



(51) International Patent Classification:

H04N 19/85 (2014.01) H04N 19/70 (2014.01)
H04N 19/30 (2014.01)

(21) International Application Number:

PCT/JP2014/003676

(22) International Filing Date:

10 July 2014 (10.07.2014)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

61/846,060 14 July 2013 (14.07.2013) US

(71) Applicant: SHARP KABUSHIKI KAISHA [JP/JP]; 22-22, Nagaike-cho, Abeno-ku, Osaka-shi, Osaka, 5458522 (JP).

(72) Inventor: DESHPANDE, Sachin G..

(74) Agent: HARAKENZO WORLD PATENT & TRADE-MARK; Daiwa Minamimorimachi Building, 2-6, Tenjin-bashi 2-chome Kita, Kita-ku, Osaka-shi, Osaka, 5300041 (JP).

(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, LT, 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, SZ, 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).

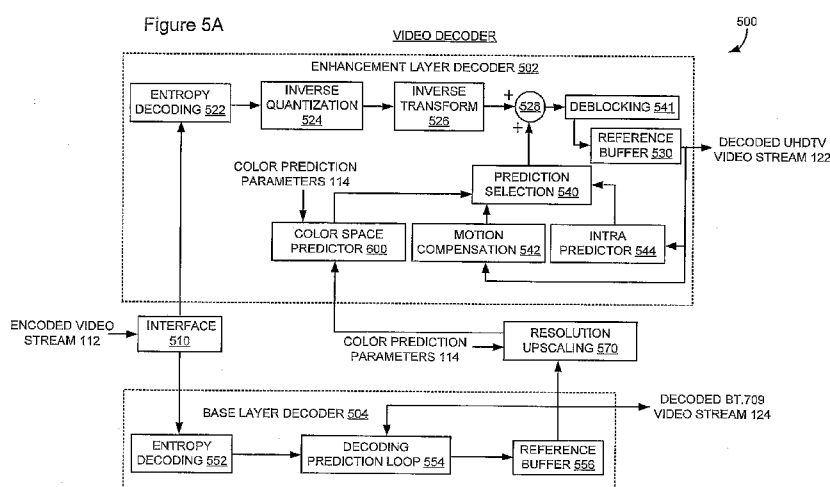
Declarations under Rule 4.17:

- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))
- as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))

Published:

- with international search report (Art. 21(3))

(54) Title: VIDEO PARAMETER SET SIGNALING



(57) Abstract: A scalable video decoder provides a method for decoding a video sequence comprising: decoding a plurality of frames of said video sequence where at least two of said frames belonging to a different one of a plurality of layers; said video sequence includes a video parameter set (VPS) syntax used for said decoding regions of an encoded video; said VPS syntax includes a VPS extension set syntax including attributes regarding representation format characteristics (e.g. color characteristics); said VPS extension set syntax including a number of representation format characteristic syntax structures and an index referencing a representation format structure for each layer; each of said number of said representation format characteristic syntax structures indicates a different set of said attributes.

Description

Title of Invention: VIDEO PARAMETER SET SIGNALING

Technical Field

[0001] This disclosure relates generally to video coding, and, more particularly, to color space prediction for video coding.

Background Art

[0002] Many systems include a video encoder to implement video coding standards and compress video data for transmission over a channel with limited bandwidth and/or limited storage capacity. These video coding standards can include multiple coding stages such as intra prediction, transform from spatial domain to frequency domain, inverse transform from frequency domain to spatial domain, quantization, entropy coding, motion estimation, and motion compensation, in order to more effectively encode frames.

Summary of Invention

Technical Problem

[0003] Traditional digital High Definition (HD) content can be represented in a format described by video coding standard International Telecommunication Union Radio-communication Sector (ITU-R) Recommendation BT.709, which defines a resolution, a color gamut, a gamma, and a quantization bit-depth for video content. With an emergence of higher resolution video standards, such as ITU-R Ultra High Definition Television (UHDTV), which, in addition to having a higher resolution, can have wider color gamut and increased quantization bit-depth compared to BT.709, many legacy systems based on lower resolution HD content may be unable to utilize compressed UHDTV content. One of the current solutions to maintain the usability of these legacy systems includes separately simulcasting both compressed HD content and compressed UHDTV content. Although a legacy system receiving the simulcasts has the ability to decode and utilize the compressed HD content, compressing and simulcasting multiple bitstreams with the same underlying content can be an inefficient use of processing, bandwidth, and storage resources.

Solution to Problem

[0004] According to the present invention, there is provided a method for decoding a video sequence, comprising:

[0005] (a) decoding a plurality of frames of said video sequence where at least two of said frames belonging to a different one of a plurality of layers;

[0006] (b) said video sequence includes a video parameter set syntax used for said decoding of regions of said encoded video;

- [0007] (c) said video parameter set syntax includes a video parameter set extension set syntax including attributes regarding representation format characteristics of an encoded video format used for said decoding of said regions of said video sequence;
- [0008] (d) said video parameter set syntax includes said video parameter set extension set syntax including the number of representation format characteristic syntax structures and an index referencing a representation format structure for each layer in said video parameter set extension set syntax;
- [0009] (e) each of said number of said representation format characteristic syntax structures in said video parameter set extension set syntax indicates a different set of said attributes regarding said representation format characteristics of said encoded video format used for decoding said regions of said video sequence.
- [0010] The foregoing and other objectives, features, and advantages of the invention will be more readily understood upon consideration of the following detailed description of the invention, taken in conjunction with the accompanying drawings.

Brief Description of Drawings

- [0011] [fig.1]Figure 1 is a block diagram example of a video coding system.
- [fig.2]Figure 2 is an example graph 200 illustrating color gamuts supported in a BT.709 video standard and in a UHDTV video standard.
- [fig.3A]Figures 3A is a block diagram example of the video encoder shown in Figure 1.
- [fig.3B]Figures 3B is a block diagram example of the video encoder shown in Figure 1.
- [fig.4]Figure 4 is a block diagram example of the color space predictor shown in Figure 3A and 3B.
- [fig.5A]Figures 5A is a block diagram example of the video decoder shown in Figure 1.
- [fig.5B]Figures 5B is a block diagram example of the video decoder shown in Figure 1.
- [fig.6]Figure 6 is a block diagram example of a color space predictor shown in Figure 5A and 5B.
- [fig.7]Figure 7 is an example operational flowchart for color space prediction in the video encoder shown in Figure 1.
- [fig.8]Figure 8 is an example operational flowchart for color space prediction in the video decoder shown in Figure 1.
- [fig.9]Figure 9 is another example operational flowchart for color space prediction in the video decoder shown in Figure 1.
- [fig.10A]Figures 10A is a block diagram example of video encoders that include color

bit depth scaling.

[fig.10B] Figures 10B is a block diagram example of video encoders that include color bit depth scaling.

[fig.11] Figure 11 is a flow diagram of an encoding method that includes bit depth scaling.

[fig.12A] Figures 12A is a block diagram example of the video decoders that include color bit depth scaling.

[fig.12B] Figures 12B is a block diagram example of the video decoders that include color bit depth scaling.

[fig.13] Figure 13 is a flow diagram of a decoding method that includes bit depth scaling.

Description of Embodiments

[0012] Figure 1 is a block diagram example of a video coding system 100. The video coding system 100 can include a video encoder 300 to receive video streams, such as an Ultra High Definition Television (UHDTV) video stream 102, standardized as BT.2020, and a BT.709 video stream 104, and to generate an encoded video stream 112 based on the video streams. The video encoder 300 can transmit the encoded video stream 112 to a video decoder 500. The video decoder 500 can decode the encoded video stream 112 to generate a decoded UHDTV video stream 122 and/or a decoded BT.709 video stream 124.

[0013] The UHDTV video stream 102 can have a different resolution, different quantization bit-depth, and represent different color gamut compared to the BT.709 video stream 104. For example, a UHDTV or BT.2020 video standard has a format recommendation that can support a 4k (3840x2160 pixels) or an 8k (7680x4320 pixels) resolution and a 10 or 12 bit quantization bit-depth. The BT.709 video standard has a format recommendation that can support a 2k (1920x1080 pixels) resolution and an 8 or 10 bit quantization bit-depth. The UHDTV format recommendation also can support a wider color gamut than the BT.709 format recommendation. Embodiments of the color gamut difference between the UHDTV video standard and the BT.709 video standard will be shown and described below in greater detail with reference to Figure 2.

[0014] The video encoder 300 can include an enhancement layer encoder 302 and a base layer encoder 304. The base layer encoder 304 can implement video encoding for High Definition (HD) content, for example, with a codec implementing a Moving Picture Experts Group (MPEG)-2 standard, or the like. The enhancement layer encoder 302 can implement video encoding for UHDTV content. In some embodiments, the enhancement layer encoder 302 can encode an UHDTV video frame by generating a prediction of at least a portion of the UHDTV image frame using a motion com-

compensation prediction, an intra-frame prediction, and a scaled color prediction from a BT.709 image frame encoded in the base layer encoder 302. The video encoder 300 can utilize the prediction to generate a prediction residue, for example, a difference between the prediction and the UHDTV image frame, and encode the prediction residue in the encoded video stream 112.

[0015] In some embodiments, when the video encoder 300 utilizes a scaled color prediction from the BT.709 image frame, the video encoder 300 can transmit color prediction parameters 114 to the video decoder 500. The color prediction parameters 114 can include parameters utilized by the video encoder 300 to generate the scaled color prediction. For example, the video encoder 300 can generate the scaled color prediction through an independent color channel prediction or an affine matrix-based color prediction, each having different parameters, such as a gain parameter per channel or a gain parameter and an offset parameter per channel. The color prediction parameters 114 can include parameters corresponding to the independent color channel prediction or the affine matrix-based color prediction utilized by the video encoder 300. In some embodiments, the encoder 300 can include the color prediction parameters 114 in a normative portion of the encoded video stream 112, for example, in a Sequence Parameter Set (SPS), a Picture Parameter Set (PPS), or another lower level section of the normative portion of the encoded video stream 112. In some embodiments, the video encoder 300 can utilize default color prediction parameters 114, which may be preset in the video decoder 500, alleviating the video encoder 300 from having to transmit color prediction parameters 114 to the video decoder 500. Embodiments of video encoder 300 will be described below in greater detail.

[0016] The video decoder 500 can include an enhancement layer decoder 502 and a base layer decoder 504. The base layer decoder 504 can implement video decoding for High Definition (HD) content, for example, with a codec implementing a Moving Picture Experts Group (MPEG)-2 standard, or the like, and decode the encoded video stream 112 to generate a decoded BT.709 video stream 124. The enhancement layer decoder 502 can implement video decoding for UHDTV content and decode the encoded video stream 112 to generate a decoded UHDTV video stream 122.

[0017] In some embodiments, the enhancement layer decoder 502 can decode at least a portion of the encoded video stream 112 into the prediction residue of the UHDTV video frame. The enhancement layer decoder 502 can generate a same or a similar prediction of the UHDTV image frame that was generated by the video encoder 300 during the encoding process, and then combine the prediction with the prediction residue to generate the decoded UHDTV video stream 122. The enhancement layer decoder 502 can generate the prediction of the UHDTV image frame through motion compensation prediction, intra-frame prediction, or scaled color prediction from a

BT.709 image frame decoded in the base layer decoder 504. Embodiments of video encoder 400 will be described below in greater detail.

[0018] Although Figure 1 shows color prediction-based video coding of an UHD TV video stream and a BT.709 video stream with video encoder 300 and video decoder 500, in some embodiments, any video streams representing different color gamuts can be encoded or decoded with color prediction-based video coding.

[0019] Figure 2 is an example graph 200 illustrating color gamuts supported in a BT.709 video standard and in a UHD TV video standard. Referring to Figure 2, the graph 200 shows a two-dimensional representation of color gamuts in an International Commission on Illumination (CIE) 1931 chrominance xy diagram format. The graph 200 includes a standard observer color gamut 210 to represent a range of colors viewable by a standard human observer as determined by the CIE in 1931. The graph 200 includes a UHD TV color gamut 220 to represent a range of colors supported the UHD TV video standard. The graph 200 includes a BT.709 color gamut 230 to represent a range of colors supported the BT.709 video standard, which is narrower than the UHD TV color gamut 220. The graph also includes a point that represents the color white 240, which is included in the standard observer color gamut 210, the UHD TV color gamut 220, and the BT.709 color gamut 230.

[0020] Figures 3A and 3B are block diagram examples of the video encoder 300 shown in Figure 1. Referring to Figure 3A, the video encoder 300 can include an enhancement layer encoder 302 and a base layer encoder 304. The base layer encoder 304 can include a video input 362 to receive a BT.709 video stream 104 having HD image frames. The base layer encoder 304 can include an encoding prediction loop 364 to encode the BT.709 video stream 104 received from the video input 362, and store the reconstructed frames of the BT.709 video stream in a reference buffer 368. The reference buffer 368 can provide the reconstructed BT.709 image frames back to the encoding prediction loop 364 for use in encoding other portions of the same frame or other frames of the BT.709 video stream 104. The reference buffer 368 can store the image frames encoded by the encoding prediction loop 364. The base layer encoder 304 can include entropy encoding function 366 to perform entropy encoding operations on the encoded-version of the BT.709 video stream from the encoding prediction loop 364 and provide an entropy encoded stream to an output interface 380.

[0021] The enhancement layer encoder 302 can include a video input 310 to receive a UHD TV video stream 102 having UHD TV image frames. The enhancement layer encoder 302 can generate a prediction of the UHD TV image frames and utilize the prediction to generate a prediction residue, for example, a difference between the prediction and the UHD TV image frames determined with a combination function 315. In some embodiments, the combination function 315 can include weighting, such as

linear weighting, to generate the prediction residue from the prediction of the UHDTV image frames. The enhancement layer encoder 302 can transform and quantize the prediction residue with a transform and quantize function 320. An entropy encoding function 330 can encode the output of the transform and quantize function 320, and provide an entropy encoded stream to the output interface 380. The output interface 380 can multiplex the entropy encoded streams from the entropy encoding functions 366 and 330 to generate the encoded video stream 112.

[0022] The enhancement layer encoder 302 can include a color space predictor 400, a motion compensation prediction function 354, and an intra predictor 356, each of which can generate a prediction of the UHDTV image frames. The enhancement layer encoder 302 can include a prediction selection function 350 to select a prediction generated by the color space predictor 400, the motion compensation prediction function 354, and/or the intra predictor 356 to provide to the combination function 315.

[0023] In some embodiments, the motion compensation prediction function 354 and the intra predictor 356 can generate their respective predictions based on UHDTV image frames having previously been encoded and decoded by the enhancement layer encoder 302. For example, after a prediction residue has been transformed and quantized, the transform and quantize function 320 can provide the transformed and quantized prediction residue to a scaling and inverse transform function 322, the result of which can be combined in a combination function 325 with the prediction utilized to generate the prediction residue and generate a decoded UHDTV image frame. The combination function 325 can provide the decoded UHDTV image frame to a deblocking function 351, and the deblocking function 351 can store the decoded UHDTV image frame in a reference buffer 340, which holds the decoded UHDTV image frame for use by the motion compensation prediction function 354 and the intra predictor 356. In some embodiments, the deblocking function 351 can filter the decoded UHDTV image frame, for example, to smooth sharp edges in the image between macroblocks corresponding to the decoded UHDTV image frame.

[0024] The motion compensation prediction function 354 can receive one or more decoded UHDTV image frames from the reference buffer 340. The motion compensation prediction function 354 can generate a prediction of a current UHDTV image frame based on image motion between the one or more decoded UHDTV image frames from the reference buffer 340 and the UHDTV image frame.

[0025] The intra predictor 356 can receive a first portion of a current UHDTV image frame from the reference buffer 340. The intra predictor 356 can generate a prediction corresponding to a first portion of a current UHDTV image frame based on at least a second portion of the current UHDTV image frame having previously been encoded and decoded by the enhancement layer encoder 302.

[0026] The color space predictor 400 can generate a prediction of the UHDTV image frames based on BT.709 image frames having previously been encoded by the base layer encoder 304. In some embodiments, the reference buffer 368 in the base layer encoder 304 can provide the reconstructed BT.709 image frame to a resolution upscaling function 370, which can scale the resolution of the reconstructed BT.709 image frame to a resolution that corresponds to the UHDTV video stream 102. The resolution upscaling function 370 can provide an upscaled resolution version of the reconstructed BT.709 image frame to the color space predictor 400. The color space predictor can generate a prediction of the UHDTV image frame based on the upscaled resolution version of the reconstructed BT.709 image frame. In some embodiments, the color space predictor 400 can scale a YUV color space of the upscaled resolution version of the reconstructed BT.709 image frame to correspond to the YUV representation supported by the UHDTV video stream 102.

[0027] There are several ways for the color space predictor 400 to scale the color space supported by BT.709 video coding standard to a color space supported by the UHDTV video stream 102, such as independent channel prediction and affine mixed channel prediction. Independent channel prediction can include converting each portion of the YUV color space for the BT.709 image frame separately into the prediction of the UHDTV image frame. The Y portion or luminance can be scaled according to Equation 1:

$$Y_{UHDTV} = g_1 \cdot Y_{BT.709} + o_1$$

[0028] The U portion or one of the chrominance portions can be scaled according to Equation 2:

$$U_{UHDTV} = g_2 \cdot U_{BT.709} + o_2$$

[0029] The V portion or one of the chrominance portions can be scaled according to Equation 3:

$$V_{UHDTV} = g_3 \cdot V_{BT.709} + o_3$$

[0030] The gain parameters g_1 , g_2 , and g_3 and the offset parameters o_1 , o_2 , and o_3 can be based on differences in the color space supported by the BT.709 video coding standard and the UHDTV video standard, and may vary depending on the content of the respective BT.709 image frame and UHDTV image frame. The enhancement layer encoder 304 can output the gain parameters g_1 , g_2 , and g_3 and the offset parameters o_1 , o_2 , and o_3 utilized by the color space predictor 400 to generate the prediction of the UHDTV image frame to the video decoder 500 as the color prediction parameters 114, for example, via the output interface 380.

