encoded color parameters is used.

In a specific embodiment, associating said decoded at least one data item with an index of the color mapping table determined at least based on a maximal split depth of the color mapping table and based on a level position of said at least one current octant in the hierarchical tree comprises storing said at least one data item in an array at a location determined by said index.

According to a specific characteristic, the at least one data item is a residual of a color parameter.

Advantageously, the decoding method further comprises adding a prediction to said decoded at least one data item in order to obtain a decoded color parameter, wherein the prediction is obtained from at least one previously decoded color parameter.

Advantageously, the method for coding further comprises reconstructing a non-explicitly encoded color parameter at least from said decoded color parameter.

A decoding device for decoding a color mapping table coded as a hierarchical tree of octants arranged in levels is also disclosed that comprises a processor configured to:

a) decoding at least one data item representative of a color parameter associated with at least one current octant of said hierarchical tree;

b) associating said decoded at least one data item with an index in the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a level position of said at least one current octant in the hierarchical tree.

A decoding device for decoding a color mapping table coded as a hierarchical tree of octants arranged in levels is disclosed that said decoding device comprises:

- means for decoding at least one data item representative of a color parameter associated with at least one current octant of said hierarchical tree;
- means for associating said decoded at least one data item with an index in the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a level position of said at least one current octant in the hierarchical tree.

The device is configured to execute the steps of the method for decoding according to any of the embodiments and variants disclosed.

A computer program product comprising program code instructions to execute the steps of the decoding method according to any of the embodiments and variants disclosed when this program is executed on a computer.

A processor readable medium having stored therein instructions for causing a processor to perform at least the steps of the decoding method according to any of the embodiments and variants disclosed.

4. BRIEF DESCRIPTION OF THE DRAWINGS

- Figure 1 shows a Scalable Video Decoder using Color Gamut Scalability according to the prior art;
- Figure 2 depicts a hierarchical tree describing a LUT with cm_octant_depth=1 and YPartNum=4 according to the prior art;

- Figure 3 depicts a hierarchical tree describing a LUT with cm_octant_depth=2 and YPartNum=4 according to the prior art;
- Figure 4 represents an example of a 3D LUT (Y and U dimensions are represented while the V dimension is not) obtained with cm_octant_depth=2, YPartNum=4 according to the prior art;
- Figure 5 depicts a device for decoding a color mapping table coded as a hierarchical tree of nodes arranged in levels according to a specific and non-limitative embodiment;
- Figure 6 represents an exemplary architecture of the decoding device configured to decode a color mapping table from a bitstream according to an exemplary embodiment;
- Figure 7 represents a flowchart of a method for decoding a color mapping table coded as a hierarchical tree of nodes arranged in levels from a bitstream F according to a specific and non-limitative embodiment;
- Figure 8 represents an example of a 3D LUT (Y and U dimensions are represented while the V dimension is not) obtained with cm_octant_depth=2, YPartNum=4 according to a specific and non-limitative embodiment;
- Figure 9 depicts a flowchart of a method for decoding a color mapping table in two steps according to a specific and non-limitative embodiment; and
- Figure 10 represents an example of a 3D LUT (Y and U dimensions are represented while the V dimension is not) obtained with cm_octant_depth=2, YPartNum=4 according to a specific and non-limitative embodiment;
- Figure 11 depicts a device for encoding a color mapping table as a hierarchical tree of nodes arranged in levels according to a specific and non-limitative embodiment;
- Figure 12 represents an exemplary architecture of the encoding device configured to encode a color mapping table in a bitstream according to an exemplary embodiment;
- Figure 13 represents a flowchart of a method for encoding a color mapping table as a hierarchical tree of nodes arranged in levels in a bitstream F according to a specific and non-limitative embodiment;

- Figure 14 represents a flowchart of a method for determining a LUT representative of a color mapping transform between two frames;
- Figures 15 and 16 represent a 3D LUT (Y and U dimensions are represented while the V dimension is not) obtained with cm_octant_depth=2, YPartNum=4 according to a specific and non-limitative embodiment; and
- Figure 17 represent examples of a decoded 3D LUT according to a specific and non-limitative embodiment.

5. DETAILED DESCRIPTION

Figure 5 depicts a device 100 for decoding a color mapping table coded as a hierarchical tree of nodes arranged in levels according to a specific and non-limitative embodiment. In the following, the word “LUT” and “color mapping table” can be used interchangeably.

Each node represents an octant, i.e. a portion of a 3D color space, e.g. YUV or RGB color space. The decoding device 100 comprises an input 10 configured to receive a bitstream. The input 10 is linked to a decoding module 12 configured to decode at least one data item representative of at least one color parameter associated with at least one current node/octant of the hierarchical tree. The decoding module 12 is linked to a module 14 configured to associate the data item with an index of the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a level position of the current node/octant in the hierarchical tree. Exemplarily, the module 14 comprises a memory in which the data item is stored at a location determined by the index.

Optionally, the decoding device further comprises a module 16 configured to add a prediction data to the data item. The module 14 or the module 16 when present is linked to an output 18.

The color mapping table may be stored in a memory internal to the decoding device 100 or external to it. In a variant, the color mapping table is sent to a destination.

Figure 6 represents an exemplary architecture of the decoding device 100 configured to decode a color mapping table from a bitstream according to an exemplary

embodiment. The decoding device 100 comprises one or more processor(s) 110 , which is(are), for example, a CPU, a GPU and/or a DSP (English acronym of Digital Signal Processor), along with internal memory 120 (e.g. RAM, ROM, EPROM). The decoding device 100 comprises one or several Input/Output interface(s) 130 adapted to display output information and/or allow a user to enter commands and/or data (e.g. a keyboard, a mouse, a touchpad, a webcam); and a power source 140 which may be external to the decoding device 100. The device 100 may also comprise network interface(s) (not shown). The bitstream may be obtained from a source. According to different embodiments, the source belongs to a set comprising:

- a local memory, e.g. a video memory, a RAM, a flash memory, a hard disk ;
- a storage interface, e.g. an interface with a mass storage, a ROM, an optical disc or a magnetic support;
- a communication interface, e.g. a wireline interface (for example a bus interface, a wide area network interface, a local area network interface) or a wireless interface (such as a IEEE 802.11 interface or a Bluetooth interface); and
- an image capturing circuit (e.g. a sensor such as, for example, a CCD (or Charge-Coupled Device) or CMOS (or Complementary Metal-Oxide-Semiconductor)).

According to different embodiments , the decoded color mapping table may be sent to a destination. As an example, the decoded color mapping table is stored in a remote or in a local memory, e.g. a video memory or a RAM, a hard disk. In a variant, the decoded color mapping table is sent to a storage interface, e.g. an interface with a mass storage, a ROM, a flash memory, an optical disc or a magnetic support and/or transmitted over a communication interface, e.g. an interface to a point to point link, a communication bus, a point to multipoint link or a broadcast network.

According to an exemplary and non-limitative embodiment , the decoding device 100 further comprises a computer program stored in the memory 120. The computer program comprises instructions which, when executed by the decoding device 100, in particular by the processor 110, make the decoding device 100 carry out the method described with reference to figure 7. According to a variant, the computer program is

stored externally to the decoding device 100 on a non-transitory digital data support, e.g. on an external storage medium such as a HDD, CD-ROM, DVD, a read-only and/or DVD drive and/or a DVD Read/Write drive, all known in the art. The decoding device 100 thus comprises an interface to read the computer program. Further, the decoding device 100 could access one or more Universal Serial Bus (USB)-type storage devices (e.g., “memory sticks.”) through corresponding USB ports (not shown).