[0031] In some embodiments, the independent channel prediction can include gain pa-

rameters g_1 , g_2 , and g_3 , and zero parameters. The Y portion or luminance can be scaled according to Equation 4:

$$Y_{UHDTV} = g_1 \cdot (Y_{BT.709} - Yzero_{BT.709}) + Yzero_{UHDTV}$$

[0032] The U portion or one of the chrominance portions can be scaled according to Equation 5:

$$U_{UHDTV} = g_2 \cdot (U_{BT.709} - Uzero_{BT.709}) + Uzero_{UHDTV}$$

[0033] The V portion or one of the chrominance portions can be scaled according to Equation 6:

$$V_{UHDTV} = g_3 \cdot (V_{BT.709} - Vzero_{BT.709}) + Vzero_{UHDTV}$$

[0034] The gain parameters g_1 , g_2 , and g_3 can be based on differences in the color space supported by the BT.709 video coding standard and the UHDTV video standard, and may vary depending on the content of the respective BT.709 image frame and UHDTV image frame. The enhancement layer encoder 304 can output the gain parameters g_1 , g_2 , and g_3 utilized by the color space predictor 400 to generate the prediction of the UHDTV image frame to the video decoder 500 as the color prediction parameters 114, for example, via the output interface 380. Since the video decoder 500 can be pre-loaded with the zero parameters, the video encoder 300 can generate and transmit fewer color prediction parameters 114, for example, three instead of six, to the video decoder 500.

[0035] In some embodiments, the zero parameters used in Equations 4-6 can be defined based on the bit-depth of the relevant color space and color channel. For example, in Table 1, the zero parameters can be defined as follows:

$Yzero_{BT.709} = 0$	$Yzero_{UHDTV} = 0$
$Uzero_{BT.709} = (1 \ll bits_{BT.709})$	$Uzero_{UHDTV} = (1 \ll bits_{UHDTV})$
$Vzero_{BT.709} = (1 \ll bits_{BT.709})$	$Vzero_{UHDTV} = (1 \ll bits_{UHDTV})$

Table 1

[0036] The affine mixed channel prediction can include converting the YUV color space for a BT.709 image frame by mixing the YUV channels of the BT.709 image frame to generate a prediction of the UHDTV image frame, for example, through a matrix multiplication function. In some embodiments, the color space of the BT.709 can be scaled according to Equation 7:

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix}_{UHDTV} = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{pmatrix} \cdot \begin{pmatrix} Y \\ U \\ V \end{pmatrix}_{BT.709} + \begin{pmatrix} o_1 \\ o_2 \\ o_3 \end{pmatrix}$$

[0037] The matrix parameters m_{11} , m_{12} , m_{13} , m_{21} , m_{22} , m_{23} , m_{31} , m_{32} , and m_{33} and the offset parameters o_1 , o_2 , and o_3 can be based on the difference in color space supported by the BT.709 video format recommendation and the UHDTV video format recommendation, and may vary depending on the content of the respective BT.709 image frame and UHDTV image frame. The enhancement layer encoder 304 can output the matrix and offset parameters utilized by the color space predictor 400 to generate the prediction of the UHDTV image frame to the video decoder 500 as the color prediction parameters 114, for example, via the output interface 380.

[0038] In some embodiments, the color space of the BT.709 can be scaled according to Equation 8:

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix}_{UHDTV} = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{33} \end{pmatrix} \cdot \begin{pmatrix} Y \\ U \\ V \end{pmatrix}_{BT.709} + \begin{pmatrix} o_1 \\ o_2 \\ o_3 \end{pmatrix}$$

[0039] The matrix parameters m_{11} , m_{12} , m_{13} , m_{22} , and m_{33} and the offset parameters o_1 , o_2 , and o_3 can be based on the difference in color space supported by the BT.709 video coding standard and the UHDTV video standard, and may vary depending on the content of the respective BT.709 image frame and UHDTV image frame. The enhancement layer encoder 304 can output the matrix and offset parameters utilized by the color space predictor 400 to generate the prediction of the UHDTV image frame to the video decoder 500 as the color prediction parameters 114, for example, via the output interface 380.

[0040] By replacing the matrix parameters m_{21} , m_{23} , m_{31} , and m_{32} with zero, the luminance channel Y of the UHDTV image frame prediction can be mixed with the color channels U and V of the BT.709 image frame, but the color channels U and V of the UHDTV image frame prediction may not be mixed with the luminance channel Y of the BT.709 image frame. The selective channel mixing can allow for a more accurate prediction of the luminance channel UHDTV image frame prediction, while reducing a number of prediction parameters 114 to transmit to the video decoder 500.

[0041] In some embodiments, the color space of the BT.709 can be scaled according to Equation 9:

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix}_{UHDTV} = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ 0 & m_{22} & m_{23} \\ 0 & m_{32} & m_{33} \end{pmatrix} \cdot \begin{pmatrix} Y \\ U \\ V \end{pmatrix}_{BT.709} + \begin{pmatrix} o_1 \\ o_2 \\ o_3 \end{pmatrix}$$

- [0042] The matrix parameters m11, m12, m13, m22, m23, m32, and m33 and the offset parameters o1, o2, and o3 can be based on the difference in color space supported by the BT.709 video standard and the UHDTV video standard, and may vary depending on the content of the respective BT.709 image frame and UHDTV image frame. The enhancement layer encoder 304 can output the matrix and offset parameters utilized by the color space predictor 400 to generate the prediction of the UHDTV image frame to the video decoder 500 as the color prediction parameters 114, for example, via the output interface 380.
- [0043] By replacing the matrix parameters m21 and m31 with zero, the luminance channel Y of the UHDTV image frame prediction can be mixed with the color channels U and V of the BT.709 image frame. The U and V color channels of the UHDTV image frame prediction can be mixed with the U and V color channels of the BT.709 image frame, but not the luminance channel Y of the BT.709 image frame. The selective channel mixing can allow for a more accurate prediction of the luminance channel UHDTV image frame prediction, while reducing a number of prediction parameters 114 to transmit to the video decoder 500.
- [0044] The color space predictor 400 can generate the scaled color space predictions for the prediction selection function 350 on a per sequence (inter-frame), a per frame, or a per slice (intra-frame) basis, and the video encoder 300 can transmit the prediction parameter 114 corresponding to the scaled color space predictions on a per sequence (inter-frame), a per frame, or a per slice (intra-frame) basis. In some embodiments, the granularity for generating the scaled color space predictions can be preset or fixed in the color space predictor 400 or dynamically adjustable by the video encoder 300 based on encoding function or the content of the UHDTV image frames.
- [0045] The video encoder 300 can transmit the color prediction parameters 114 in a normative portion of the encoded video stream 112, for example, in a Sequence Parameter Set (SPS), a Picture Parameter Set (PPS), or another lower level section of the normative portion of the encoded video stream 112. In some embodiments, the color prediction parameters 114 can be inserted into the encoded video stream 112 with a syntax that allows the video decoder 500 to identify that the color prediction parameters 114 are present in the encoded video stream 112, to identify a precision or size of the parameters, such as a number of bits utilized to represent each parameter, and identify a type of color space prediction the color space predictor 400 of the video encoder 300 utilized to generate the color space prediction.
- [0046] In some embodiments, the normative portion of the encoded video stream 112 can include a flag (use_color_space_prediction), for example, one or more bits, which can announce an inclusion of color space parameters 114 in the encoded video stream 112. The normative portion of the encoded video stream 112 can include a size

parameter (color__predictor_num_fraction_bits_minus1), for example, one or more bits, which can identify a number of bits or precision utilized to represent each parameter. The normative portion of the encoded video stream 112 can include a predictor type parameter (color__predictor_idc), for example, one or more bits, which can identify a type of color space prediction utilized by the video encoder 300 to generate the color space prediction. The types of color space prediction can include independent channel prediction, affine prediction, their various implementations, or the like. The color prediction parameters 114 can include gain parameters, offset parameters, and/or matrix parameters depending on the type of prediction utilized by the video encoder 300.

[0047] Referring to Figure 3B, a video encoder 301 can be similar to video encoder 300 shown and described above in Figure 3A with the following differences. The video encoder 301 can switch the color space predictor 400 with the resolution upscaling function 370. The color space predictor 400 can generate a prediction of the UHDTV image frames based on BT.709 image frames having previously been encoded by the base layer encoder 304.

[0048] In some embodiments, the reference buffer 368 in the base layer encoder 304 can provide the encoded BT.709 image frame to the color space predictor 400. The color space predictor can scale a YUV color space of the encoded BT.709 image frame to correspond to the YUV representation supported by the UHDTV video format. The color space predictor 400 can provide the color space prediction to a resolution upscaling function 370, which can scale the resolution of the color space prediction of the encoded BT.709 image frame to a resolution that corresponds to the UHDTV video format. The resolution upscaling function 370 can provide a resolution upscaled color space prediction to the prediction selection function 350.

[0049] Figure 4 is a block diagram example of the color space predictor 400 shown in Figure 3A. Referring to Figure 4, the color space predictor 400 can include a color space prediction control device 410 to receive a reconstructed BT.709 video frame 402, for example, from a base layer encoder 304 via a resolution upscaling function 370, and select a prediction type and timing for a generation for a color space prediction 406. In some embodiments, the color space prediction control device 410 can pass the reconstructed BT.709 video frame 402 to at least one of an independent channel prediction function 420, an affine prediction function 430, or a cross-color prediction function 440. Each of the prediction functions 420, 430, and 440 can generate a color space prediction of a UHDTV image frame (or portion thereof) from the reconstructed BT.709 video frame 402, for example, by scaling the color space of a BT.709 image frame to a color space of the UHDTV image frame.

[0050] The independent color channel prediction function 420 can scale YUV components

of the encoded BT.709 video stream 402 separately, for example, as shown above in Equations 1-6. The affine prediction function 430 can scale YUV components of the reconstructed BT.709 video frame 402 with a matrix multiplication, for example, as shown above in Equation 7. The cross-color prediction function 440 can scale YUV components of the encoded BT.709 video stream 402 with a modified matrix multiplication that can eliminate mixing of a Y component from the encoded BT.709 video stream 402 when generating the U and V components of the UHDTV image frame, for example, as shown above in Equations 8 or 9.

[0051] In some embodiments, the color space predictor 400 can include a selection device 450 to select an output from the independent color channel prediction function 420, the affine prediction function 430, and the cross-color prediction function 440. The selection device 450 also can output the color prediction parameters 114 utilized to generate the color space prediction 406. The color prediction control device 410 can control the timing of the generation of the color space prediction 406 and the type of operation performed to generate the color space prediction 406, for example, by controlling the timing and output of the selection device 450. In some embodiments, the color prediction control device 410 can control the timing of the generation of the color space prediction 406 and the type of operation performed to generate the color space prediction 406 by selectively providing the encoded BT.709 video stream 402 to at least one of the independent color channel prediction function 420, the affine prediction function 430, and the cross-color prediction function 440.

[0052] Figures 5A and 5B are block diagram examples of the video decoder 500 shown in Figure 1. Referring to Figure 5A, the video decoder can include an interface 510 to receive the encoded video stream 112, for example, from a video encoder 300. The interface 510 can demultiplex the encoded video stream 112 and provide encoded UHDTV image data to an enhancement layer decoder 502 of the video decoder 500 and provide encoded BT.709 image data to a base layer decoder 504 of the video decoder 500. The base layer decoder 504 can include an entropy decoding function 552 and a decoding prediction loop 554 to decode encoded BT.709 image data received from the interface 510, and store the decoded BT.709 video stream 124 in a reference buffer 556. The reference buffer 556 can provide the decoded BT.709 video stream 124 back to the decoding prediction loop 554 for use in decoding other portions of the same frame or other frames of the encoded BT.709 image data. The base layer decoder 504 can output the decoded BT.709 video stream 124. In some embodiments, the output from the decoding prediction loop 554 and input to the reference buffer 556 may be residual frame data rather than the reconstructed frame data.

[0053] The enhancement layer decoder 502 can include an entropy decoding function 522, an inverse quantization function 524, an inverse transform function 526, and a com-

combination function 528 to decode the encoded UHDTV image data received from the interface 510. A deblocking function 541 can filter the decoded UHDTV image frame, for example, to smooth sharp edges in the image between macroblocks corresponding to the decoded UHDTV image frame, and store the decoded UHDTV video stream 122 in a reference buffer 530. In some embodiments, the encoded UHDTV image data can correspond to a prediction residue, for example, a difference between a prediction and a UHDTV image frame as determined by the video encoder 300. The enhancement layer decoder 502 can generate a prediction of the UHDTV image frame, and the combination function 528 can add the prediction of the UHDTV image frame to encoded UHDTV image data having undergone entropy decoding, inverse quantization, and an inverse transform to generate the decoded UHDTV video stream 122. In some embodiments, the combination function 528 can include weighting, such as linear weighting, to generate the decoded UHDTV video stream 122.

- [0054] The enhancement layer decoder 502 can include a color space predictor 600, a motion compensation prediction function 542, and an intra predictor 544, each of which can generate the prediction of the UHDTV image frame. The enhancement layer decoder 502 can include a prediction selection function 540 to select a prediction generated by the color space predictor 600, the motion compensation prediction function 542, and/or the intra predictor 544 to provide to the combination function 528.
- [0055] In some embodiments, the motion compensation prediction function 542 and the intra predictor 544 can generate their respective predictions based on UHDTV image frames having previously been decoded by the enhancement layer decoder 502 and stored in the reference buffer 530. The motion compensation prediction function 542 can receive one or more decoded UHDTV image frames from the reference buffer 530. The motion compensation prediction function 542 can generate a prediction of a current UHDTV image frame based on image motion between the one or more decoded UHDTV image frames from the reference buffer 530 and the UHDTV image frame.
- [0056] The intra predictor 544 can receive a first portion of a current UHDTV image frame from the reference buffer 530. The intra predictor 544 can generate a prediction corresponding to a first portion of a current UHDTV image frame based on at least a second portion of the current UHDTV image frame having previously been decoded by the enhancement layer decoder 502.
- [0057] The color space predictor 600 can generate a prediction of the UHDTV image frames based on BT.709 image frames decoded by the base layer decoder 504. In some embodiments, the reference buffer 556 in the base layer decoder 504 can provide a portion of the decoded BT.709 video stream 124 to a resolution upscaling function 570, which can scale the resolution of the encoded BT.709 image frame to a resolution

that corresponds to the UHDTV video format. The resolution upscaling function 570 can provide an upscaled resolution version of the encoded BT.709 image frame to the color space predictor 600. The color space predictor can generate a prediction of the UHDTV image frame based on the upscaled resolution version of the encoded BT.709 image frame. In some embodiments, the color space predictor 600 can scale a YUV color space of the upscaled resolution version of the encoded BT.709 image frame to correspond to the YUV representation supported by the UHDTV video format.

[0058] The color space predictor 600 can operate similarly to the color space predictor 400 in the video encoder 300, by scaling the color space supported by BT.709 video coding standard to a color space supported by the UHDTV video format, for example, with independent channel prediction, affine mixed channel prediction, or cross-color channel prediction. The color space predictor 600, however, can select a type of color space prediction to generate based, at least in part, on the color prediction parameters 114 received from the video encoder 300. The color prediction parameters 114 can explicitly identify a particular a type of color space prediction, or can implicitly identify the type of color space prediction, for example, by a quantity and/or arrangement of the color prediction parameters 114.

[0059] As discussed above, in some embodiments, the normative portion of the encoded video stream 112 can include a flag (use_color_space_prediction), for example, one or more bits, which can announce an inclusion of color space parameters 114 in the encoded video stream 112. The normative portion of the encoded video stream 112 can include a size parameter (color__predictor_num_fraction_bits_minus1), for example, one or more bits, which can identify a number of bits or precision utilized to represent each parameter. The normative portion of the encoded video stream 112 can include a predictor type parameter (color__predictor_idc), for example, one or more bits, which can identify a type of color space prediction utilized by the video encoder 300 to generate the color space prediction. The types of color space prediction can include independent channel prediction, affine prediction, their various implementations, or the like. The color prediction parameters 114 can include gain parameters, offset parameters, and/or matrix parameters depending on the type of prediction utilized by the video encoder 300.

[0060] The color space predictor 600 identify whether the video encoder 300 utilize color space prediction in generating then encoded video stream 112 based on the flag (use_color_space_prediction). When color prediction parameters 114 are present in the encoded video stream 112, the color space predictor 600 can parse the color prediction parameters 114 to identify a type of color space prediction utilized by the video encoded based on the predictor type parameter (color__predictor_idc), and a size or precision of the parameters (color__predictor_num_fraction_bits_minus1), and locate

the color space parameters to utilize to generate a color space prediction.

[0061] For example, the video decoder 500 can determine whether the color prediction parameters 114 are present in the encoded video stream 112 and parse the color prediction parameters 114 based on the following example code in Table 2:

```

use_color_space_prediction
if(use_color_space_prediction) {
    color_predictor_num_fraction_bits_minus1
    color_prediction_idc
    if(color_prediction_idc==0) {
        for( i = 0; i < 3; i++){
            color_predictor_gain [ i ]
        }
    }
    if(color_prediction_idc==1) {
        for( i = 0; i < 3; i++){
            color_predictor_gain [ i ]
            color_predictor_offset [ i ]
        }
    }
    if(color_prediction_idc==2) {
        for( i = 0; i < 3; i++){
            for( j= 0; j < 3; j++){
                cross_color_predictor_gain [ i ][j]
            }
            color_predictor_offset [ i ]
        }
    }
}

```

Table 2

[0062] The example code in Table 2 can allow the video decoder 500 to identify whether color prediction parameters 114 are present in the encoded video stream 112 based on the use_color_space_prediction flag. The video decoder 500 can identify the precision or size of the color space parameters based on the size parameter (color__predictor_num_fraction_bits_minus1), and can identify a type of color space prediction utilized by the video encoder 300 based on the type parameter (color__predictor_idc). The example code in Table 2 can allow the video decoder 500 to parse the color space parameters from the encoded video stream 112 based on the identified size of the color space parameters and the identified type color space prediction utilized by the video encoder 300, which can identify the number, semantics, and location of the color space parameters. Although the example code in Table 2 shows the affine prediction including 9 matrix parameters and 3 offset parameters, in some embodiments, the color prediction parameters 114 can include fewer matrix and/or offset parameters, for example, when the matrix parameters are zero, and the example code can be modified to parse the color prediction parameters 114 accordingly.

[0063] The color space predictor 600 can generate color space predictions for the prediction selection function 540 on a per sequence (inter-frame), a per frame, or a per slice (intra-frame) basis. In some embodiments, the color space predictor 600 can generate

the color space predictions with a fixed or preset timing or dynamically in response to a reception of the color prediction parameters 114 from the video encoder 300.