According to exemplary and non-limitative embodiments, the decoding device 100 is a device, which belongs to a set comprising:

- a mobile device ;
- a communication device ;
- a game device ;
- a set top box;
- a TV set;
- a tablet (or tablet computer) ;
- a laptop ;
- a display and
- a decoding chip.

Figure 7 represents a flowchart of a method for decoding a color mapping table coded as a hierarchical tree of nodes/octants arranged in levels from a bitstream F according to a specific and non-limitative embodiment .

Exemplarily, the color mapping table (e.g. a Look-Up Table) is described using a hierarchical tree of maximal depth cm_octant_depth, where the final nodes/octants are further split into YPartNum leaves. In this latter case the splitting is done in one dimension only, namely the Y dimension. It will be appreciated, however, that the invention is not restricted to the splitting of the final nodes/octants in the Y dimension. For each of these YPartNum leaves, at least one color parameter is decoded. The decoding of one color parameter may comprise decoding of residual values, e.g. res_y, res_u, res_v as defined in JCTVC_Q1008_v3. The residual values and the decoded color parameters may be stored in 3D arrays (e.g. res_y[yIdx][uIdx][vIdx],

res_u[yIdx][uIdx][vIdx], res_v[yIdx][uIdx][vIdx] and LutY [yIdx][uIdx][vIdx], LutU [yIdx][uIdx][vIdx], LutV [yIdx][uIdx][vIdx] respectively).

In a step S10, at least one data item representative of at least one color parameter associated with at least one node/octant of the hierarchical tree is decoded. The data item can be the color parameter itself or can be a residual representative of the color parameter.

In a step S12, the decoded at least one data item is associated with an index of the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and further based on the level position of the current node/octant in the hierarchical tree. To this aim, the syntax defined in the document *JCTVC-Q1008-V3* is amended as indicated in Table 2 (modified elements are shown in italics and bold). According to this table, the color mapping table is clearly encoded as a hierarchical tree of octants.

color_mapping_octants(depth, yIdx, uIdx, vIdx, length) {	**Descriptor**
if (depth < cm_octant_depth)	
split_octant_flag	u(1)
if (split_octant_flag)	
for(k = 0; k < 2; k++)	
for(m = 0; m < 2 ; m++)	
for(n = 0; n < 2; n++)	
color_mapping_octants(depth + 1, yIdx + YPartNum * k * length / 2, uIdx + m * length / 2, vIdx + n * length / 2, length / 2)	
else	
for(i = 0; i < YPartNum; i++)	
for(vertex = 0; vertex < 4; vertex++) {	
coded_vertex_flag[yIdx + ***(i<<sh)***][uIdx][vIdx][vertex]	u(1)
if(coded_vertex_flag[yIdx + i][uIdx][vIdx][vertex]) {	
res_y[yIdx + ***(i<<sh)***][uIdx][vIdx][vertex]	se(v)
res_u[yIdx + ***(i<<sh)***][uIdx][vIdx][vertex]	se(v)
res_v[yIdx + ***(i<<sh)***][uIdx][vIdx][vertex]	se(v)
}	
}	
}	

Table 2: Proposed Color LUT syntax.

where sh = cm_octant_depth – depth, where cm_octant_depth specifies the maximal split depth of the color mapping table and is further disclosed in section F.7.4.6.1 of JCTVC_Q1008_v3 and depth defines the current level or current depth in the hierarchical tree (i.e. the level/depth of the current node/octant in the hierarchical tree) as indicated in figures 2 and 3. In the following, the word “level” and “depth” can be used interchangeably.

Figure 8 represents the same configuration as depicted in Figure 4 but obtained with the method disclosed with respect to figure 7. Introducing the shift by i<<sh in table 2 makes it possible to obtain a nigh-uniform distribution of the color parameters in the LUT in case of non-explicitly-encoded color parameters (i.e. in the case where the

current depth is strictly inferior to cm_octant_depth). In Table 2 “yIdx” is the index (Y direction/component) of the current octant in the color mapping table and “i” is the relative index (aka offset) of the further split in the Y direction of the i^{th} leave .

In an optional step S14, a prediction is added to the decoded at least one data item, e.g. a residual, representative of the color parameter in order to decode a color parameter. Exemplarily, the prediction data is obtained from one or several previously decoded color parameter(s).

The method is advantageously iterated on all the nodes/octants to decode the whole color mapping table.

In a specific and non-limiting embodiment , the decoding/reconstruction of the color parameters is done in two steps as depicted in **Figure 9**. In a first step S20, all explicitly encoded color parameters are decoded. In a second step S22, the non-explicitly encoded color parameters are reconstructed from neighboring decoded explicitly encoded color parameters. In that way, the reconstruction process of these color parameters, i.e. non-explicitly encoded, can be done from any decoded explicitly-encoded color parameters, whatever its position in the LUT and independently of the parsing/decoding order of the explicitly-encoded color parameters, because the residuals are inferred to be zero for the non-explicitly encoded color parameters and the prediction is done with previously (decoding order) reconstructed colors parameters.

In the first step S20, the explicitly encoded color parameters are decoded, stored and marked (as explicitly encoded). This information (i.e. the marking as explicitly encoded) can be used when reconstructing at the non-explicitly-encoded color parameters reconstruction stage.

In the second step S22, non-explicitly-encoded color parameters (identified by “?” in the **figure 10**) are reconstructed (by copy or by interpolation) from neighboring decoded explicitly-encoded color parameters (circles) but not in decoding order (multi-pass process). In this example, the distance derivation of non-explicitly-encoded color parameters to the corresponding explicitly-encoded color parameters is inferior or equal to R/4, that is less than with previous existing solution (Figure 4) thanks to a better sampling. In figure 10, the non-explicitly encoded color parameters (identified with a

question mark) are derived from explicitly encoded color parameters (identified with circles). The derivation process is illustrated by the arrows.

Advantageously, a lower distance derivation between explicitly-encoded color parameters and non-explicitly-encoded color parameters is used. Thus the non-explicitly-encoded color parameters are better predicted. In the case where there exist non-explicitly-encoded color parameters (use of split_octant_flag), a better color space sampling of the 3D LUT is obtained.

Figure 11 depicts a device 200 for encoding a color mapping table as a hierarchical tree of nodes arranged in levels according to a specific and non-limitative embodiment. Each node represents an octant, i.e. a portion of a 3D color space, e.g. YUV or RGB color space. The encoding device 200 comprises an input 20 configured to receive a color mapping table, e.g. a 3D LUT. The input 20 is linked to a module 24 either directly or via an optional module 22. The module 24 is configured to select at least one data item representative of at least one color parameter associated with at least one current node/octant of a hierarchical tree of octants arranged in levels, wherein the selected data item is associated with an index of the color mapping table determined at least based on a maximal split depth of the color mapping table and based on the level of at least one current octant in a hierarchical tree of octants arranged in levels. Exemplarily, the module 24 comprises a memory in which the data item is stored at a location determined by the index. The module 24 is linked to an encoding module 26 configured to encode the selected at least one data item.

Optionally, the encoding device 200 further comprises the module 22 configured to subtract a prediction from a color parameter to obtain a residual which is the data item to be encoded.

The bitstream may be stored in a memory internal to the encoding device 200 or external to it. In a variant, the bitstream is sent to a destination.