- [0064] Referring to Figure 5B, a video decoder 501 can be similar to video decoder 500 shown and described above in Figure 5A with the following differences. The video decoder 501 can switch the color space predictor 600 with the resolution upscaling function 570. The color space predictor 600 can generate a prediction of the UHDTV image frames based on portions of the decoded BT.709 video stream 124 from the base layer decoder 504.
- [0065] In some embodiments, the reference buffer 556 in the base layer decoder 504 can provide the portions of the decoded BT.709 video stream 124 to the color space predictor 600. The color space predictor 600 can scale a YUV color space of the portions of the decoded BT.709 video stream 124 to correspond to the YUV representation supported by the UHDTV video standard. The color space predictor 600 can provide the color space prediction to a resolution upscaling function 570, which can scale the resolution of the color space prediction to a resolution that corresponds to the UHDTV video standard. The resolution upscaling function 570 can provide a resolution upscaled color space prediction to the prediction selection function 540.
- [0066] Figure 6 is a block diagram example of a color space predictor 600 shown in Figure 5A. Referring to Figure 6, the color space predictor 600 can include a color space prediction control device 610 to receive the decoded BT.709 video stream 122, for example, from a base layer decoder 504 via a resolution upscaling function 570, and select a prediction type and timing for a generation for a color space prediction 606. The color space predictor 600 can select a type of color space prediction to generate based, at least in part, on the color prediction parameters 114 received from the video encoder 300. The color prediction parameters 114 can explicitly identify a particular a type of color space prediction, or can implicitly identify the type of color space prediction, for example, by a quantity and/or arrangement of the color prediction parameters 114. In some embodiments, the color space prediction control device 610 can pass the decoded BT.709 video stream 122 and color prediction parameters 114 to at least one of an independent channel prediction function 620, an affine prediction function 630, or a cross-color prediction function 640. Each of the prediction functions 620, 630, and 640 can generate a color space prediction of a UHDTV image frame (or portion thereof) from the decoded BT.709 video stream 122, for example, by scaling the color space of a BT.709 image frame to a color space of the UHDTV image frame based on the color space parameters 114.
- [0067] The independent color channel prediction function 620 can scale YUV components of the decoded BT.709 video stream 122 separately, for example, as shown above in Equations 1-6. The affine prediction function 630 can scale YUV components of the

decoded BT.709 video stream 122 with a matrix multiplication, for example, as shown above in Equation 7. The cross-color prediction function 640 can scale YUV components of the decoded BT.709 video stream 122 with a modified matrix multiplication that can eliminate mixing of a Y component from the decoded BT.709 video stream 122 when generating the U and V components of the UHDTV image frame, for example, as shown above in Equations 8 or 9.

- [0068] In some embodiments, the color space predictor 600 can include a selection device 650 to select an output from the independent color channel prediction function 620, the affine prediction function 630, and the cross-color prediction function 640. The color prediction control device 610 can control the timing of the generation of the color space prediction 606 and the type of operation performed to generate the color space prediction 606, for example, by controlling the timing and output of the selection device 650. In some embodiments, the color prediction control device 610 can control the timing of the generation of the color space prediction 606 and the type of operation performed to generate the color space prediction 606 by selectively providing the decoded BT.709 video stream 122 to at least one of the independent color channel prediction function 620, the affine prediction function 630, and the cross-color prediction function 640.
- [0069] Figure 7 is an example operational flowchart for color space prediction in the video encoder 300. Referring to Figure 7, at a first block 710, the video encoder 300 can encode a first image having a first image format. In some embodiments, the first image format can correspond to a BT.709 video standard and the video encoder 300 can include a base layer to encode BT.709 image frames.
- [0070] At a block 720, the video encoder 300 can scale a color space of the first image from the first image format into a color space corresponding to a second image format. In some embodiments, the video encoder 300 can scale the color space between the BT.709 video standard and an Ultra High Definition Television (UHDTV) video standard corresponding to the second image format.
- [0071] There are several ways for the video encoder 300 to scale the color space supported by BT.709 video coding standard to a color space supported by the UHDTV video format, such as independent channel prediction and affine mixed channel prediction. For example, the independent color channel prediction can scale YUV components of encoded BT.709 image frames separately, for example, as shown above in Equations 1-6. The affine mixed channel prediction can scale YUV components of the encoded BT.709 image frames with a matrix multiplication, for example, as shown above in Equations 7-9.
- [0072] In some embodiments, the video encoder 300 can scale a resolution of the first image from the first image format into a resolution corresponding to the second image format.

For example, the UHD TV video standard can support a 4k (3840x2160 pixels) or an 8k (7680x4320 pixels) resolution and a 10 or 12 bit quantization bit-depth. The BT.709 video standard can support a 2k (1920x1080 pixels) resolution and an 8 or 10 bit quantization bit-depth. The video encoder 300 can scale the encoded first image from a resolution corresponding to the BT.709 video standard into a resolution corresponding to the UHD TV video standard.

- [0073] At a block 730, the video encoder 300 can generate a color space prediction based, at least in part, on the scaled color space of the first image. The color space prediction can be a prediction of a UHD TV image frame (or portion thereof) from a color space of a corresponding encoded BT.709 image frame. In some embodiments, the video encoder 300 can generate the color space prediction based, at least in part, on the scaled resolution of the first image.
- [0074] At a block 740, the video encoder 300 can encode a second image having the second image format based, at least in part, on the color space prediction. The video encoder 300 can output the encoded second image and color prediction parameters utilized to scale the color space of the first image to a video decoder.
- [0075] Figure 8 is an example operational flowchart for color space prediction in the video decoder 500. Referring to Figure 8, at a first block 810, the video decoder 500 can decode an encoded video stream to generate a first image having a first image format. In some embodiments, the first image format can correspond to a BT.709 video standard and the video decoder 500 can include a base layer to decode BT.709 image frames.
- [0076] At a block 820, the video decoder 500 can scale a color space of the first image corresponding to the first image format into a color space corresponding to a second image format. In some embodiments, the video decoder 500 can scale the color space between the BT.709 video standard and an Ultra High Definition Television (UHD TV) video standard corresponding to the second image format.
- [0077] There are several ways for the video decoder 500 to scale the color space supported by BT.709 video coding standard to a color space supported by the UHD TV video standard, such as independent channel prediction and affine mixed channel prediction. For example, the independent color channel prediction can scale YUV components of the encoded BT.709 image frames separately, for example, as shown above in Equations 1-6. The affine mixed channel prediction can scale YUV components of the encoded BT.709 image frames with a matrix multiplication, for example, as shown above in Equations 7-9.
- [0078] The video decoder 500 can select a type of color space scaling to perform, such as independent channel prediction or one of the varieties of affine mixed channel prediction based on channel prediction parameters the video decoder 500 receives from the video

encoder 300. In some embodiments, the video decoder 500 can perform a default or preset color space scaling of the decoded BT.709 image frames.

- [0079] In some embodiments, the video decoder 500 can scale a resolution of the first image from the first image format into a resolution corresponding to the second image format. For example, the UHD TV video standard can support a 4k (3840x2160 pixels) or an 8k (7680x4320 pixels) resolution and a 10 or 12 bit quantization bit-depth. The BT.709 video standard can support a 2k (1920x1080 pixels) resolution and an 8 or 10 bit quantization bit-depth. The video decoder 500 can scale the decoded first image from a resolution corresponding to the BT.709 video standard into a resolution corresponding to the UHD TV video standard.
- [0080] At a block 830, the video decoder 500 can generate a color space prediction based, at least in part, on the scaled color space of the first image. The color space prediction can be a prediction of a UHD TV image frame (or portion thereof) from a color space of a corresponding decoded BT.709 image frame. In some embodiments, the video decoder 500 can generate the color space prediction based, at least in part, on the scaled resolution of the first image.
- [0081] At a block 840, the video decoder 500 can decode the encoded video stream into a second image having the second image format based, at least in part, on the color space prediction. In some embodiments, the video decoder 500 can utilize the color space prediction to combine with a portion of the encoded video stream corresponding to a prediction residue from the video encoder 300. The combination of the color space prediction and the decoded prediction residue can correspond to a decoded UHD TV image frame or portion thereof.
- [0082] Figure 9 is another example operational flowchart for color space prediction in the video decoder 500. Referring to Figure 9, at a first block 910, the video decoder 500 can decode at least a portion of an encoded video stream to generate a first residual frame having a first format. The first residual frame can be a frame of data corresponding to a difference between two image frames. In some embodiments, the first format can correspond to a BT.709 video standard and the video decoder 500 can include a base layer to decode BT.709 image frames.
- [0083] At a block 920, the video decoder 500 can scale a color space of the first residual frame corresponding to the first format into a color space corresponding to a second format. In some embodiments, the video decoder 500 can scale the color space between the BT.709 video standard and an Ultra High Definition Television (UHD TV) video standard corresponding to the second format.
- [0084] There are several ways for the video decoder 500 to scale the color space supported by BT.709 video coding standard to a color space supported by the UHD TV video standard, such as independent channel prediction and affine mixed channel prediction.

For example, the independent color channel prediction can scale YUV components of the encoded BT.709 image frames separately, for example, as shown above in Equations 1-6. The affine mixed channel prediction can scale YUV components of the encoded BT.709 image frames with a matrix multiplication, for example, as shown above in Equations 7-9.

- [0085] The video decoder 500 can select a type of color space scaling to perform, such as independent channel prediction or one of the varieties of affine mixed channel prediction based on channel prediction parameters the video decoder 500 receives from the video encoder 300. In some embodiments, the video decoder 500 can perform a default or preset color space scaling of the decoded BT.709 image frames.
- [0086] In some embodiments, the video decoder 500 can scale a resolution of the first residual frame from the first format into a resolution corresponding to the second format. For example, the UHD TV video standard can support a 4k (3840x2160 pixels) or an 8k (7680x4320 pixels) resolution and a 10 or 12 bit quantization bit-depth. The BT.709 video standard can support a 2k (1920x1080 pixels) resolution and an 8 or 10 bit quantization bit-depth. The video decoder 500 can scale the decoded first residual frame from a resolution corresponding to the BT.709 video standard into a resolution corresponding to the UHD TV video standard.
- [0087] At a block 930, the video decoder 500 can generate a color space prediction based, at least in part, on the scaled color space of the first residual frame. The color space prediction can be a prediction of a UHD TV image frame (or portion thereof) from a color space of a corresponding decoded BT.709 image frame. In some embodiments, the video decoder 500 can generate the color space prediction based, at least in part, on the scaled resolution of the first residual frame.
- [0088] At a block 940, the video decoder 500 can decode the encoded video stream into a second image having the second format based, at least in part, on the color space prediction. In some embodiments, the video decoder 500 can utilize the color space prediction to combine with a portion of the encoded video stream corresponding to a prediction residue from the video encoder 300. The combination of the color space prediction and the decoded prediction residue can correspond to a decoded UHD TV image frame or portion thereof.
- [0089] 'Color Bit Depth Scaling'
- [0090] Color bit depth scaling can provide enhancement of color coding and decoding in video compression, such as High Efficiency Video Coding (HEVC), a video coding standard currently under development and published in draft form, or other video compression systems. The bit depth scaling improves handling of differing color characteristics (e.g., resolution, quantization bit-depth, and color gamut) employed in different digital video formats, such as HD BT.709 and UHD TV BT.2020, for

example, particularly during decoding. The following description is made with reference to HEVC, namely a publicly defined test model of a Scalable HEVC Extension, but is similarly applicable to other analogous video compression systems.

- [0091] Encoders 300 and 301 of Figures 3A and 3B provide encoding of HD and UHDTV videos streams and each includes a color space predictor 400 that can generate a prediction of a UHDTV image frame (or picture) based on the upscaled resolution version of the reconstructed BT.709 image frame (or picture). As described above, the color space predictor 400 in some embodiments can scale a YUV color space of the upscaled resolution version of the reconstructed BT.709 image frame to correspond to the YUV representation supported by the UHDTV video stream 102.
- [0092] Figures 10A and 10B are block diagram examples of video encoders 1000 and 1001 that are analogous to encoders 300 and 301, respectively, and include corresponding elements indicated by the same reference numerals. In addition, encoders 1000 and 1001 each includes a bit depth scaling function 1010, rather than the color space predictor 400, to provide enhanced color bit depth scaling of frames or pictures, including bit depth scaling of reference pictures.
- [0093] Video encoders 1000 and 1001 make reference to reference pictures (or frames), stored in reference buffers 340 and 368, in processing the pictures of a video stream.
- [0094] Figure 11 is a simplified flow diagram of a video encoding method 1100 that includes bit depth scaling as performed by function 1010 and is described with reference to HEVC encoding.
- [0095] With regard to a current picture CurrPic, step 1110 provides a sampling process for picture sample values using as inputs an array rsPicSampleL of luma samples, an array rsPicSampleCb of chroma samples of the component Cb, and an array rsPicSampleCr of chroma samples of the component Cr, and providing as outputs an array rlPicSampleL of luma samples, an array rlPicSampleCb of chroma samples of the component Cb, and an array rlPicSampleCr of chroma samples of the component Cr.
- [0096] Step 1120 provides a sampling process for reference pictures to obtain a sampled inter-layer reference picture rsPic from a video picture input rlPic as input. Step 1120 may be invoked at the beginning of the encoding process for a first P or B slice of a current picture CurrPic.
- [0097] Step 1125 provides a scaling of the bit depth of the inter-layer reference picture.
- [0098] Step 1130 provides encoding of an inter-layer reference picture set to obtain a list of inter-layer pictures, which includes sampling bit depth scaled inter layer reference picture rsbPic. Step 1140 provides encoding of unit tree coding layers. Step 1150 provides encoding of slice segment layers, including encoding processes for each P or B slice and constructing reference picture list for each P or B slice. Step 1160 provides encoding of network abstraction layer (NAL) units, or packets.

- [0099] Decoders 500 and 501 of Figures 5A and 5B provide decoding of encoded video streams that may correspond to HD and UHDTV videos streams. Decoders 500 and 501 and each includes a color space predictor 600 that can generate a prediction of UHDTV image frames (or pictures) based on BT.709 image frames decoded by the base layer decoder 504, as described above.
- [0100] Figures 12A and 12B are block diagram examples of video decoders 1200 and 1201 that are analogous to decoders 500 and 501, respectively, and include corresponding elements indicated by the same reference numerals. In addition, decoders 1200 and 1201 each include a bit depth scaling function 1210, rather than the color space predictor 600 of decoders 500 and 501, to utilize the bit depth scaling of frames or pictures. Video decoders 1200 and 1201 provide decoding of encoded video streams, which include network abstraction layer units (or packets) with slices of coded pictures (or frames). The decoding obtains and utilizes reference pictures and inter-layer reference picture sets to obtain the picture sample values of the successive pictures of a video stream.
- [0101] Figure 13 is a flow diagram of one implementation of a decoding method 1300 that includes bit depth scaling processes as performed by function 1210 and is described with reference to HEVC decoding. With regard to a current picture CurrPic, step 1310 provides decoding of network abstraction layer (NAL) units, or packets. Step 1320 provides decoding with regard to slice segment layers, including decoding processes for each P or B slice and constructing a reference picture list for each P or B slice. Step 1330 provides decoding with regard to unit tree coding layers. Step 1340 provides decoding with regard to an inter-layer reference picture set to obtain a list of inter-layer pictures, which includes deriving a resampled bit depth scaled inter layer reference picture rsbPic.
- [0102] Step 1350 provides a resampling process for reference pictures to obtain a resampled inter-layer reference picture rsPic from a decoded picture rlPic as input. Step 1350 may be invoked at the beginning of the decoding process for a first P or B slice of a current picture CurrPic. Step 1360 provides a resampling process for picture sample values using as inputs an array rlPicSampleL of luma samples, an array rlPicSampleCb of chroma samples of the component Cb, and an array rlPicSampleCr of chroma samples of the component Cr, and providing as outputs an array rsPicSampleL of luma samples, an array rsPicSampleCb of chroma samples of the component Cb, and an array rsPicSampleCr of chroma samples of the component Cr.
- [0103] Steps 1310-1360 generally correspond to conventional HEVC decoding, except for the deriving a resampled bit depth scaled inter layer reference picture rsbPic in step 1340. As novel added steps, method 1300 includes a step 1370 that provides a bit depth scaling process for reference pictures and a step 1380 that provides a bit depth

scaling process for picture sample values

- [0104] Bit depth scaling process for a reference picture of step 1370 operates on the resampled inter layer reference picture rsPic as an input and provides as an output a resampled bit depth scaled inter layer reference picture rsbPic. A benefit of resampled bit depth scaled inter layer reference picture rsbPic is that it accommodates forming inter-layer references from pictures at different bit-depths. Step 1370 uses variables nBdbY and nBdbC, which specify the bit depth of the samples of the luma array and bit depth of the samples of the chroma array of the current picture CurrPic, and variables nBdY and nBdC, which specify the bit depth of the samples of the luma array and bit depth of the samples of the chroma array of the resampled reference layer picture rsPic. Step 1370 derives a resampled bit depth scaled inter layer reference picture rsbPic with bit depth scaling as follows.

if nBdY is equal to nBdbY and nBdC is equal to nBdbC
 - rsbPic is set to rsPic,
 otherwise rsPic is derived by follows:

- [0105] The bit depth scaling of step 1380 is invoked with the resampled sample values of rsPicSample as input, and with the resampled bit depth scaled sample values of rsbPicSample as output. Bit depth scaling process for picture sample values of step 1380 operates on inputs:

(ScaledW)x(ScaledH) array rsPicSampleL of luma samples with bit depth nBdY,

(ScaledW/2)x(ScaledH/2) array rsPicSampleCb of chroma samples of the component Cb with bit depth nBdC, and

(ScaledW/2)x(ScaledH/2) array rsPicSampleCr of chroma samples of the component Cr with bit depth nBdC

- [0106] and provides as outputs:

(ScaledW)x(ScaledH) array rsbPicSampleL of luma samples with bit depth nBdbYI,

(ScaledW/2)x(ScaledH/2) array rsbPicSampleCb of chroma samples of the component Cb with bit depth nBdbCI, and

(ScaledW/2)x(ScaledH/2) array rsbPicSampleCr of chroma samples of the component Cr with bit depth nBdbC.