Figure 12 represents an exemplary architecture of the encoding device 200 configured to encode a color mapping table in a bitstream according to an exemplary embodiment. The encoding device 200 comprises one or more processor(s) 210 ,

which is(are), for example, a CPU, a GPU and/or a DSP (English acronym of Digital Signal Processor), along with internal memory 220 (e.g. RAM, ROM, EPROM). The encoding device 200 comprises one or several Input/Output interface(s) 230 adapted to display output information and/or allow a user to enter commands and/or data (e.g. a keyboard, a mouse, a touchpad, a webcam); and a power source 240 which may be external to the encoding device 200. The device 200 may also comprise network interface(s) (not shown). The color mapping table may be obtained from a source. According to different embodiments , the source belongs to a set comprising:

- a local memory, e.g. a video memory, a RAM, a flash memory, a hard disk ;
- a storage interface, e.g. an interface with a mass storage, a ROM, an optical disc or a magnetic support;
- a communication interface, e.g. a wireline interface (for example a bus interface, a wide area network interface, a local area network interface) or a wireless interface (such as a IEEE 802.11 interface or a Bluetooth interface); and
- an image capturing circuit (e.g. a sensor such as, for example, a CCD (or Charge-Coupled Device) or CMOS (or Complementary Metal-Oxide-Semiconductor)).

According to different embodiments, the bitstream may be sent to a destination. As an example, the bitstream is stored in a remote or in a local memory, e.g. a video memory or a RAM, a hard disk. In a variant, the bitstream is sent to a storage interface, e.g. an interface with a mass storage, a ROM, a flash memory, an optical disc or a magnetic support and/or transmitted over a communication interface, e.g. an interface to a point to point link, a communication bus, a point to multipoint link or a broadcast network.

According to an exemplary and non-limitative embodiment, the encoding device 200 further comprises a computer program stored in the memory 220. The computer program comprises instructions which, when executed by the encoding device 200, in particular by the processor 210, make the encoding device 200 carry out the method described with reference to figure 13. According to a variant, the computer program is stored externally to the encoding device 200 on a non-transitory digital data support, e.g. on an external storage medium such as a HDD, CD-ROM, DVD, a read-only and/or

DVD drive and/or a DVD Read/Write drive, all known in the art. The encoding device 200 thus comprises an interface to read the computer program. Further, the encoding device 200 could access one or more Universal Serial Bus (USB)-type storage devices (e.g., "memory sticks.") through corresponding USB ports (not shown).

According to exemplary and non-limitative embodiments, the encoding device 200 is a device, which belongs to a set comprising:

- a mobile device ;
- a communication device ;
- a game device ;
- a set top box;
- a TV set;
- a tablet (or tablet computer) ;
- a laptop ;
- a display and
- an encoding chip.

Figure 13 represents a flowchart of a method for encoding a color mapping table in a bitstream F according to a specific and non-limitative embodiment .

Exemplarily, the color mapping table (e.g. defined as Look-Up Table) is described using a hierarchical tree of maximal depth cm_octant_depth, where the final nodes/octants are further split into YPartNum leaves. In this latter case the splitting is done in one dimension only, namely the Y dimension. It will be appreciated, however, that the invention is not restricted to the splitting of the final nodes/octants in the Y dimension. For each of these YPartNum leaves, color parameters are encoded. The encoding of one color parameter may comprise coding of residual values, e.g. res_y, res_u, res_v as defined in JCTVC_Q1008_v3. The residual values and the color parameters may be stored in 3D arrays (e.g. res_y[yIdx][uIdx][vIdx], res_u[yIdx][uIdx][vIdx], res_v[yIdx][uIdx][vIdx] and LutY [yIdx][uIdx][vIdx], LutU [yIdx][uIdx][vIdx], LutV [yIdx][uIdx][vIdx] respectively).

In a step S32, at least one data item representative of at least one color parameter associated with at least one node/octant of the hierarchical tree is selected.

The data item can be the color parameter itself or can be a residual representative of the color parameter. The selected at least one data item is associated with an index of the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and further based on the level position of the current node/octant in the hierarchical tree. To this aim, the syntax defined in the document *JCTVC-Q1008-V3* is amended as indicated in Table 3. According to this table, the color mapping table is clearly encoded as a hierarchical tree of octants.

color_mapping_octants(depth, yIdx, uIdx, vIdx, length) {	**Descriptor**
if (depth < cm_octant_depth)	
split_octant_flag	u(1)
if (split_octant_flag)	
for(k = 0; k < 2; k++)	
for(m = 0; m < 2 ; m++)	
for(n = 0; n < 2; n++)	
color_mapping_octants(depth + 1, yIdx + YPartNum * k * length / 2, uIdx + m * length / 2, vIdx + n * length / 2, length / 2)	
else	
for(i = 0; i < YPartNum; i++)	
for(vertex = 0; vertex < 4; vertex++) {	
coded_vertex_flag[yIdx + ***(i<<sh)***][uIdx][vIdx][vertex]	u(1)
if(coded_vertex_flag[yIdx + i][uIdx][vIdx][vertex]) {	
res_y[yIdx + ***(i<<sh)***][uIdx][vIdx][vertex]	se(v)
res_u[yIdx + ***(i<<sh)***][uIdx][vIdx][vertex]	se(v)
res_v[yIdx + ***(i<<sh)***][uIdx][vIdx][vertex]	se(v)
}	
}	
}	

Table 3: Proposed Color LUT syntax.

where sh = cm_octant_depth – depth, where cm_octant_depth specifies the maximal split depth of the color mapping table and is further disclosed in section F.7.4.6.1 of

JCTVC_Q1008_v3 and depth defines the current level in the hierarchical tree, as indicated in figures 2 and 3. Introducing the shift by i<<sh in table 2 makes it possible to obtain a nigh-uniform distribution of the color parameters in the LUT in case of non-explicitly-encoded color parameters (i.e. in the case where the current depth is strictly inferior to cm_octant_depth).

In a step S34, the at least one data item representative of at least one color parameter associated with at least one node/octant of the hierarchical tree is coded in a bitstream.

In an optional step S30, encoding at least one data item representative of a color parameter comprises subtracting a prediction from the color parameter to obtain a residual, said residual being the data item, wherein said prediction is obtained from at least one previously encoded and reconstructed color parameter.

The method is advantageously iterated on all the nodes/octants to encode the whole color mapping table. Some color parameters are non-explicitly-encoded.

In a specific embodiment, the color mapping table is determined as follows previous to its encoding as depicted on Figure 14. In a first step S40, a full LUT (i.e. a LUT in which all the indices of the LUT are associated with data) is computed from a first frame P1 and a second frame P2. As an example, computing the full LUT comprises minimizing a difference between P2 and T(P1), where T(P1) is the first frame P1 color transformed with the LUT. As example, the difference is an average of the squared sample differences. In a variant, the difference is a PSNR.

In a second step S42, the level of splitting in each branch of the hierarchical tree is determined with a pruning stage for example. The pruning stage allows for merging 8 octants in the octree. For example, in Figure 15, eight octants (delineated with bold lines) with 4 partitions in Y direction corresponding to 8 x 4 data parameters set (1,2,...32) are merged into one single octant with 4 partitions in Y direction corresponding to 4 color data parameters set (1',3',5',7') as depicted on figure 16.The values (1',3',5',7') and possibly A' may be recomputed in order to minimize the difference between T(P1) (computed with the reconstructed equivalent LUT) and P2 because (1',3',5',7') and A' are copied at several locations (index) in the reconstructed LUT. Indeed, since A' is used as predictor for the 8 color parameters in the Y direction

and no residuals is encoded (inferred to be zero), the reconstructed 8 color parameters are equal to A'.