- [0107] These output arrays correspond to reference pictures used for encoding the enhancement layer pictures. A benefit of bit-depth scaling of picture samples is accommodating prediction between pictures having samples that are at different bit-depths.

- [0108] Bit depth scaling process for picture sample values of step 1380 operates as follows. For each luma sample location (xP = 0 .. ScaledW-1, yP = 0 .. ScaledH-1) in the luma

sample array rsPicSampleL1, the corresponding luma sample value is derived as:

$$\text{rsbPicSampleL}[xP, yP] = \text{rsPicSampleL}[xP, yP] \ll (\text{nBdbY} - \text{nBdY}).$$

- [0109] For each chroma sample location ($xP = 0 \dots \text{ScaledW}/2 - 1$, $yP = 0 \dots \text{ScaledH}/2 - 1$) in the chroma sample array for the component Cb rsPicSampleCb, the corresponding chroma sample value is derived as

$$\text{rsbPicSampleCb}[xP, yP] = \text{rsPicSampleCb}[xP, yP] \ll (\text{nBdbC} - \text{nBdC})$$

- [0110] For each chroma sample location ($xP = 0 \dots \text{ScaledW}/2 - 1$, $yP = 0 \dots \text{ScaledH}/2 - 1$) in the chroma sample array for the component Cr rsPicSampleCr, the corresponding chroma sample value is derived as:

$$\text{rsbPicSampleCr}[xP, yP] = \text{rsPicSampleCr}[xP, yP] \ll (\text{nBdbC} - \text{nBdC}).$$

- [0111] These equations compensate the reference picture for the sample bit-depth difference between the base and enhancement layers.

- [0112] It will be appreciated that the bit depth scaling described above may be implemented in various alternative embodiments. For example, the bit depth variables used in steps 1370 and 1380 could be used to generate the color gamut scalable (CGS) enhancement layer. In one implementation, the bit depth scaling could require that motion compensation for the color gamut scalable (CGS) enhancement layer picture take place by using weighted prediction by utilizing uni-prediction as with the predictor being a base layer picture (e.g., re-sampled and bit depth scaled). A benefit of this implementation is that weighted prediction process defined in existing HEVC base specification could be utilized to perform color space prediction.

- [0113] In another embodiment, whenever a layer i is a CGS enhancement layer, a $\text{direct_dependency_flag}[i][i-1]$ could be set equal to 1 and a $\text{direct_dependency_flag}[i][j]$ could be equal to 0 for $j < i-1$. This means that only a layer with index $i-1$ may be a direct reference layer for the layer with index i , thereby operating to constrain layer dependency signaling when using this color gamut scalable coding. A benefit of constraining layer dependency signaling is that reference picture list is simplified. As another alternative, whenever the layer i is a CGS enhancement layer, then:

$$\sum_{j=0}^{i-1} \text{direct_dependency_flag}[i][j] == 1.$$

- [0114] As a result, layer with index i may have only one direct reference layer from other layers. A benefit of constraining layer dependency signaling is that reference picture list is simplified.

- [0115] In another implementation, the decoding process for each slice for the CGS en-

hancement layer picture can begin with deriving as follows a reference picture list RefPicList0 with regard to a variable NumRpsCurrTempList0, which refers to the number of entries in a temporary reference picture list - RefPicListTemp0 - which is later used to create the list RefPicList0 :

Set NumRpsCurrTempList0 equal to
 Max(num_ref_idx_l0_active_minus1 + 1, NumPocTotalCurr),

[0116] in which num_ref_idx_l0_active_minus1 + 1 and NumPocTotalCurr are temporary variables, respectively, and then construct the list RefPicList0 as follows.

for(rIdx = 0; rIdx <= num_ref_idx_l0_active_minus1 ; rIdx++)

RefPicList0[rIdx] = ref_pic_list_modification_flag_l0 ?
 RefPicSetInterLayer [list_entry_l0[rIdx]] : RefPicSetInterLayer [rIdx]

[0117] It could also be a requirement that when the layer i is a CGS enhancement layer, num_ref_idx_l0_active_minus1 may be equal to 0.

[0118] Video compression systems such as HEVC, and the predecessor video compression standard H.264/MPEG-4 AVC, employ a video parameter set (VPS) structure in which video parameter sets, including extensions of video parameter sets, contain information that can be used to decode several regions of encoded video. For example, current HEVC includes a syntax for extending video parameter sets under vps_extension(), as set forth in Table 3:

vps_extension() {	Descriptor
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
ave_base_layer_flag	u(1)
splitting_flag	u(1)
for(i = 0, NumScalabilityTypes = 0; i < 16; i++) {	
scalability_mask[i]	u(1)
NumScalabilityTypes += scalability_mask[i]	
}	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id_len_minus1 [j]	
...	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	
}	

Table 3

[0119] Conventional video parameter sets under vps_extension() in HEVC, as set forth in Table 3, provide only limited characterization of color characteristics of an encoded video format. In contrast, an expanded vps_extension() set forth in Table 4 includes specific attributes regarding the color characteristics of an encoded video format, thereby signaling color gamut scalability and bit depth information regarding enhancement layers in the vps extension. The information about bit depth of luma and chroma components of each layer and about chromaticity coordinates of the source primaries of each layer can be useful for session negotiation in allowing end devices to select layers to decode based on their bit depth and color support capability.

vps_extension() {	Descriptor
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
ave_base_layer_flag	u(1)
splitting_flag	u(1)
for(i = 0, NumScalabilityTypes = 0; i < 16; i++) {	
scalability_mask[i]	u(1)
NumScalabilityTypes += scalability_mask[i]	
}	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension- id_ len_minus1 [j]	u(1)
...	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
bitdepth_colorgamut_info(i)	
}	
bitdepth_colorgamut_info(id){	
bit depth layer luma minus8[id]	ue(v)
bit depth layer chroma minus8[id]	ue(v)
layer color _gamut[id]	u(1)
}	

Table 4

[0120] The an expanded vps_extension() set includes the attributes:

[0121] 'bit_depth_layer_luma_minus8[id]+8' which may specify the bit depth of the samples of the luminance (sometimes referred to as "luma") array for the layer with layer id id, as specified by:

$$\text{BitDepthLy}[\text{id}] = 8 + \text{bit_depth_layer_luma_minus8}[\text{id}],$$

- [0122] with `bit_depth_layer_luma_minus8` in the range of 0 to 6, inclusive, according to or indicating the bit-depth of the luma component of the video in the range 8 to 14.
- [0123] '`bit_depth_layer_chroma_minus8[id] + 8`' which may specify the bit depth of the samples of the chrominance (sometimes referred to as "chroma") arrays for the layer with layer id `id`, as specified by:
- $$\text{BitDepthLc}[id] = 8 + \text{bit_depth_layer_chroma_minus8}[id],$$
- [0124] with `bit_depth_layer_chroma_minus8` in the range of 0 to 6, inclusive, according to or indicating the bit-depth of the chroma components of the video in the range 8 to 14.
- [0125] '`layer_color_gamut[id]`' is set equal to 1 to specify that the chromaticity coordinates of the source primaries for layer id are defined as per Rec. ITU-R BT.2020, and `layer_color_gamut[id]` is set equal to 0 to specify that the chromaticity coordinates of the source primaries for layer id are defined as per Rec. ITU-R BT.709.
- [0126] In an alternative embodiment, separate bit depth may be signaled for chroma components Cb and Cr. In another alternative embodiment, the `bitdepth_colorgamut_info()` could also be signaled for the base layer. In this case the for loop index in the `vps_extension` can start from `i=0` instead of `i=1`. In still another alternative embodiment, color primaries other than BT.709 and BT.2020 may be indicated such as, for example, by a syntax element similar to `colour_primaries` syntax element signalled in video usability information (VUI) of HEVC draft specification could be signaled for each layer to indicate its color primary.
- [0127] Picture parameter sets ("PPS") carry data valid on a picture by picture basis. Accordingly, the PPS is a syntax structure containing syntax elements that apply to zero or more entire coded pictures as determined by a syntax element, such as that found in each slice segment header.
- [0128] Sequence parameter sets ("SPS") may be used to carry data valid for an entire video sequence. Accordingly, the SPS is a syntax structure containing syntax elements that apply to zero or more entire coded video sequences ("CVS") as determined by the content of a syntax element found in the PPS referred to by a syntax element, such as that found in each slice segment header.
- [0129] Video parameter sets ("VPS") may be used to carry data valid for an entire video sequence. Accordingly, the VPS is a syntax structure containing syntax elements that apply to zero or more entire coded video sequences as determined by the content of a syntax element found in the SPS referred to by a syntax element found in the PPS referred to by a syntax element found in each slice segment header.
- [0130] A modified syntax for video parameter set extension (`vps_extension()`) which is signaled inside a video parameter set VPS is shown in Table 5.

vps_extension() {	Descriptor
...	
vps_num_rep_formats_minus1	u(4)
for(i = 0; i <= vps_num_rep_formats_minus1; i++)	
rep_format()	
for(i = 1; i <= vps_max_layers_minus1; i++)	
if(vps_num_rep_formats_minus1 > 0)	
vps_rep_format_idx[i]	u(4)
max_one_active_ref_layer_flag	u(1)
...	
}	

Table 5

- [0131] **max_one_active_ref_layer_flag** is signaled in VPS extension.
max_one_active_ref_layer_flag equal to 1 may specify that at most one picture is used for inter-layer prediction for each picture in the CVS. **max_one_active_ref_layer_flag** equal to 0 may specify that more than one picture may be used for inter-layer prediction for each picture in the CVS.
- [0132] **nuh_layer_id** may specify the identifier of the layer.
- [0133] **vps_max_layers_minus1** plus 1 may specify the maximum number of layers that may be present in the CVS, wherein a layer may e.g. be a spatial scalable layer, a quality scalable layer, a texture view or a depth view.
- [0134] **layer_id_in_nuh[i]** may specify the value of the **nuh_layer_id** syntax element in VCL NAL units of the i-th layer. For i in a range from 0 to **vps_max_layers_minus1**, inclusive, when not present, the value of **layer_id_in_nuh[i]** may be inferred to be equal to i. When i is greater than 0, **layer_id_in_nuh[i]** may be greater than **layer_id_in_nuh[i - 1]**. For i in a range from 0 to **vps_max_layers_minus1**, inclusive, the variable **LayerIdxInVps[layer_id_in_nuh[i]]** may be set equal to i.
- [0135] **vps_num_rep_formats_minus1** plus 1 may specify the number of the following **rep_format()** syntax structures in the VPS. The first report format syntax structure **rep_format()** with i=0 provides information applicable to layer with **nuh_layer_id** equal to 0. It is a requirement of bitstream conformance that the values of **rep_format()** for i=0 for **chroma_format_vps_idc**, **separate_colour_plane_vps_flag**, **pic_width_vps_in_luma_samples**, **pic_height_vps_in_luma_samples**, **bit_depth_vps_luma_minus8**, or **bit_depth_vps_chroma_minus8** may not be less than respectively, **chroma_format_idc**, **separate_colour_plane_flag**, **pic_width_in_luma_samples**, **pic_height_in_luma_samples**, **bit_depth_luma_minus8**, or **bit_depth_chroma_minus8** syntax elements in any SPS in the CVS with **nuh_layer_id** equal to 0.
- [0136] In some embodiments the term representation format may be used to include syntax elements **chroma_format_idc**, **separate_colour_plane_flag**,

pic_width_in_luma_samples, pic_height_in_luma_samples, bit_depth_luma_minus8, or bit_depth_chroma_minus8 in SPS and/ or syntax elements chroma_format_vps_idc, separate_colour_plane_vps_flag, pic_width_vps_in_luma_samples, pic_height_vps_in_luma_samples, bit_depth_vps_luma_minus8, or bit_depth_vps_chroma_minus8 in VPS. In some embodiments the representation format information may be called source format information.

[0137] In some embodiments it may be required that the first representation format information including bit-depth of luma samples, bit depth of chroma samples and color chromaticity information be sent corresponding to the format of layer with nuh_layer_id equal to 0. In some case the layer with nuh_layer_id equal to 0 may be a base layer. In some embodiments source format information sent in first representation format may provides a meaningful upper bound to corresponding source information SPS syntax elements values for active SPS with nuh_layer_id equal to 0.

[0138] vps_rep_format_idx[i] may specify the index, into the list of rep_format() syntax structures in the VPS, of the rep_format() syntax structure that applies to the layer with nuh_layer_id equal to layer_id_in_nuh[i]. When i is equal to 0 or vps_num_rep_formats_minus1 is equal to 0, the value of vps_rep_format_idx[i] is inferred to be equal to 0. The value of vps_rep_format_idx[i] may be in the range of 0 to vps_num_rep_formats - 1, inclusive.

[0139] A syntax structure for rep_format() is shown in Table 6.

[0140]	rep_format() {	Descriptor
	chroma_format_vps_idc	u(2)
	if(chroma_format_vps_idc == 3)	
	separate_colour_plane_vps_flag	u(1)
	pic_width_vps_in_luma_samples	u(16)
	pic_height_vps_in_luma_samples	u(16)
	bit_depth_vps_luma_minus8	u(3)
	bit_depth_vps_chroma_minus8	u(3)
	layer_color_gamut	u(1)
	}	

Table 6

[0141] chroma_format_vps_idc, separate_colour_plane_vps_flag, pic_width_vps_in_luma_samples, pic_height_vps_in_luma_samples, bit_depth_vps_luma_minus8, and bit_depth_vps_chroma_minus8 are used for inference of the values of the SPS syntax elements chroma_format_idc, separate_colour_plane_flag, pic_width_in_luma_samples, pic_height_in_luma_samples, bit_depth_luma_minus8, and bit_depth_chroma_minus8, respectively, for each SPS that refers to the VPS.

[0142] For each of these syntax elements, all constraints, if any, that apply to the value of

the corresponding SPS syntax element also apply.

[0143] layer_color_gamut equal to 1 may specify that the chromaticity coordinates of the source primaries are defined as per Rec. ITU-R BT.2020. layer_color_gamut[id] equal to 0 may specify that the chromaticity coordinates of the source primaries are defined as per Rec. ITU-R BT.709.

[0144] A syntax structure for a sequence parameter set (SPS) is shown in Table 7 below.

seq_parameter_set_rbsp() {
sps_video_parameter_set_id
if(nuh_layer_id == 0) {
sps_max_sub_layers_minus1
sps_temporal_id_nesting_flag
profile_tier_level(1, sps_max_sub_layers_minus1)
}
sps_seq_parameter_set_id
if(nuh_layer_id > 0)
update_rep_format_flag
if(update_rep_format_flag) {
chroma_format_idc
if(chroma_format_idc == 3)
separate_colour_plane_flag
pic_width_in_luma_samples
pic_height_in_luma_samples
}
...
if(update_rep_format_flag) {
bit_depth_luma_minus8
bit_depth_chroma_minus8
}
...
}

Table 7

[0145] sps_video_parameter_set_id may specify the value of the vps_video_parameter_set_id of the active VPS. vps_video_parameter_set_id identifies the VPS for reference by other syntax elements.

[0146] sps_max_sub_layers_minus1 plus 1 may specify the maximum number of temporal sub-layers that may be present in each CVS referring to the SPS. The value of sps_max_sub_layers_minus1 may be in the range of 0 to 6, inclusive. A temporal sub-layer is a temporal scalable layer of a temporal scalable bitstream, consisting of video coding layer network abstraction (VCL NAL) units with a particular value of temporal identifier and the associated non-VCL NAL units.

[0147] sps_temporal_id_nesting_flag, when sps_max_sub_layers_minus1 is greater than 0, may specify whether inter prediction is additionally restricted for CVSs referring to the SPS. When sps_max_sub_layers_minus1 is equal to 0, sps_temporal_id_nesting_flag may be equal to 1.

[0148] The syntax element sps_temporal_id_nesting_flag may be used to indicate that temporal up-switching, i.e. switching from decoding up to any TemporalId tIdN to

decoding up to any TemporalId tIdM that is greater than tIdN, is always possible in the CVS.

- [0149] `sps_seq_parameter_set_id` provides an identifier for the SPS for reference by other syntax elements. The value of `sps_seq_parameter_set_id` may be in the range of 0 to 15, inclusive.
- [0150] `chroma_format_idc` may specify the chroma sampling relative to the luma sampling. The value of `chroma_format_idc` may be in the range of 0 to 3, inclusive.
- [0151] `separate_colour_plane_flag` equal to 1 may specify that the three colour components of the 4:4:4 chroma format are coded separately. `separate_colour_plane_flag` equal to 0 may specify that the colour components are not coded separately. When `separate_colour_plane_flag` is not present, it may be inferred to be equal to 0. When `separate_colour_plane_flag` is equal to 1, the coded picture may consist of three separate components, each of which may consist of coded samples of one colour plane (Y, Cb, or Cr) and may use the monochrome coding syntax. In this case, each colour plane may be associated with a specific `colour_plane_id` value.
- [0152] Depending on the value of `separate_colour_plane_flag`, the value of the variable `ChromaArrayType` may be assigned as follows:
- [0153] If `separate_colour_plane_flag` is equal to 0, `ChromaArrayType` may be set equal to `chroma_format_idc`.
- [0154] Otherwise (`separate_colour_plane_flag` is equal to 1), `ChromaArrayType` may be set equal to 0.
- [0155] `pic_width_in_luma_samples` may specify the width of each decoded picture in units of luma samples. `pic_width_in_luma_samples` may not be equal to 0 and may be an integer multiple of a variable `MinCbSizeY`.
- [0156] `pic_height_in_luma_samples` may specify the height of each decoded picture in units of luma samples. `pic_height_in_luma_samples` may not be equal to 0 and may be an integer
- [0157] `bit_depth_luma_minus8` may specify the bit depth of the samples of the luma array `BitDepthY` and the value of the luma quantization parameter range offset `QpBdOffsetY` as follows:

$$\text{BitDepthY} = 8 + \text{bit_depth_luma_minus8}$$

$$\text{QpBdOffsetY} = 6 * \text{bit_depth_luma_minus8}$$
- [0158] `bit_depth_luma_minus8` may be in the range of 0 to 6, inclusive.
- [0159] `bit_depth_chroma_minus8` may specify the bit depth of the samples of the chroma arrays `BitDepthC` and the value of the chroma quantization parameter range offset `QpBdOffsetC` as follows:

$$\text{BitDepthC} = 8 + \text{bit_depth_chroma_minus8}$$

$$\text{QpBdOffsetC} = 6 * \text{bit_depth_chroma_minus8}$$

- [0160] `bit_depth_chroma_minus8` may be in the range of 0 to 6, inclusive.
- [0161] `update_rep_format_flag` equal to 1 may specify that the syntax elements `chroma_format_idc`, `separate_colour_plane_flag`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `bit_depth_luma_minus8`, and `bit_depth_chroma_minus8` are explicitly signalled in the SPS and all the layers with `nuh_layer_id` greater than zero that refer to this SPS use these values instead of those signalled in the VPS when the `nuh_layer_id` of the SPS is greater than 0. `update_rep_format_flag` equal to 0 may specify that the syntax elements `chroma_format_idc`, `separate_colour_plane_flag`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `bit_depth_luma_minus8`, and `bit_depth_chroma_minus8` are not signalled in the SPS and all the layers that refer to this SPS use the values signaled in the VPS. When not present, the value of `update_rep_format_flag` is inferred to be equal to 1. Thus for a base layer or a layer with `nuh_layer_id` equal to 0 `update_rep_format_flag` may be inferred to be equal to 1 and the syntax elements `chroma_format_idc`, `separate_colour_plane_flag`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `bit_depth_luma_minus8`, and `bit_depth_chroma_minus8` are explicitly signalled in this SPS with `nuh_layer_id` equal to 0.
- [0162] When a current picture with `nuh_layer_id` `layerIdCurr` greater than 0 refers to an SPS, the values of `chroma_format_idc`, `separate_colour_plane_flag`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `bit_depth_luma_minus8`, and `bit_depth_chroma_minus8` may be inferred or constrained as follows:

If the `nuh_layer_id` of the active layer SPS is equal to 0,

- if `vps_rep_format_idx[j]` in the active VPS where `j` is equal to `LayerIdxInVps[layerIdCurr]` is equal to 0 the values of `chroma_format_idc`, `separate_colour_plane_flag`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `bit_depth_luma_minus8`, and `bit_depth_chroma_minus8` may be inferred to be equal to the respective values from the active layer SPS.
- otherwise (`vps_rep_format_idx[j]` in the active VPS where `j` is equal to `LayerIdxInVps[layerIdCurr]` is not equal to 0) the values of `chroma_format_idc`, `separate_colour_plane_flag`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `bit_depth_luma_minus8`, and `bit_depth_chroma_minus8` may be inferred to be equal to `chroma_format_vps_idc`, `separate_colour_plane_vps_flag`, `pic_width_vps_in_luma_samples`, `pic_height_vps_in_luma_samples`, `bit_depth_vps_luma_minus8`, and `bit_depth_vps_chroma_minus8`, respectively, of the `vps_rep_format_idx[j]`-th `rep_format()` syntax structure in the active VPS where `j` is equal to `LayerIdxInVps[layerIdCurr]` and the values of `chroma_format_idc`, `separate_colour_plane_flag`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `bit_depth_luma_minus8`, and `bit_depth_chroma_minus8` of the active layer SPS may be ignored.

Otherwise when the `nuh_layer_id` of the active layer SPS is greater than zero, the following may be used for inference of various values:

- if `update_rep_format_flag` is equal to 0, the values of `chroma_format_idc`, `separate_colour_plane_flag`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `bit_depth_luma_minus8`, and `bit_depth_chroma_minus8` may be inferred to be equal to `chroma_format_vps_idc`, `separate_colour_plane_vps_flag`, `pic_width_vps_in_luma_samples`, `pic_height_vps_in_luma_samples`, `bit_depth_vps_luma_minus8`, and `bit_depth_vps_chroma_minus8`, respectively, of the `vps_rep_format_idx[j]`-th `rep_format()` syntax structure in the active VPS, where `j` is equal to `LayerIdxInVps[layerIdCurr]`.
- Otherwise (`update_rep_format_flag` is not equal to 0), it may be a requirement of bitstream conformance that the value of `chroma_format_idc`, `separate_colour_plane_flag`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `bit_depth_luma_minus8`, or `bit_depth_chroma_minus8` may be less than or equal to `chroma_format_vps_idc`, `separate_colour_plane_vps_flag`, `pic_width_vps_in_luma_samples`, `pic_height_vps_in_luma_samples`, `bit_depth_vps_luma_minus8`, or `bit_depth_vps_chroma_minus8`, respectively, of the `vps_rep_format_idx[j]`-th `rep_format()` syntax structure in the active VPS, where `j` is equal to `LayerIdxInVps[layerIdCurr]`.

[0163] In some embodiments parameter update inference when SPS carries representation format information may allow updating the representation format information for higher layers from a SPS of base layer. In some embodiment higher layers may be layers with `nuh_layer_id` greater than zero. In some embodiments base layer may be layer with `nuh_layer_id` equal to 0. Thus in some embodiments parameter update inference when SPS carries representation format information may allow updating the representation format information for layers with `nuh_layer_id` greater than zero from

a SPS of layer with nuh_layer_id equal to 0. The SPS with nuh_layer_id equal to zero may be the active SPS for the layers with nuh_layer_id greater than or equal to 0.

[0164] A few more embodiments for signaling rep_format() and related syntax elements are described next. These are further variants of the Table 5.

[0165] A modified syntax for video parameter set extension (vps_extension()) which is signaled inside a video parameter set VPS is shown in Table 8. This is a variant of the syntax in Table 5.

vps_extension() {	Descriptor
...	
rep_format_idx_present_flag	u(1)
if(rep_format_idx_present_flag)	
vps_num_rep_formats_minus1	u(4)
for(i = 0; i <= vps_num_rep_formats_minus1; i++)	
rep_format()	
if(rep_format_idx_present_flag) {	
for(i = 1; i <= vps_max_layers_minus1; i++)	
if(vps_num_rep_formats_minus1 > 0)	
vps_rep_format_idx[i]	u(4)
}	
max_one_active_ref_layer_flag	u(1)
...	
}	

Table 8

[0166] rep_format_idx_present_flag equal to 1 indicates that the syntax elements vps_num_rep_formats_minus1 and vps_rep_format_idx[i] may be present. rep_format_idx_present_flag equal to 0 indicates that the syntax elements vps_num_rep_formats_minus1 and vps_rep_format_idx[i] are not present. In this case vps_num_rep_formats_minus1 is inferred to be equal to vps_max_layers_minus1 and vps_rep_format_idx[i] is inferred to be equal to i for each of i=0,...,vps_max_layers_minus1.

[0167] Other syntax elements may have the same semantics meanings as in Table 5.

[0168] Two modified syntax for video parameter set extension (vps_extension()) which is signaled inside a video parameter set VPS are shown in Table 9A and Table 9B. These is a variant of the syntax in Table 5.

vps_extension() {	Descriptor
...	
vps_num_rep_formats	u(4)
for(i = 0; i < vps_num_rep_formats; i++)	
rep_format()	
for(i = 1; i <= vps_max_layers_minus1; i++)	
if(vps_num_rep_formats > 1)	
vps_rep_format_idx[i]	u(4)
max_one_active_ref_layer_flag	u(1)
...	
}	

Table 9 A

vps_extension() {	Descriptor
...	
rep_format_idx_present_flag	u(1)
if(rep_format_idx_present_flag)	
vps_num_rep_formats	u(4)
for(i = 0; i < vps_num_rep_formats; i++)	
rep_format()	
if(rep_format_idx_present_flag) {	
for(i = 1; i <= vps_max_layers_minus1; i++)	
if(vps_num_rep_formats > 1)	
vps_rep_format_idx[i]	u(4)
}	
max_one_active_ref_layer_flag	u(1)
...	
}	

Table 9 B

- [0169] vps_num_rep_formats specifies the number of the following rep_format() syntax structures in the VPS. If vps_num_rep_formats is equal to 0 then no representation format information is signaled in VPS. In this case rep_format_idx_present_flag may be set to 1.
- [0170] Other syntax elements may have the same semantics meanings as in Table 5.
- [0171] A modified syntax for video parameter set extension (vps_extension()) which is signaled inside a video parameter set VPS is shown in Table 10. This is a variant of the syntax in Table 5.

vps_extension() {	Descriptor
...	
vps_rep_format_information_present_flag	u(1)
If(vps_rep_format_information_present_flag) {	
vps_num_rep_formats_minus1	u(4)
for(i = 0; i <= vps_num_rep_formats_minus1; i++)	
rep_format()	
for(i = 1; i <= vps_max_layers_minus1; i++)	
if(vps_num_rep_formats_minus1 > 0)	
vps_rep_format_idx[i]	u(4)
}	
max_one_active_ref_layer_flag	u(1)
...	
}	

Table 10

- [0172] vps_rep_format_information_present_flag equal to 1 indicates that the syntax elements vps_num_rep_formats_minus1 and vps_rep_format_idx[i] may be present. vps_rep_format_information_present_flag equal to 0 indicates that the syntax elements vps_num_rep_formats_minus1 and vps_rep_format_idx[i] are not present.
- [0173] Other syntax elements may have the same semantics meanings as in Table 5.
- [0174] The system and apparatus described above may use dedicated processor systems, micro controllers, programmable logic devices, microprocessors, or any combination thereof, to perform some or all of the operations described herein. Some of the operations described above may be implemented in software and other operations may be implemented in hardware. Any of the operations, processes, and/or methods described herein may be performed by an apparatus, a device, and/or a system substantially similar to those as described herein and with reference to the illustrated figures.
- [0175] The processing device may execute instructions or "code" stored in memory. The memory may store data as well. The processing device may include, but may not be limited to, an analog processor, a digital processor, a microprocessor, a multi-core processor, a processor array, a network processor, or the like. The processing device may be part of an integrated control system or system manager, or may be provided as a portable electronic device configured to interface with a networked system either locally or remotely via wireless transmission.
- [0176] The processor memory may be integrated together with the processing device, for example RAM or FLASH memory disposed within an integrated circuit micro-processor or the like. In other examples, the memory may comprise an independent device, such as an external disk drive, a storage array, a portable FLASH key fob, or the like. The memory and processing device may be operatively coupled together, or in communication with each other, for example by an I/O port, a network connection, or

the like, and the processing device may read a file stored on the memory. Associated memory may be "read only" by design (ROM) by virtue of permission settings, or not. Other examples of memory may include, but may not be limited to, WORM, EPROM, EEPROM, FLASH, or the like, which may be implemented in solid state semiconductor devices. Other memories may comprise moving parts, such as a known rotating disk drive. All such memories may be "machine-readable" and may be readable by a processing device.

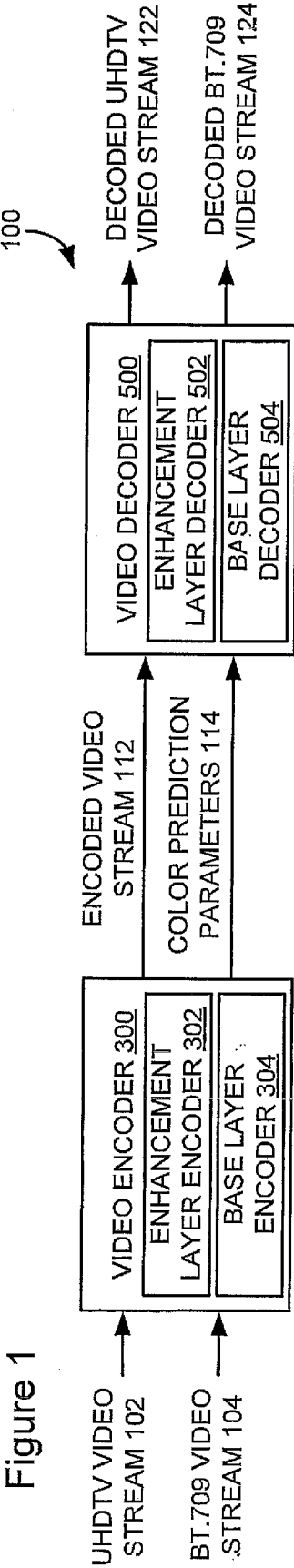
- [0177] Operating instructions or commands may be implemented or embodied in tangible forms of stored computer software (also known as "computer program" or "code"). Programs, or code, may be stored in a digital memory and may be read by the processing device. "Computer-readable storage medium" (or alternatively, "machine-readable storage medium") may include all of the foregoing types of memory, as well as new technologies of the future, as long as the memory may be capable of storing digital information in the nature of a computer program or other data, at least temporarily, and as long as the stored information may be "read" by an appropriate processing device. The term "computer-readable" may not be limited to the historical usage of "computer" to imply a complete mainframe, mini-computer, desktop or even laptop computer. Rather, "computer-readable" may comprise storage medium that may be readable by a processor, a processing device, or any computing system. Such media may be any available media that may be locally and/or remotely accessible by a computer or a processor, and may include volatile and non-volatile media, and removable and non-removable media, or any combination thereof.
- [0178] A program stored in a computer-readable storage medium may comprise a computer program product. For example, a storage medium may be used as a convenient means to store or transport a computer program. For the sake of convenience, the operations may be described as various interconnected or coupled functional blocks or diagrams. However, there may be cases where these functional blocks or diagrams may be equivalently aggregated into a single logic device, program or operation with unclear boundaries.
- [0179] One of skill in the art will recognize that the concepts taught herein can be tailored to a particular application in many other ways. In particular, those skilled in the art will recognize that the illustrated examples are but one of many alternative implementations that will become apparent upon reading this disclosure.
- [0180] Although the specification may refer to "an", "one", "another", or "some" example(s) in several locations, this does not necessarily mean that each such reference is to the same example(s), or that the feature only applies to a single example.
- [0181] It is to be understood that the claims are not limited to the precise configuration and components illustrated above. Various modifications, changes and variations may be

made in the arrangement, operation and details of the systems, methods, and apparatus described herein without departing from the scope of the claims.

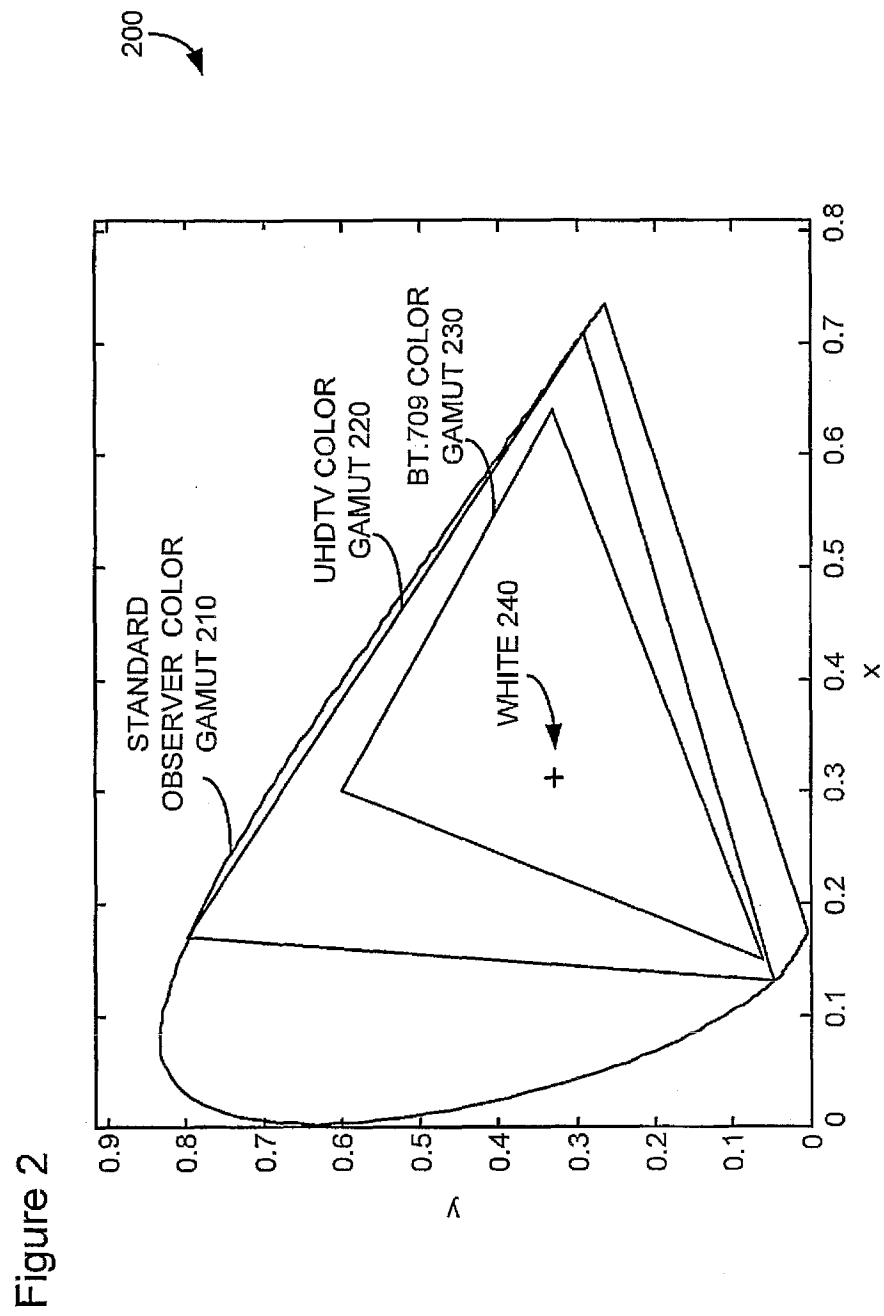
Claims

- [Claim 1] A method for decoding a video sequence comprising:
- (a) decoding a plurality of frames of said video sequence where at least two of said frames belonging to a different one of a plurality of layers;
 - (b) said video sequence includes a video parameter set syntax used for said decoding of regions of said encoded video;
 - (c) said video parameter set syntax includes a video parameter set extension set syntax including attributes regarding representation format characteristics of an encoded video format used for said decoding of said regions of said video sequence;
 - (d) said video parameter set syntax includes said video parameter set extension set syntax including the number of representation format characteristic syntax structures and an index referencing a representation format structure for each layer in said video parameter set extension set syntax;
 - (e) each of said number of said representation format characteristic syntax structures in said video parameter set extension set syntax indicates a different set of said attributes regarding said representation format characteristics of said encoded video format used for decoding said regions of said video sequence.
- [Claim 2] The method of claim 1 wherein said index referencing a representation format structure for each layer is only signaled in said video sequence if a representation format index present flag is equal to 1.
- [Claim 3] The method of claim 1 wherein said index referencing a representation format structure for each layer is not signaled in said video sequence if a representative format index present flag is equal to 0.
- [Claim 4] The method of claim 3 wherein said representative format index present flag is inferred to be equal to 0 when said representative format index present flag is not present.
- [Claim 5] The method of claim 1 wherein a corresponding index *i* value for said index referring to said representation format structure for each said layer is inferred to be *i* for each of a corresponding set of said layers when the index is not signaled.