Advantageously, the minimization may take into account the LUT encoding cost (and possibly the full video stream cost) using a Lagrangian criterion for instance.

The step "pruning" may be iterated several times until a criterion/condition C1 is reached. Exemplarily, the "pruning" step may be iterated until the difference between P2 and T(P1) is below a threshold value. In a variant, the "pruning" step may be iterated until the difference between P2 and T(P1) does not decrease between two successive iterations. C1 may take into account the LUT encoding cost and possibly the full video stream cost as well.

In a step S44, the LUT after pruning is encoded according to the method disclosed with respect to figure 13.

The final full resolution LUT (as decoded by the decoder) is depicted in Figure 17, where we see A' is copied at the locations 9, 10…16.

The implementations described herein may be implemented in, for example, a method or a process, an apparatus, a software program, a data stream, or a signal. Even if only discussed in the context of a single form of implementation (for example, discussed only as a method or a device), the implementation of features discussed may also be implemented in other forms (for example a program). An apparatus may be implemented in, for example, appropriate hardware, software, and firmware. The methods may be implemented in, for example, an apparatus such as, for example, a processor, which refers to processing devices in general, including, for example, a computer, a microprocessor, an integrated circuit, or a programmable logic device. Processors also include communication devices, such as, for example, computers, cell phones, portable/personal digital assistants ("PDAs"), and other devices that facilitate communication of information between end-users.

Implementations of the various processes and features described herein may be embodied in a variety of different equipment or applications, particularly, for example, equipment or applications. Examples of such equipment include an encoder, a decoder, a post-processor processing output from a decoder, a pre-processor providing input to

an encoder, a video coder, a video decoder, a video codec, a web server, a set-top box, a laptop, a personal computer, a cell phone, a PDA, and other communication devices. As should be clear, the equipment may be mobile and even installed in a mobile vehicle.

Additionally, the methods may be implemented by instructions being performed by a processor, and such instructions (and/or data values produced by an implementation) may be stored on a processor-readable medium such as, for example, an integrated circuit, a software carrier or other storage device such as, for example, a hard disk, a compact diskette ("CD"), an optical disc (such as, for example, a DVD, often referred to as a digital versatile disc or a digital video disc), a random access memory ("RAM"), or a read-only memory ("ROM"). The instructions may form an application program tangibly embodied on a processor-readable medium. Instructions may be, for example, in hardware, firmware, software, or a combination. Instructions may be found in, for example, an operating system, a separate application, or a combination of the two. A processor may be characterized, therefore, as, for example, both a device configured to carry out a process and a device that includes a processor-readable medium (such as a storage device) having instructions for carrying out a process. Further, a processor-readable medium may store, in addition to or in lieu of instructions, data values produced by an implementation.

As will be evident to one of skill in the art, implementations may produce a variety of signals formatted to carry information that may be, for example, stored or transmitted. The information may include, for example, instructions for performing a method, or data produced by one of the described implementations. For example, a signal may be formatted to carry as data the rules for writing or reading the syntax of a described embodiment, or to carry as data the actual syntax-values written by a described embodiment. Such a signal may be formatted, for example, as an electromagnetic wave (for example, using a radio frequency portion of spectrum) or as a baseband signal. The formatting may include, for example, encoding a data stream and modulating a carrier with the encoded data stream. The information that the signal carries may be, for example, analog or digital information. The signal may be transmitted over a variety of

different wired or wireless links, as is known. The signal may be stored on a processor-readable medium.

A number of implementations have been described. Nevertheless, it will be understood that various modifications may be made. For example, elements of different implementations may be combined, supplemented, modified, or removed to produce other implementations. Additionally, one of ordinary skill will understand that other structures and processes may be substituted for those disclosed and the resulting implementations will perform at least substantially the same function(s), in at least substantially the same way(s), to achieve at least substantially the same result(s) as the implementations disclosed. Accordingly, these and other implementations are contemplated by this application.

Claims

1. A method for decoding a color mapping table comprising:
- decoding (S10) at least one data item representative of a color parameter associated with at least one current octant of a hierarchical tree of octants;
- associating (S12) said decoded at least one data item with an index in the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a current depth of said at least one current octant in the hierarchical tree.

2. The method for decoding according to claim 1, wherein associating said decoded at least one data item with an index of the color mapping table determined at least based on a maximal split depth of the color mapping table and based on a current depth of said at least one current octant in the hierarchical tree comprises storing said at least one data item in an array at a location determined by said index.

3. The method for decoding according to claim 1 or 2, wherein said at least one data item is a residual of the color parameter.

4. The method for decoding according to any of claims 1 to 3, further comprising adding a prediction to said decoded at least one data item in order to obtain a decoded color parameter, wherein said prediction is obtained from at least one previously decoded color parameter.

5. The method for decoding according to claim 4, further comprising reconstructing a non-explicitly encoded color parameter at least from said decoded color parameter.

6. The method for decoding according to any of claims 1 to 5, wherein three data items representative of a color parameter are decoded, one for each color component.

7. The method for decoding according to any of claims 1 to 6, wherein said index equal yIdx + *(i*<< cm_octant_depth – depth*)* where yIdx is the index in one direction of the current octant, i is the relative index of a further split in said direction of an i^{th} leave of the hierarchical tree and cm_octant_depth is maximal split depth and depth is the level of said at least one current octant in the hierarchical tree.

8. A decoding device for decoding a color mapping table coded as a hierarchical tree of octants arranged in levels, said decoding device comprising a processor configured to:
a) decoding at least one data item representative of a color parameter associated with at least one current octant of said hierarchical tree;
b) associating said decoded at least one data item with an index in the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a level position of said at least one current octant in the hierarchical tree.

9. A device for decoding a color mapping table comprising:
- means for decoding at least one data item representative of a color parameter associated with at least one current octant of a hierarchical tree of octants ;
- means for associating said decoded at least one data item with an index in the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a current depth of said at least one current octant in the hierarchical tree.

10. The device according to claim 9, wherein said device is configured to execute the steps of the method for decoding according to any of claims 2 to 7.

11. A computer program product comprising program code instructions to execute the steps of the decoding method according to any of claims 1 to 7 when this program is executed on a computer.

12. A processor readable medium having stored therein instructions for causing a processor to perform at least the steps of the decoding method according to any of claims 1 to 7.

13. A method for encoding a color mapping table comprising:

- selecting at least one data item representative of a color parameter associated with at least one current octant of a hierarchical tree of octants, wherein said selected at least one data item is associated with an index in the color mapping table determined at least based on a maximal split depth of the color mapping table and based on a current depth of at least one current octant in the hierarchical tree; and
- encoding said selected at least one data item.

14. An encoding device for encoding a color mapping table comprising:

- means for selecting at least one data item representative of a color parameter associated with at least one current octant of a hierarchical tree of octants, wherein said selected at least one data item is associated with an index in the color mapping table determined at least based on a maximal split depth of the color mapping table and based on a current depth of at least one current octant in the hierarchical tree; and
- means for encoding said selected at least one data item.

15. An encoding device for encoding a color mapping table comprising at least a processor configured to :

- select at least one data item representative of a color parameter associated with at least one current octant of a hierarchical tree of octants, wherein said selected at least one data item is associated with an index in the color mapping table determined at least based on a maximal split depth of the color mapping table and based on a current depth of at least one current octant in the hierarchical tree; and
- encode said selected at least one data item.