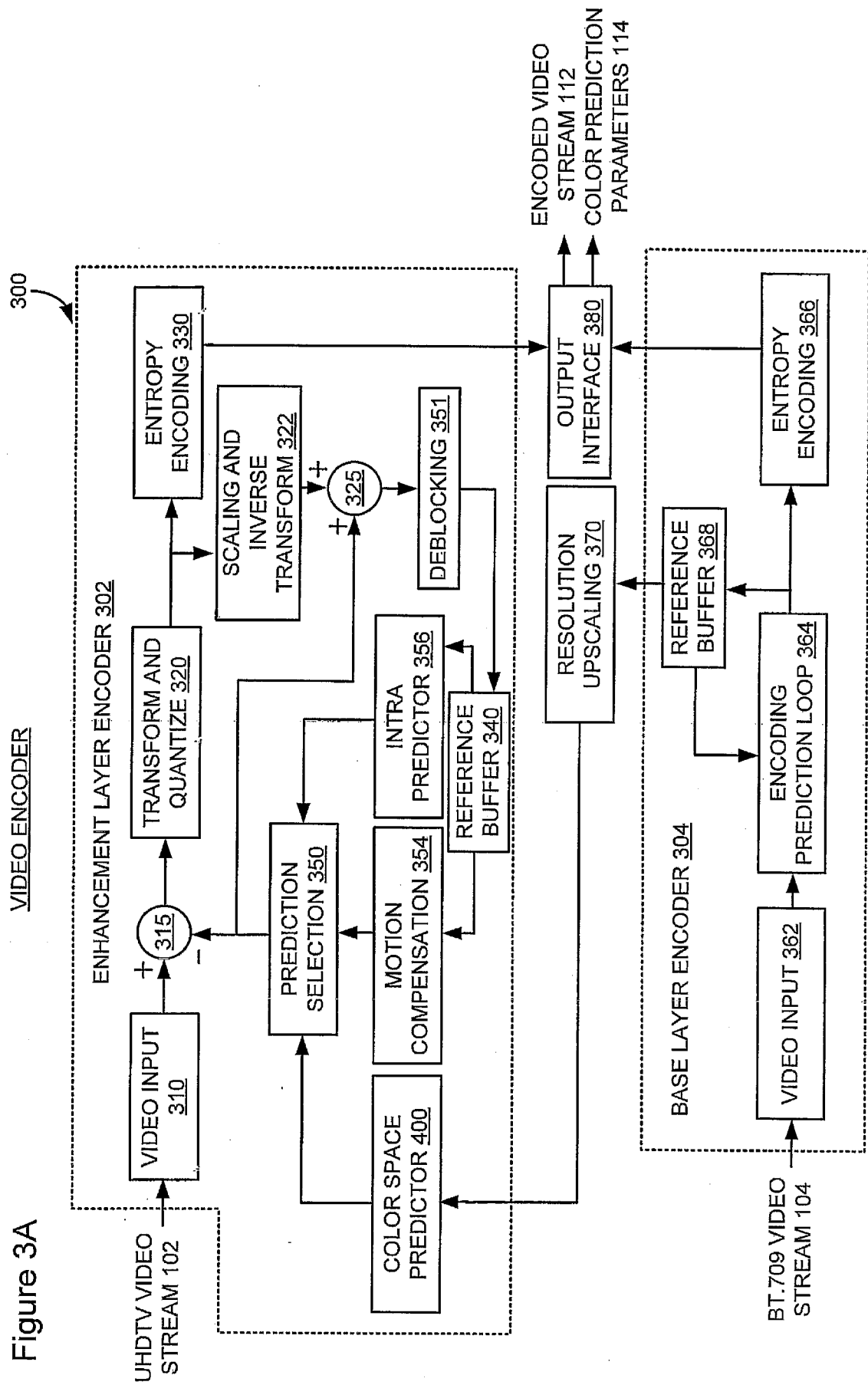
[Fig. 1]



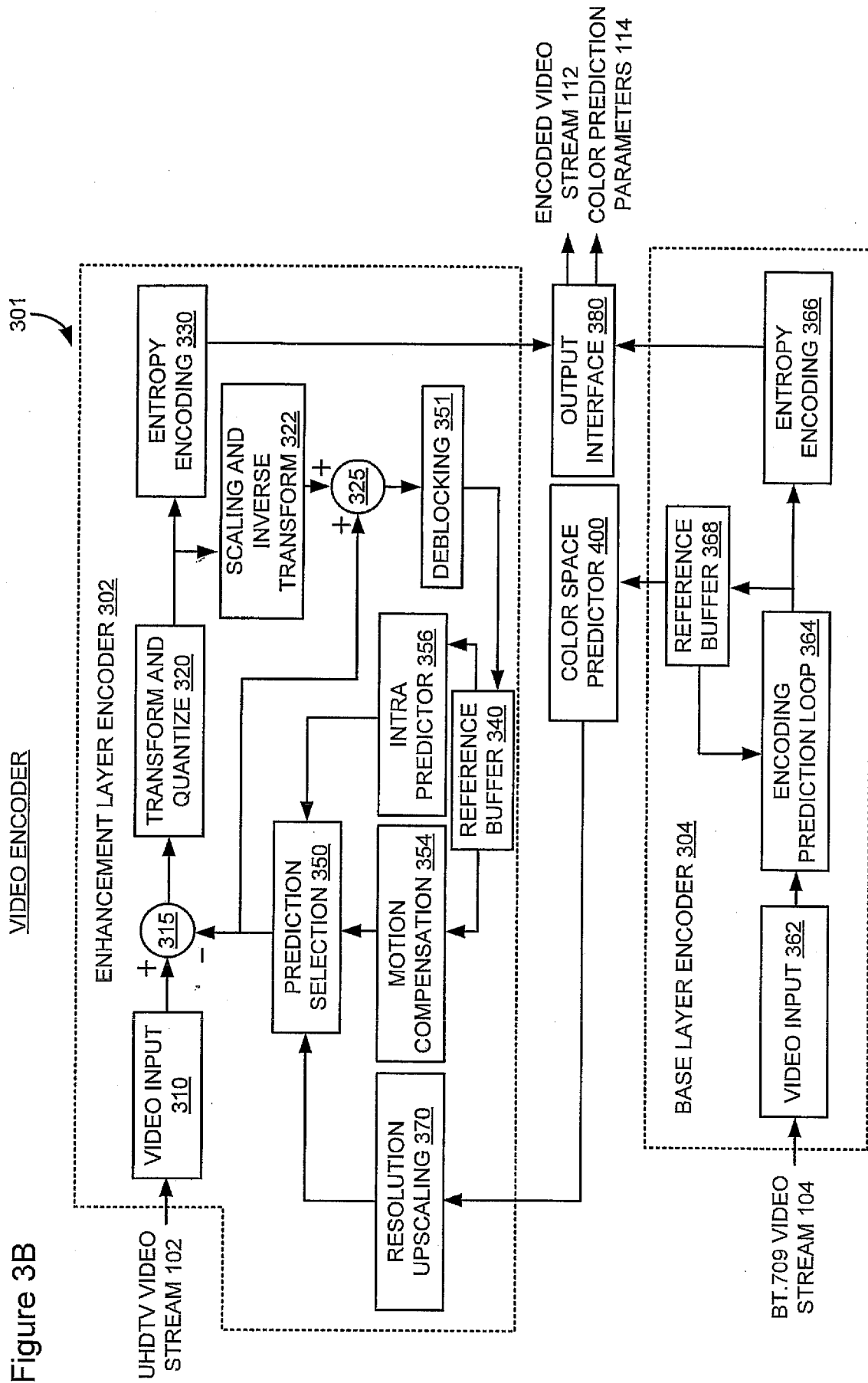
[Fig. 2]



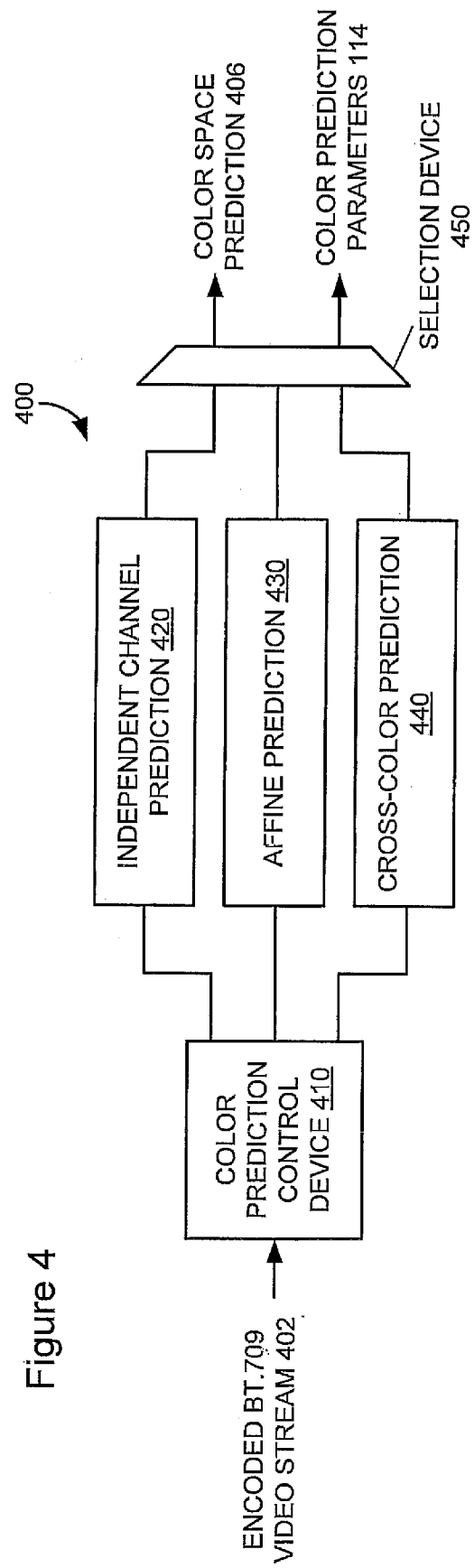
[Fig. 3A]



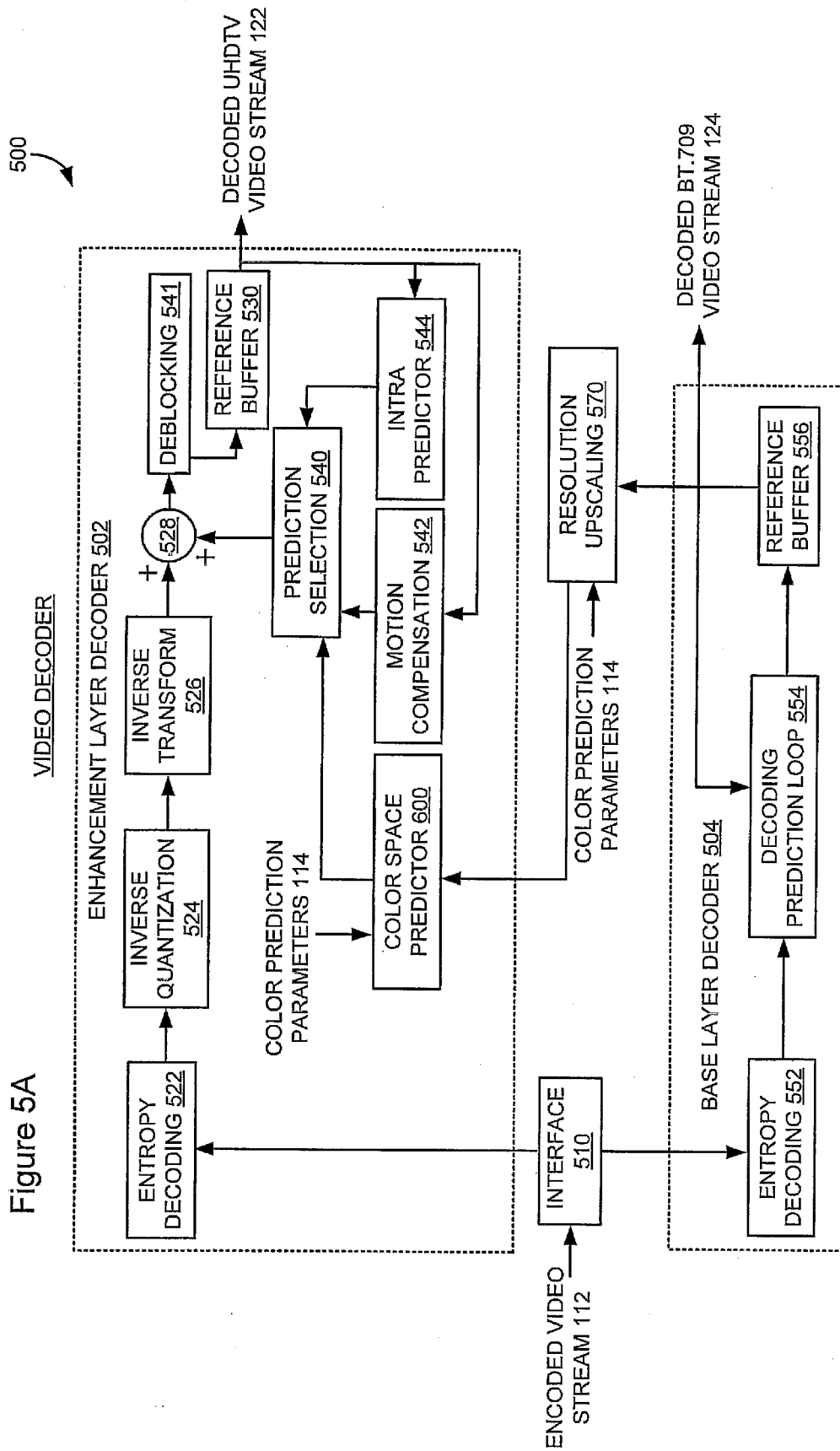
[Fig. 3B]



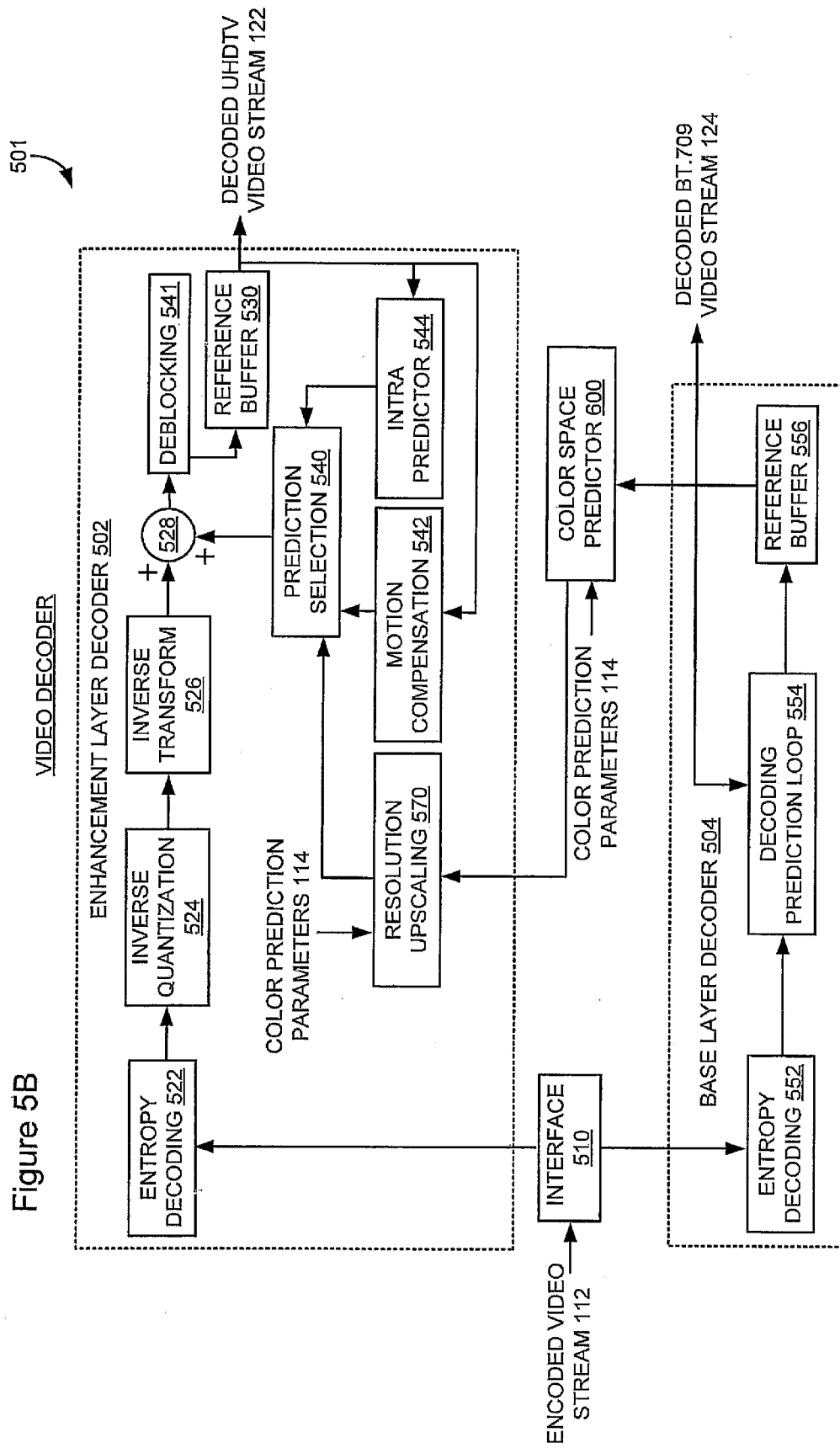
[Fig. 4]