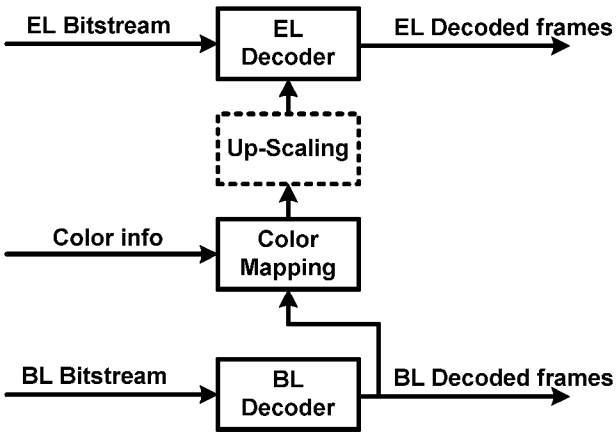


FIGURE 1 – **Prior Art**

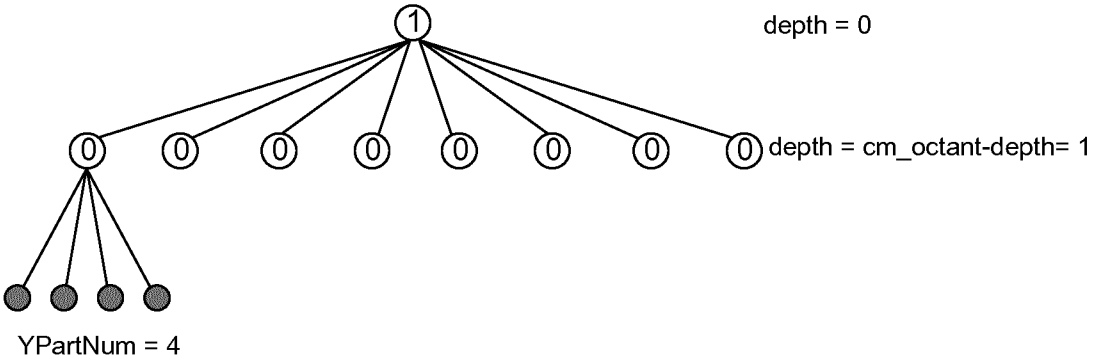


FIGURE 2 – Prior Art

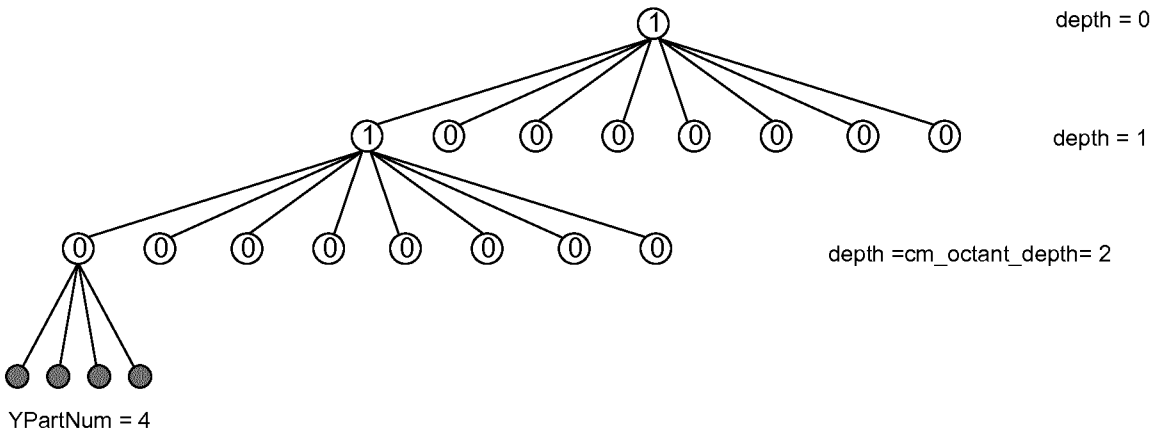


FIGURE 3 – Prior Art

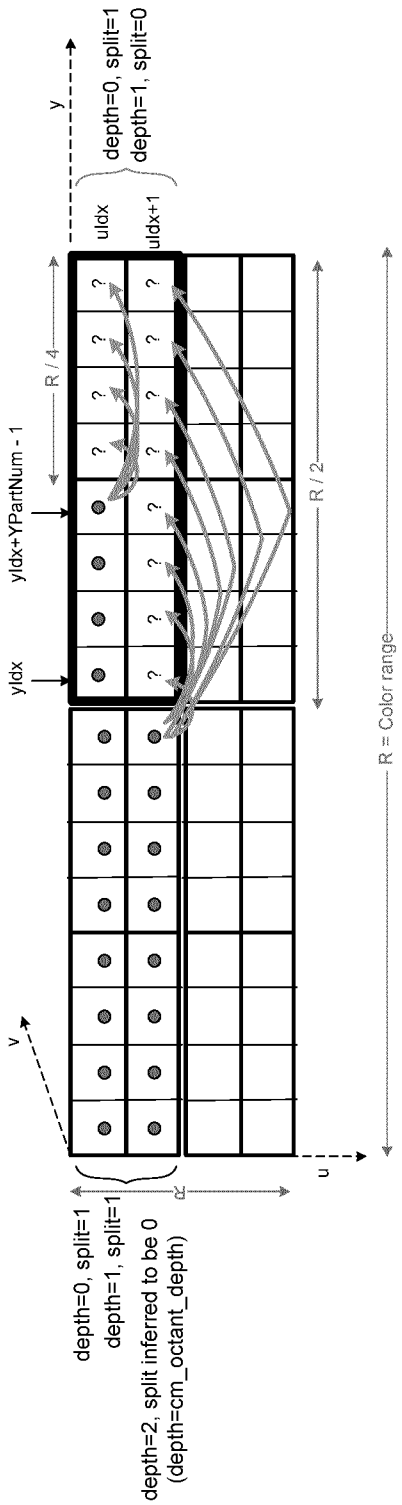


FIGURE 4 – Prior Art

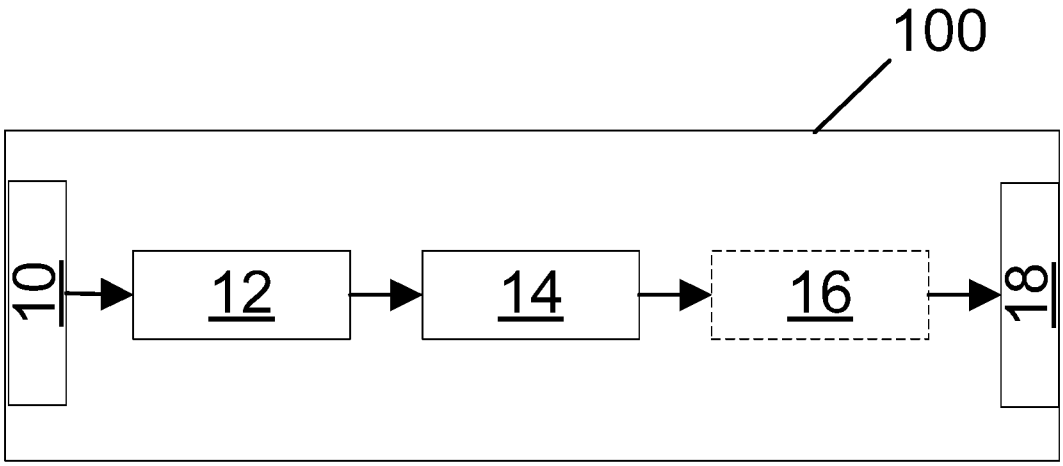


FIGURE 5

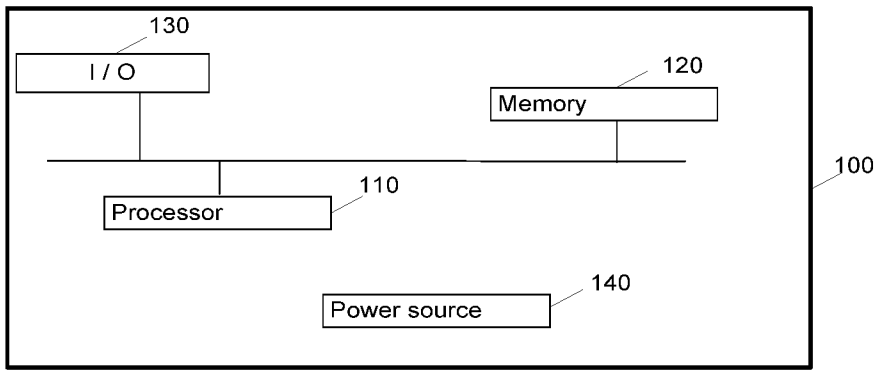


FIGURE 6

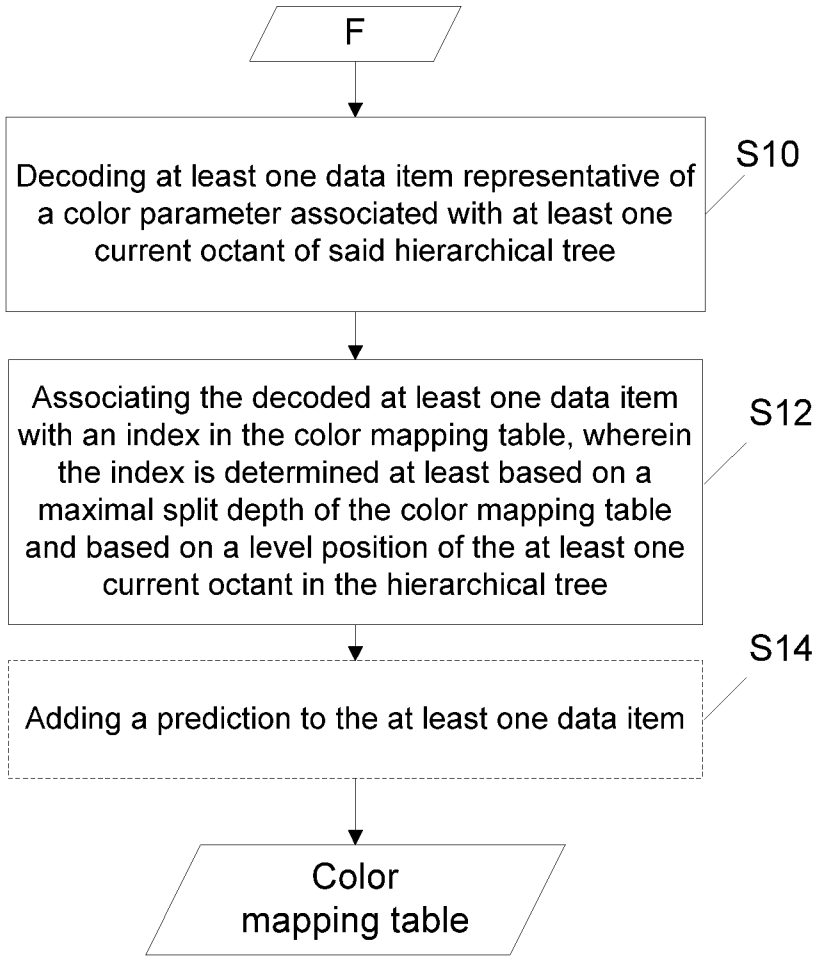

F
Decoding at least one data item representative of a color parameter associated with at least one current octant of said hierarchical tree
S10
Associating the decoded at least one data item with an index in the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a level position of the at least one current octant in the hierarchical tree
S12
Adding a prediction to the at least one data item
S14
Color mapping table

FIGURE 7

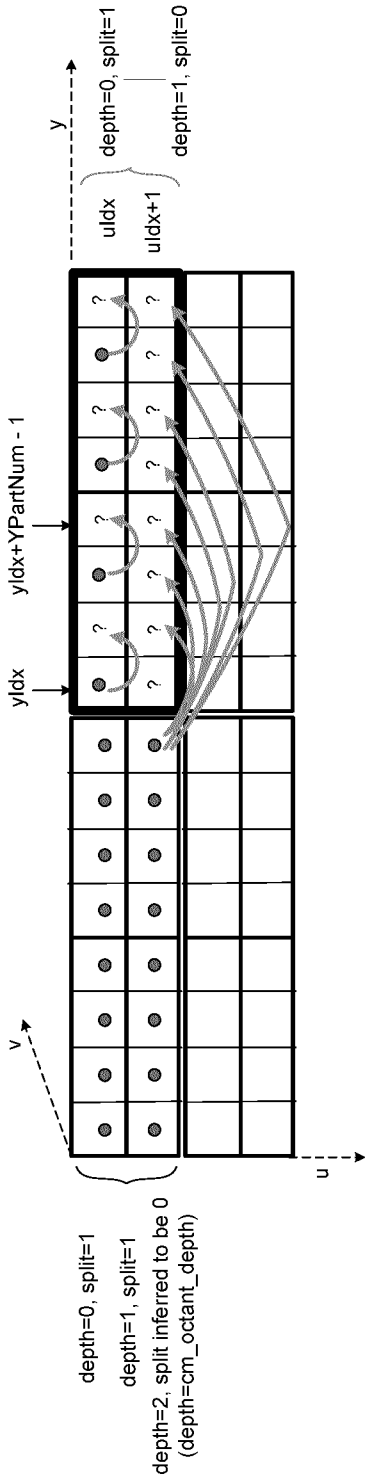

y
depth=0, split=1
depth=1, split=0
uIdx
uIdx+1
yIdx+YPartNum - 1
yIdx
v
u
depth=0, split=1
depth=1, split=1
depth=2, split inferred to be 0
(depth=cm_octant_depth)

FIGURE 8

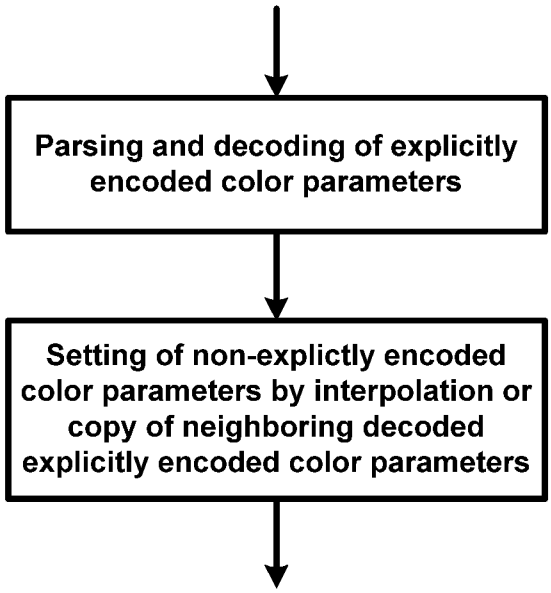

Parsing and decoding of explicitly encoded color parameters
Setting of non-explictly encoded color parameters by interpolation or copy of neighboring decoded explicitly encoded color parameters