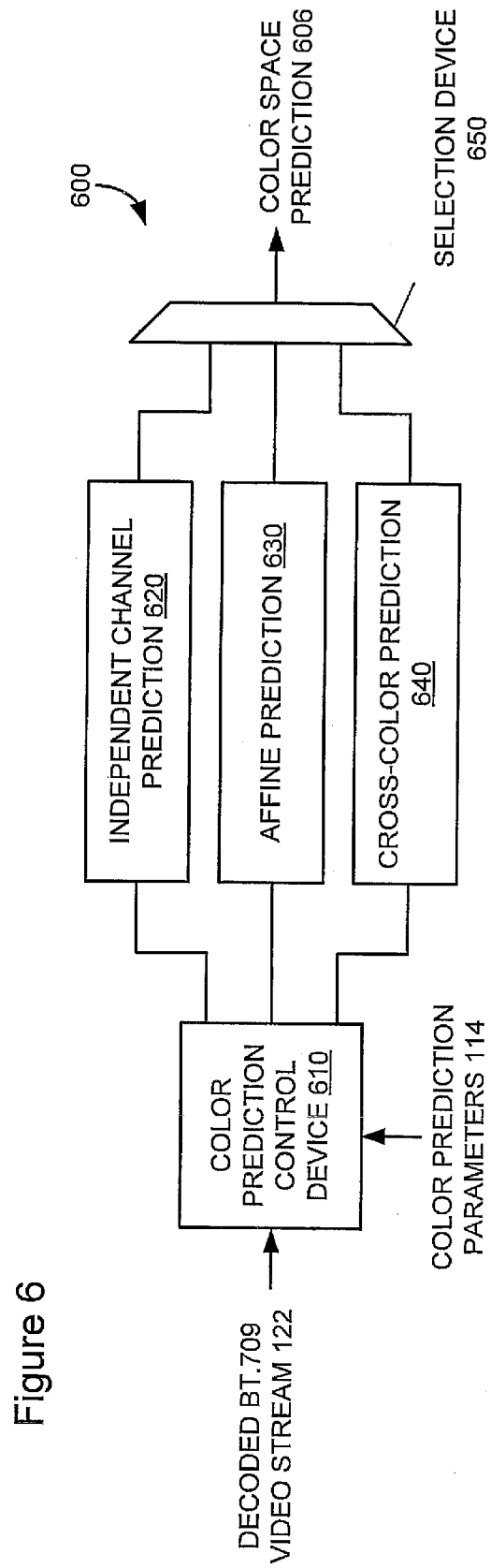
[Fig. 5A]



[Fig. 5B]

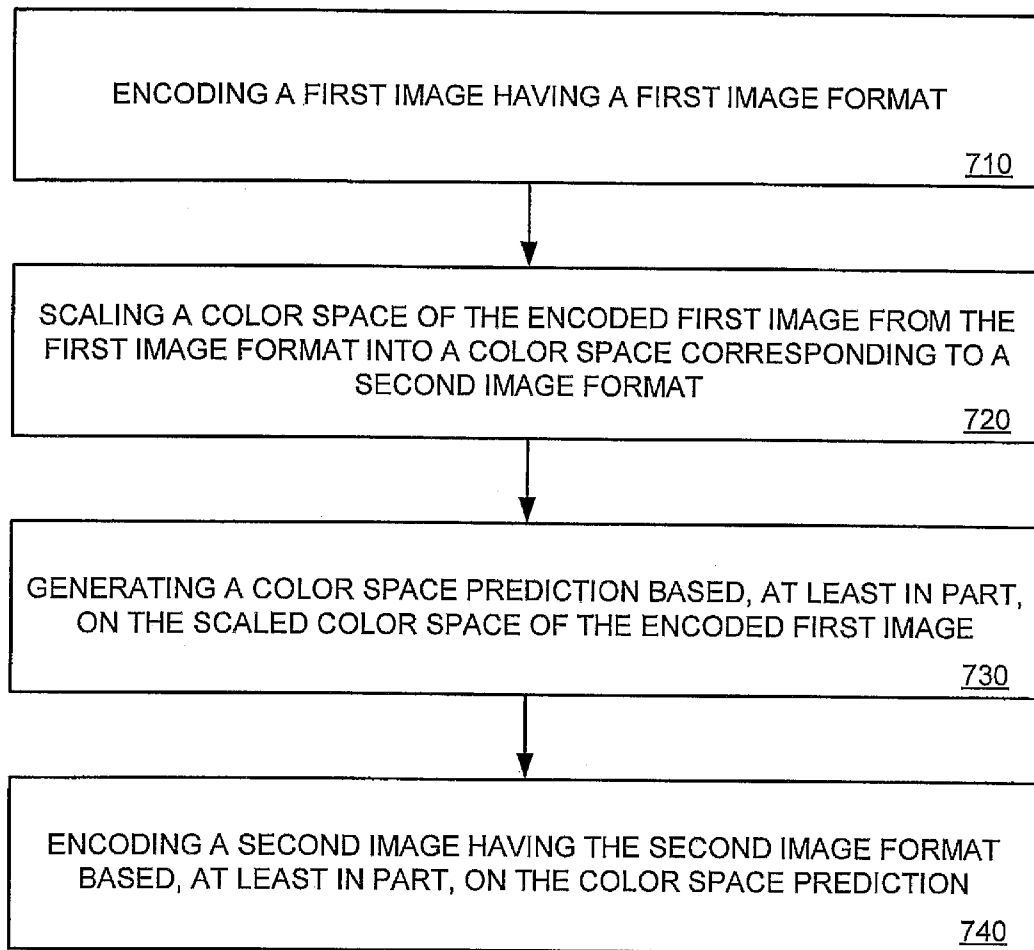


[Fig. 6]



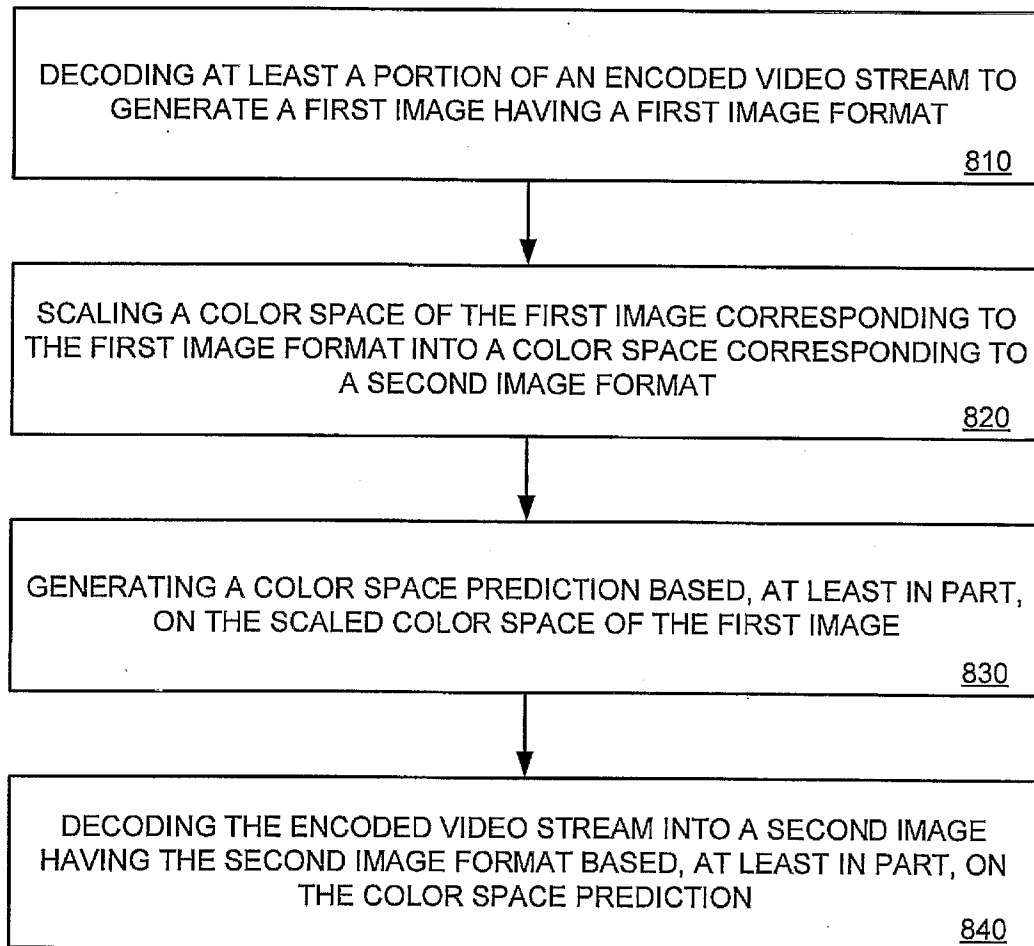
[Fig. 7]

Figure 7



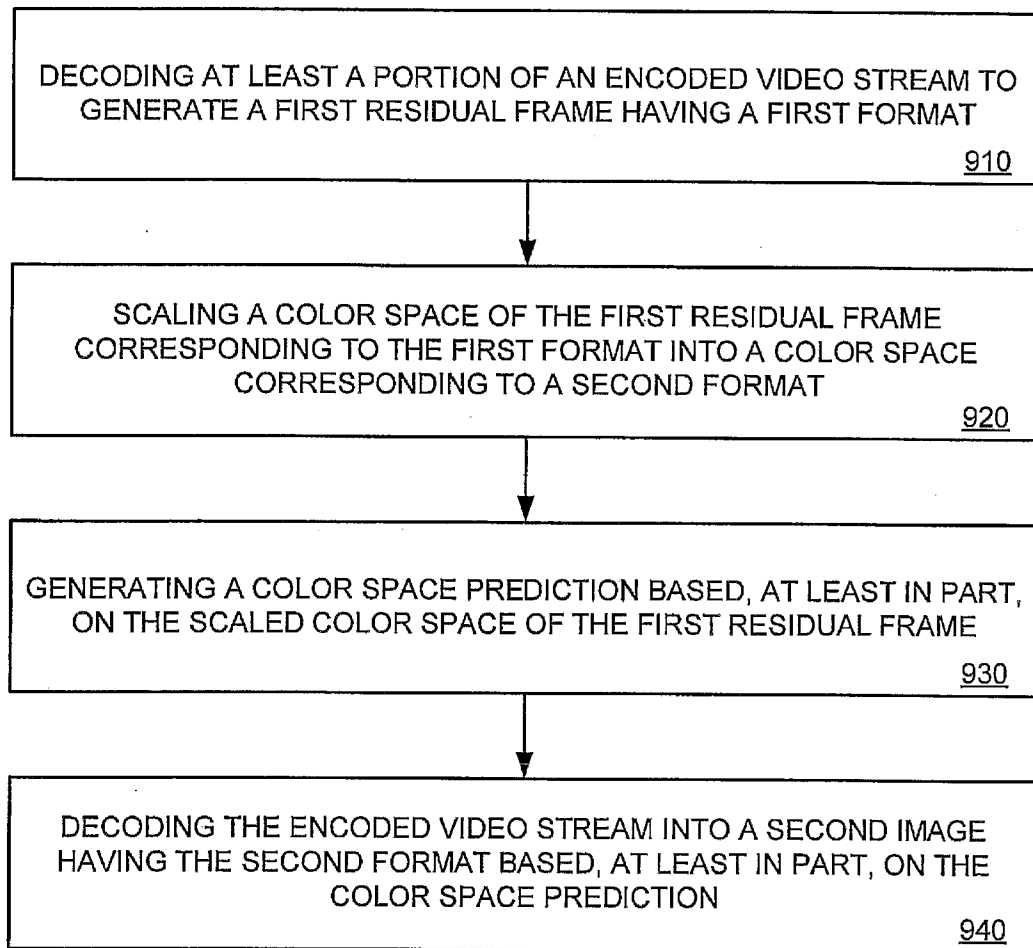
[Fig. 8]

Figure 8

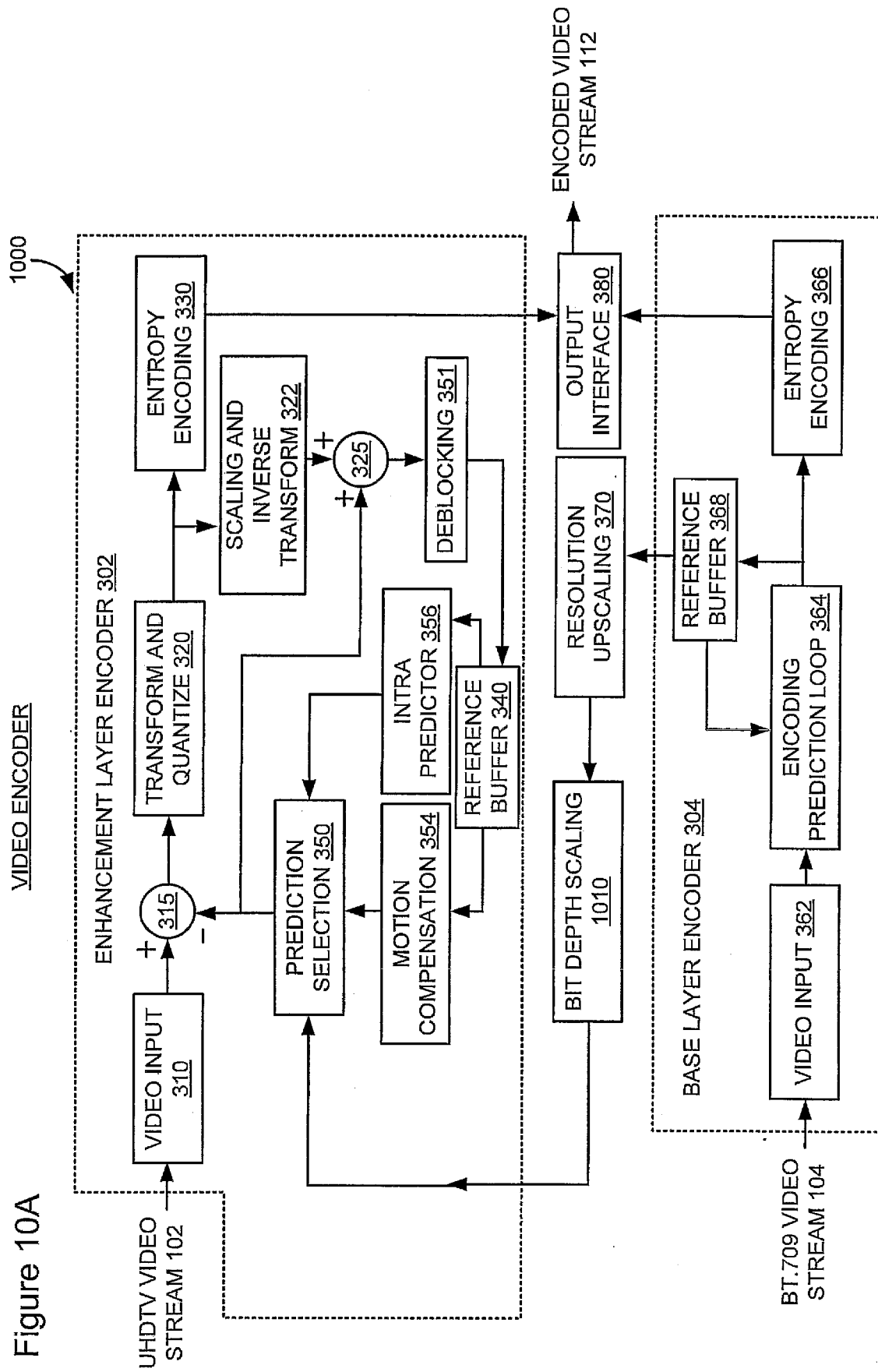


[Fig. 9]

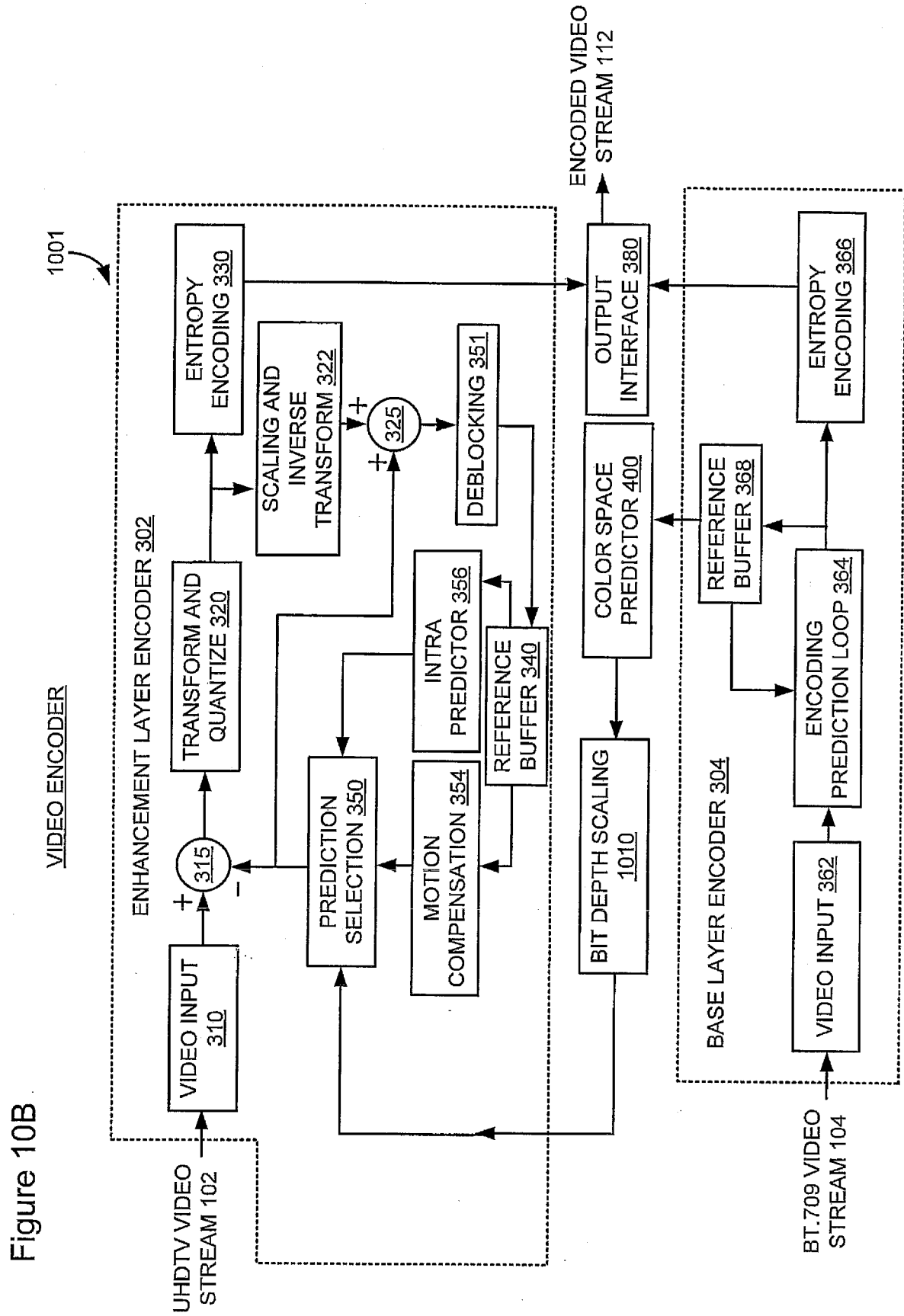
Figure 9



[Fig. 10A]

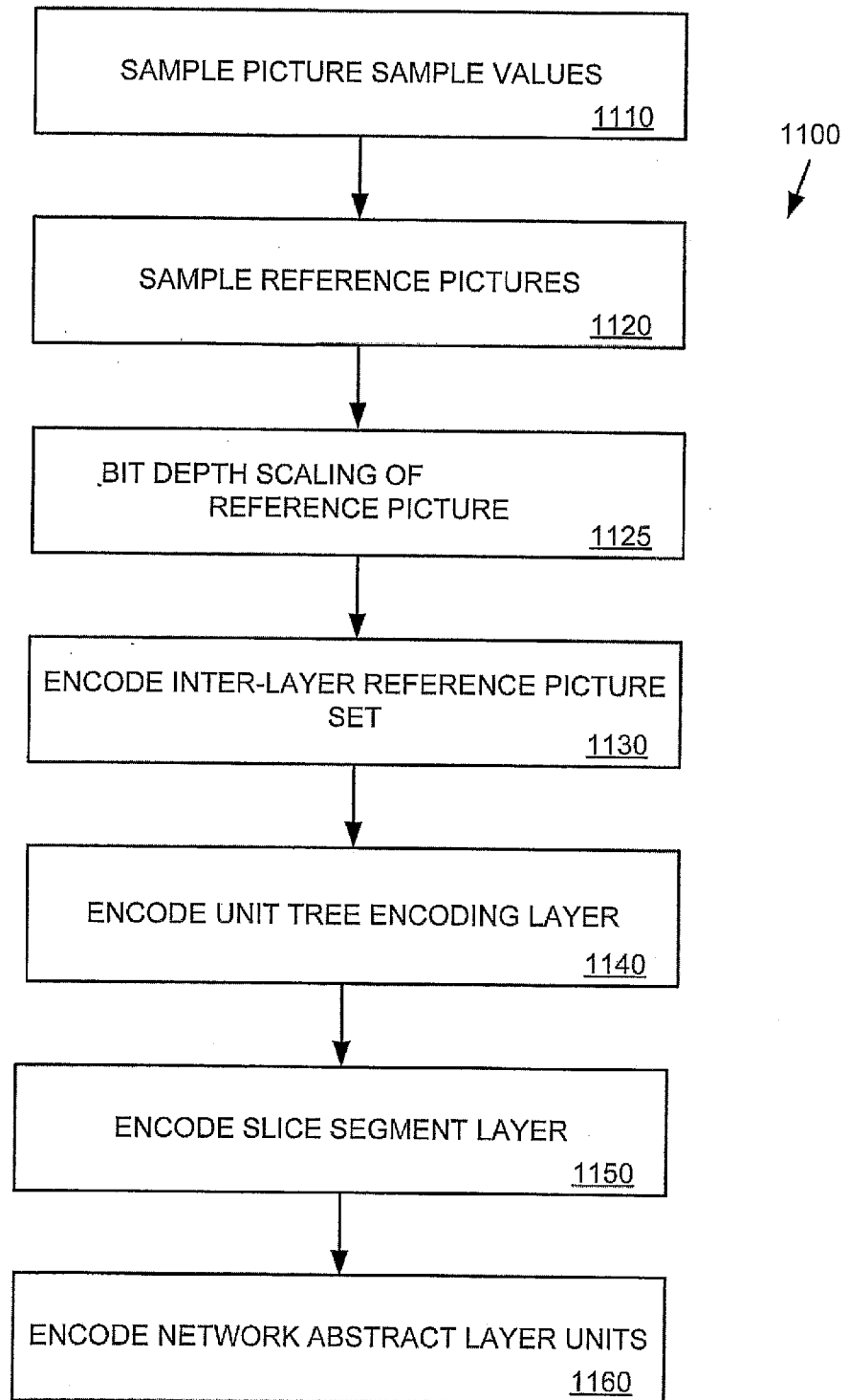


[Fig. 10B]



[Fig. 11]

Figure 11



[Fig. 12A]

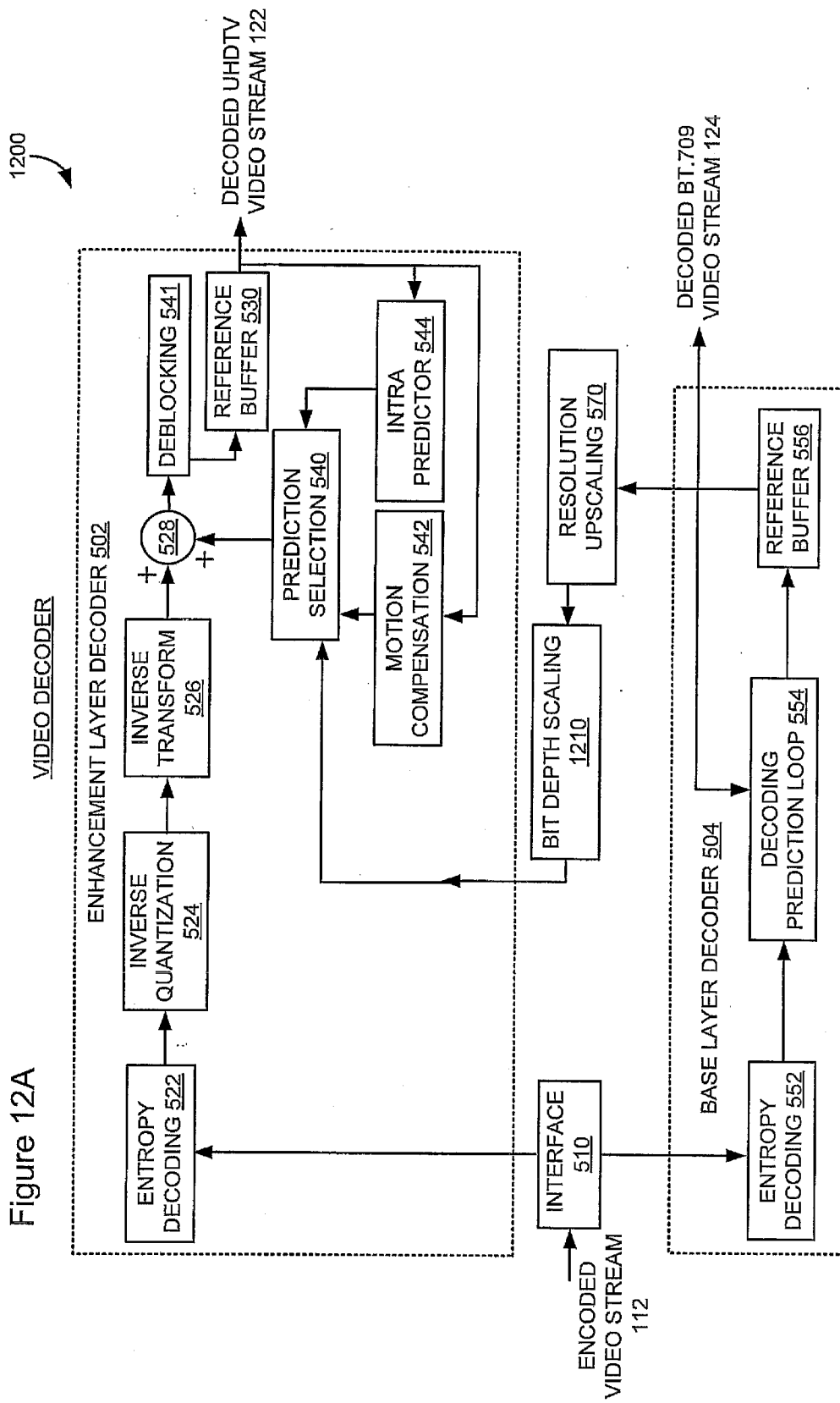
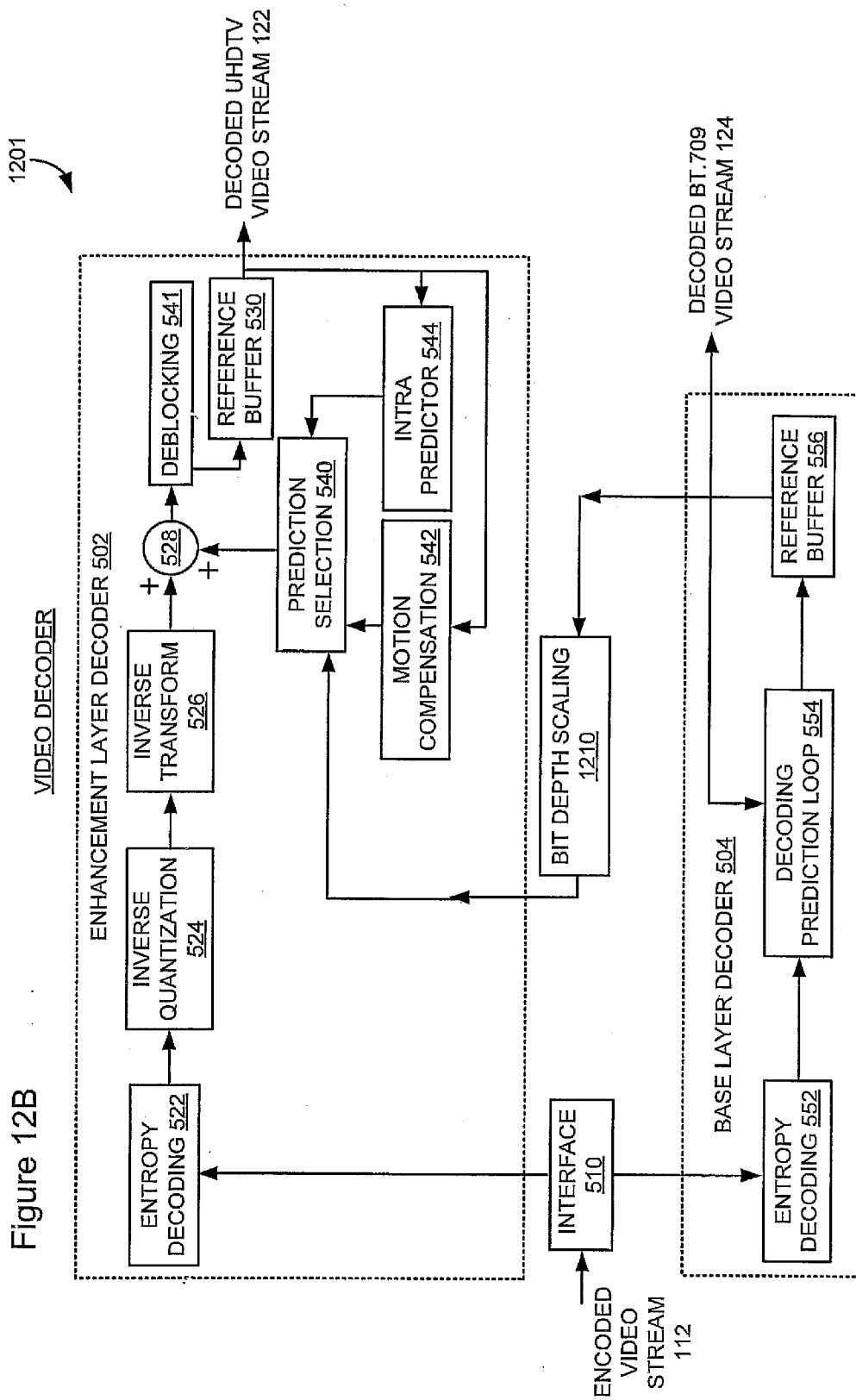
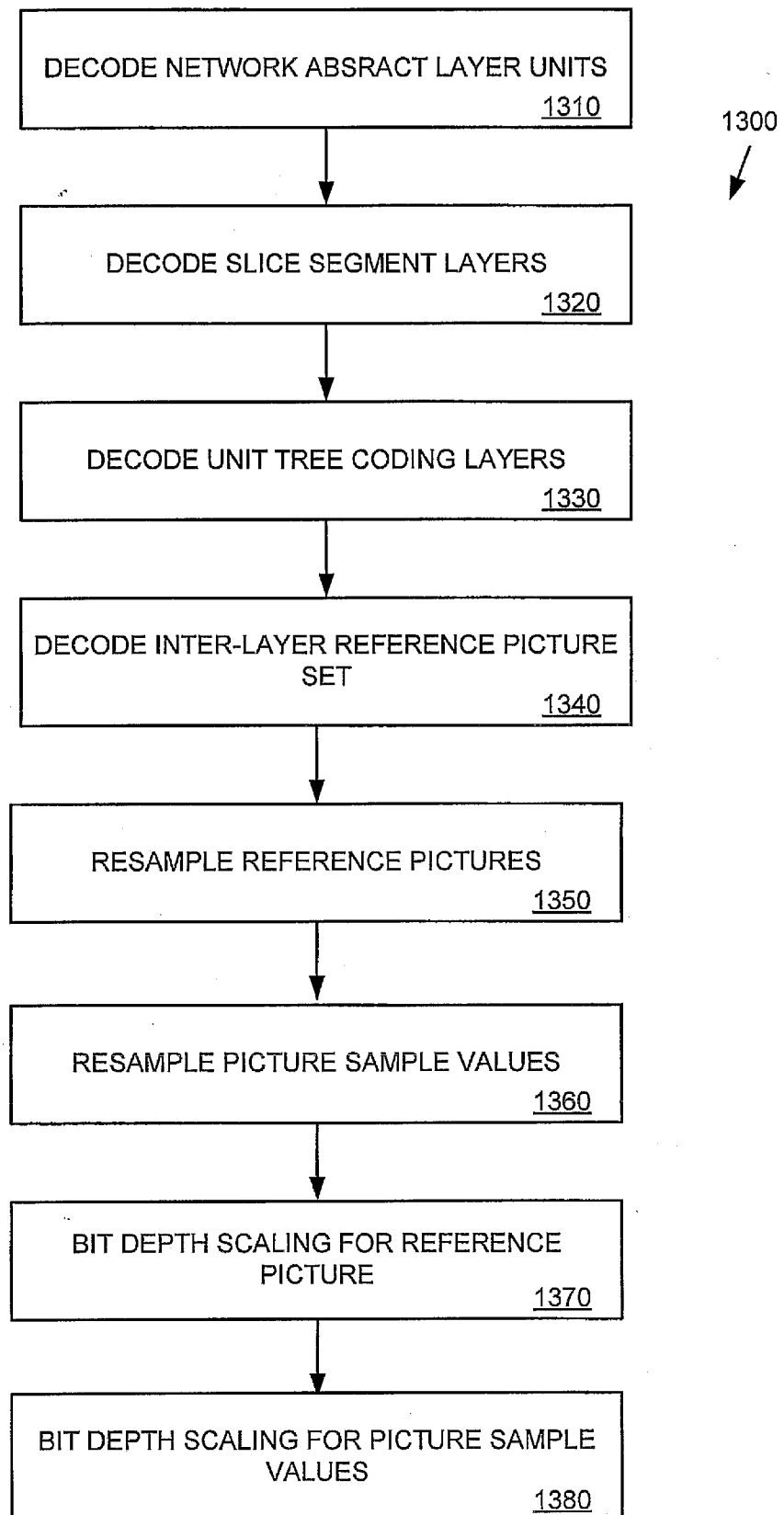


Figure 12B



[Fig. 13]

Figure 13



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2014/003676

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl. H04N19/85(2014.01)i, H04N19/30(2014.01)i, H04N19/70(2014.01)i

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)

Int.Cl. H04N19/00-19/98

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

Published examined utility model applications of Japan 1922-1996
 Published unexamined utility model applications of Japan 1971-2014
 Registered utility model specifications of Japan 1996-2014
 Published registered utility model applications of Japan 1994-2014

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Ying Chen and Ye-Kui Wang, AHG10: Video parameter set for HEVC extensions, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTVC-J0124r1, 10th Meeting: Stockholm, SE, 2012.07, pp.1-15	1-5
A	Sachin Deshpande et al., On Color Gamut Scalable Video Coding, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JCTVC-M0214-CG_HLS_V2, 13th Meeting: Incheon, KR, 2013.04, pp.1-6	1-5



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

12.09.2014

Date of mailing of the international search report

30.09.2014

Name and mailing address of the ISA/JP

Japan Patent Office

3-4-3, Kasumigaseki, Chiyoda-ku, Tokyo 100-8915, Japan

Authorized officer

Kenji IWAI

Telephone No. +81-3-3581-1101 Ext. 3541

5C

9465

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2014/003676

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Byeongdoo Choi et al., Video parameter set design, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JCTVC-L0132, 12th Meeting: Geneva, CH, 2013.01, pp.1-5	1-5
P,X	WO 2014/008286 A1 (QUALCOMM INCORPORATED) 2014.01.09, [0134], [0169] - [0179], FIG.5 & US 2014/0003491 A1 & US 2014/0003492 A1 & US 2014/0003493 A1 & WO 2014/008287 A & WO 2014/008290 A	1-5
P,X	Adarsh K. Ramasubramonian et al., MV-HEVC/SHVC HLS: Representation format information in VPS, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JCTVC-N0092r1, JCT3V-E0060r1, 14th Meeting: Vienna, AT, 2013.07, pp.1-5	1-5
P,X	Sachin Deshpande, On Source Representation Information Signaling in VPS, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JCTVC-N0238, 14th Meeting: Vienna, 2013.07, pp.1-3	1-5