FIGURE 9

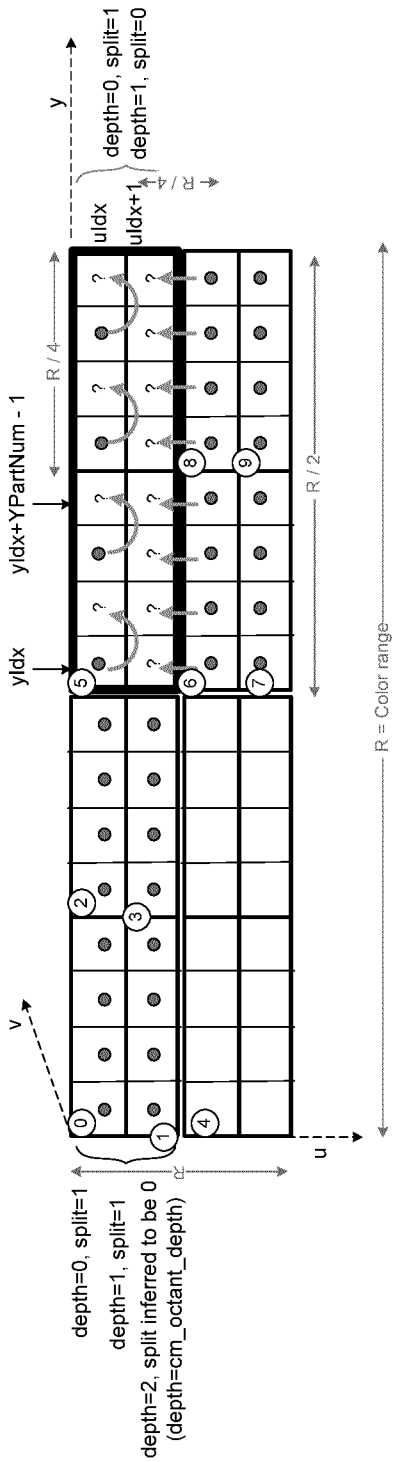

y
depth=0, split=1
depth=1, split=0
uIdx
uIdx+1
R / 4
R / 4
yIdx+YPartNum - 1
yIdx
R / 2
R = Color range
v
u
R
depth=0, split=1
depth=1, split=1
depth=2, split inferred to be 0
(depth=cm_octant_depth)

FIGURE 10

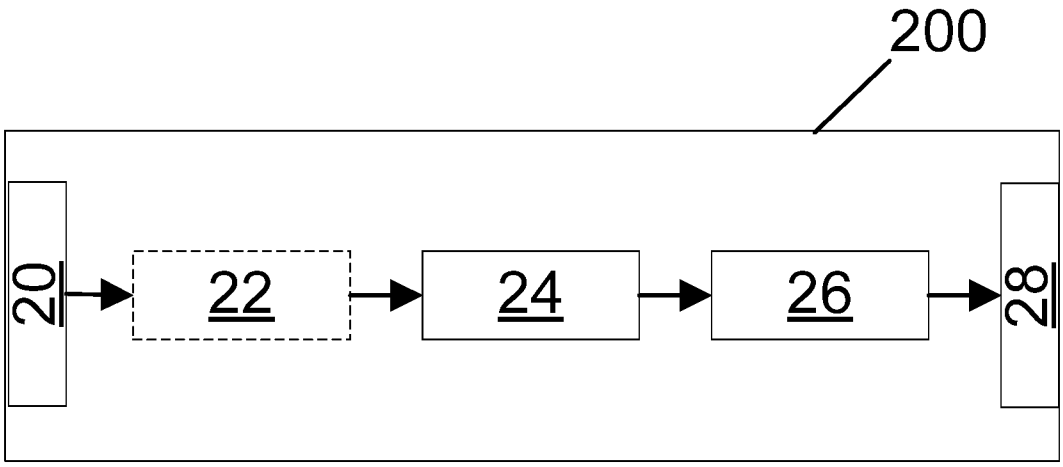


FIGURE 11

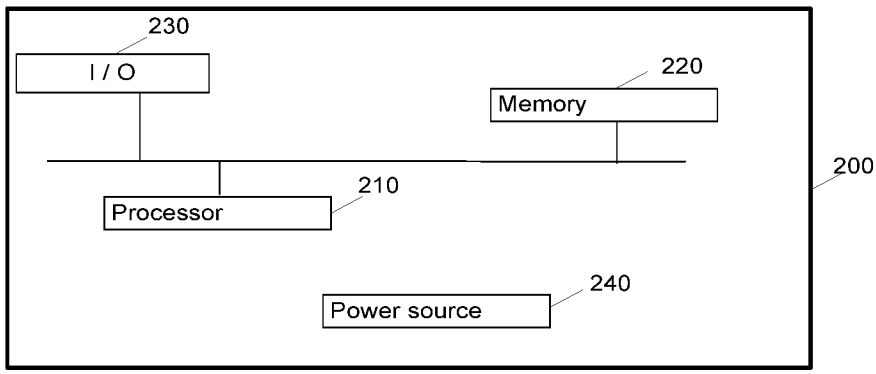


FIGURE 12

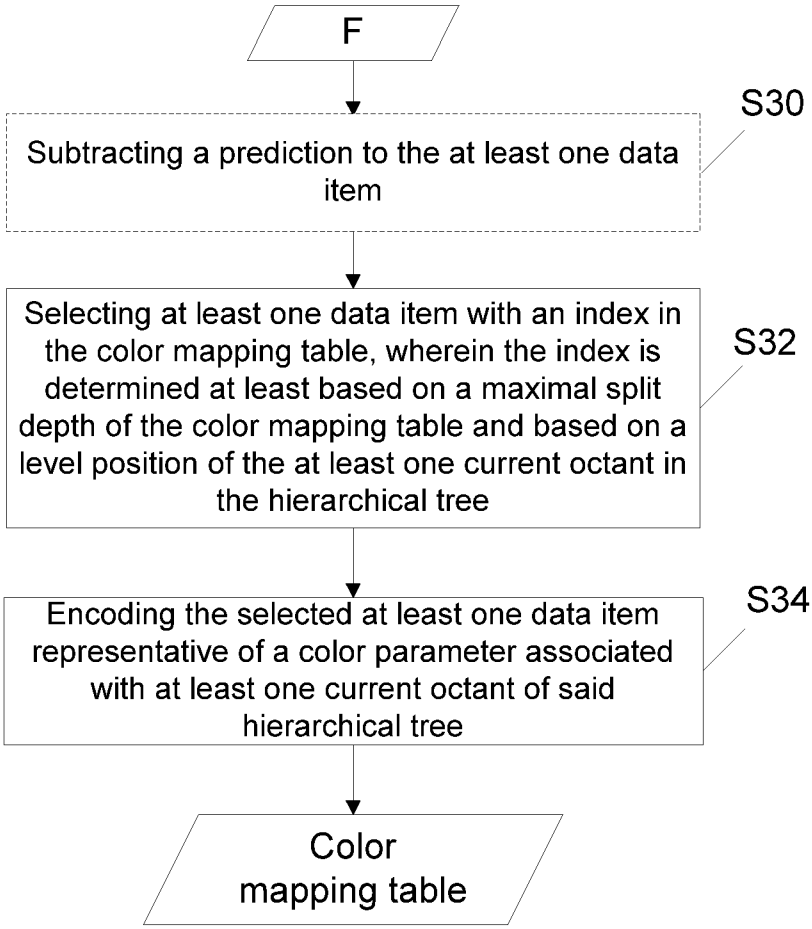

F
S30
Subtracting a prediction to the at least one data item
Selecting at least one data item with an index in the color mapping table, wherein the index is determined at least based on a maximal split depth of the color mapping table and based on a level position of the at least one current octant in the hierarchical tree
S32
Encoding the selected at least one data item representative of a color parameter associated with at least one current octant of said hierarchical tree
S34
Color mapping table

FIGURE 13

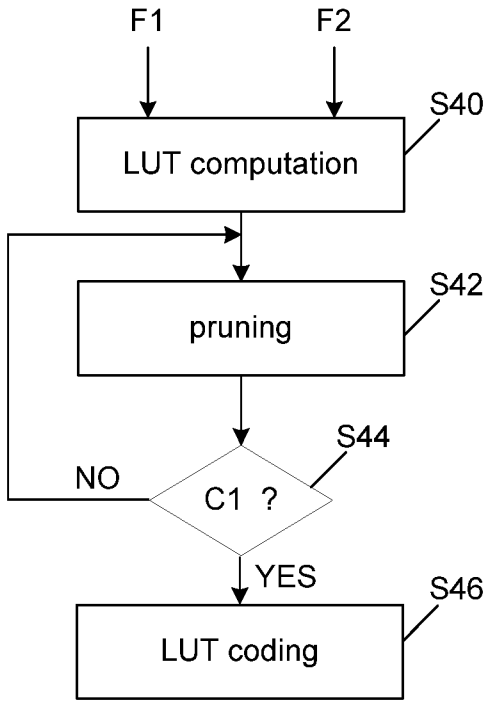


FIGURE 14

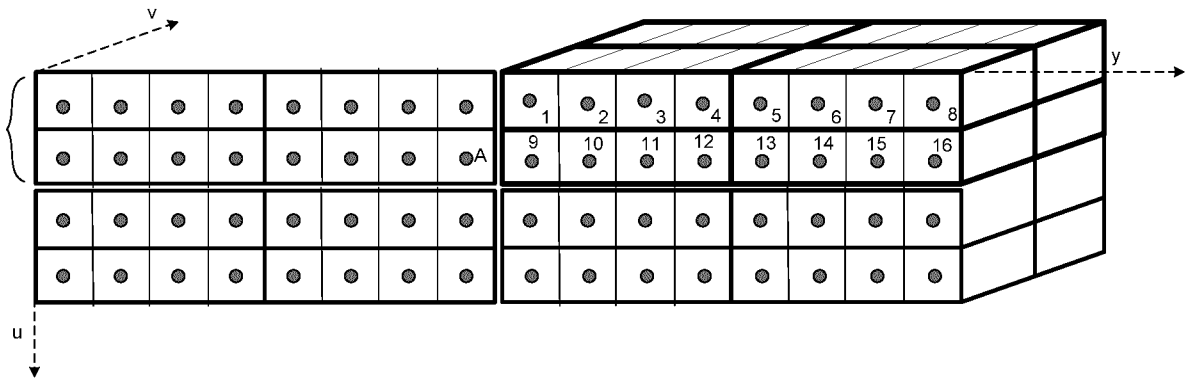


FIGURE 15

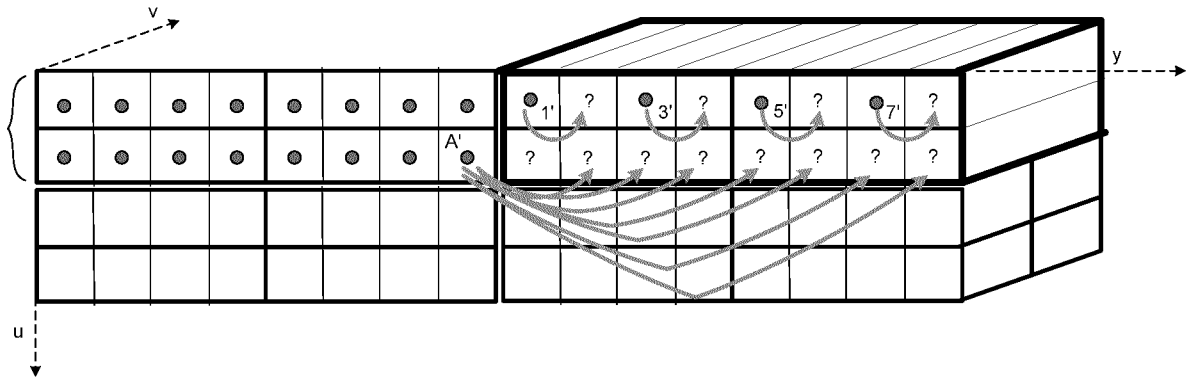


FIGURE 16

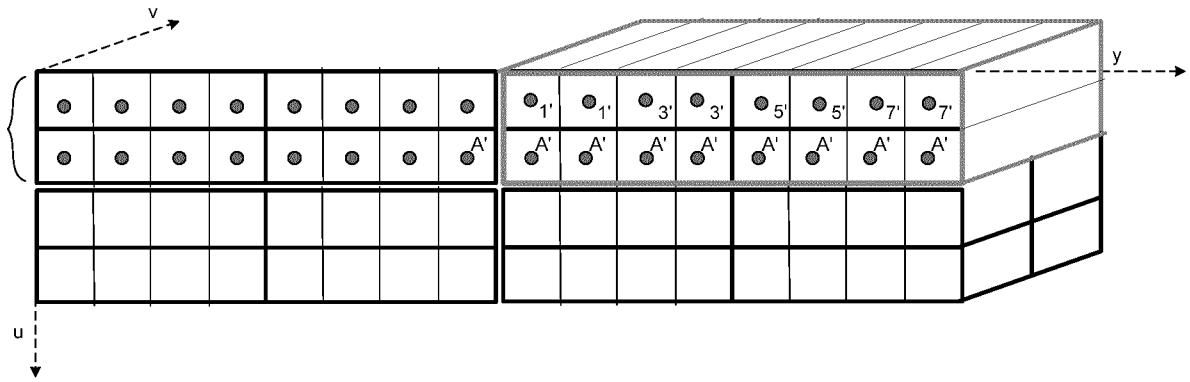


FIGURE 17

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2015/062909

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N19/50 H04N19/30 H04N19/463 H04N19/186
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	BORDES P ET AL: "Bug fix in SHVC draft specification", 18. JCT-VC MEETING; 30-6-2014 - 9-7-2014; SAPPORO; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/, no. JCTVC-R0164, 21 June 2014 (2014-06-21) , XP030116447, the whole document ----- -/--	1-15

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
21 August 2015	31/08/2015

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Kontopodis, D

1

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	LI X ET AL: "Non-SCE1: Asymmetric 3D LUT for Color Gamut Scalability", 16. JCT-VC MEETING; 9-1-2014 - 17-1-2014; SAN JOSE; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-P0063-v2, 9 January 2014 (2014-01-09), XP030115532, sections 3.1, 3.2	1-15
A	CHEN J ET AL: "High efficiency video coding (HEVC) scalable extension Draft 6", 17. JCT-VC MEETING; 27-3-2014 - 4-4-2014; VALENCIA; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/, no. JCTVC-Q1008, 15 April 2014 (2014-04-15), XP030116232, cited in the application section F.7.3.2.3.2 section H.8.1.4.3.2	1-15

1