



(51) International Patent Classification:

H04N 19/70 (2014.01) H04N 19/96 (2014.01)

(21) International Application Number:

PCT/US2017/049937

(22) International Filing Date:

01 September 2017 (01.09.2017)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

62/383,369 02 September 2016 (02.09.2016) US

(71) Applicant: VID SCALE, INC. [US/US]; 200 Bellevue Parkway, Suite 300, Wilmington, Delaware 19809 (US).

(72) Inventors: XIU, Xiaoyu; 9115 Judicial Drive, Suite #4109, San Diego, California 92122 (US). HE, Yuwen; 13542 Silver Vine Path, San Diego, California 92130 (US).

(74) Agent: STECK, Jeffrey Alan; Invention Mine LLC, 216 South Jefferson Street, Suite 102, Chicago, Illinois 60661 (US).

(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, DJ, DK, DM, DO,

DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

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

Published:

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

(54) Title: METHODS AND APPARATUS FOR CODED BLOCK FLAG CODING IN QUAD-TREE PLUS BINARY-TREE BLOCK PARTITIONING

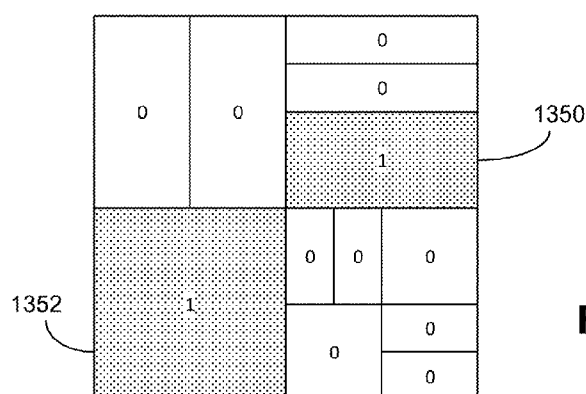
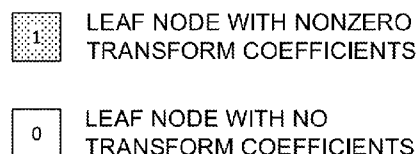
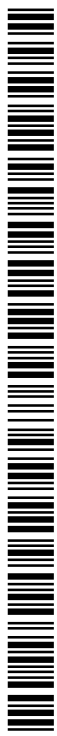


FIG. 13

(57) Abstract: Systems and methods are proposed herein for coded block flag (CBF) signaling. In some embodiments, a hierarchical signaling method is used to signal the CBFs of chroma components for the quad-tree plus binary tree (QTBT) structure. A CBF flag may be signaled at each QTBT node level for each chroma component, indicating whether any descendent QTBT leaf node under the current level is associated with a non-zero coefficient. In some embodiments, for inter-coded pictures, a flag at the QTBT root node may indicate whether there are non-zero transform coefficients in the descendent leaf nodes that originate from the current root node. When the flag is equal to 1, the coefficients of the descendent leaf nodes under the current node may be signaled; otherwise, no further residual information is transmitted and all the transform coefficients are inferred to be 0.



METHODS AND APPARATUS FOR CODED BLOCK FLAG CODING IN QUAD-TREE PLUS BINARY-TREE BLOCK PARTITIONING

CROSS-REFERENCE TO RELATED APPLICATION

[0001] The present application is a non-provisional filing of, and claims benefit under 35 U.S.C. §119(e) from, U.S. Provisional Patent Application Serial No. 62/383,369 entitled "METHODS AND APPARATUS FOR CODED BLOCK FLAG CODING IN QUAD-TREE PLUS BINARY-TREE BLOCK PARTITIONING," filed September 2, 2016, which is hereby incorporated by reference in its entirety.

BACKGROUND

[0002] Video coding systems are widely used to compress digital video signals to reduce the storage need and/or transmission bandwidth of such signals. Among the various types of video coding systems, such as block-based, wavelet-based, and object-based systems, nowadays block-based hybrid video coding systems are the most widely used and deployed. Examples of block-based video coding systems include international video coding standards such as the MPEG1/2/4 part 2, H.264/MPEG-4 part 10 AVC, VC-1, and the latest video coding standard called High Efficiency Video Coding (HEVC), which was developed by JCT-VC (Joint Collaborative Team on Video Coding) of ITU-T/SG16/Q.6/VCEG and ISO/IEC/MPEG.

[0003] The first version of the HEVC standard was finalized in Oct. 2013. HEVC offers approximately 50% bit-rate saving or equivalent perceptual quality compared to the prior generation video coding standard H.264/MPEG AVC. Although the HEVC standard provides significant coding improvements than its predecessor, there is evidence that superior coding efficiency can be achieved with additional coding tools over HEVC. Both VCEG and MPEG have started the exploration work of new coding technologies for future video coding standardization. In Oct. 2015, ITU-T VCEG and ISO/IEC MPEG formed the Joint Video Exploration Team (JVET) to begin significant study of advanced technologies that could enable substantial enhancement of coding efficiency over HEVC. In the same month, one software codebase, called Joint Exploration Model (JEM) was established for future video coding exploration work. The JEM reference software was based on HEVC Test Model (HM) that was developed by JCT-VC for HEVC. Additional proposed coding tools can be integrated into the JEM software for testing using JVET common test conditions (CTCs).

SUMMARY

[0004] Systems and methods are proposed herein for coded block flag (CBF) signaling. Exemplary embodiments may include one or more of the following features.

[0005] In some embodiments, a hierarchical signaling method is used to signal the CBFs of chroma components for the quad-tree plus binary tree (QTBT) structure. Specifically, one CBF flag may be signaled at each QTBT node level for a particular chroma component, indicating whether any descendent QTBT leaf node under the current level is associated with a non-zero coefficient.

[0006] In some embodiments, for pictures using motion-compensated prediction (e.g., P/B-slices), a signal may be provided at the QT/BT root node indicating whether there are significant (non-zero) transform coefficients present in the descendent leaf nodes that originate from the current root node if that QT/BT has any descendent nodes. When the flag is equal to 1, the coefficients of the descendent leaf nodes under the current node may be signaled using the existing CBF signaling as described above; otherwise, no further residual information is transmitted and all the transform coefficients are inferred to be 0.

[0007] In some embodiments, redundancy removal methods are also employed to reduce the overhead of CBF signaling under certain circumstances where the CBF values can be inferred.

[0008] In an exemplary method, a video is coded in a bitstream, wherein the video comprises a plurality of pictures. In the method, each picture is coded as a plurality of blocks arranged as leaf nodes in at least one hierarchical QTBT structure, such that each leaf node is a descendent node of a respective parent node in at least one level. The structure may include multiple layers of parent nodes. For each of a plurality of the parent nodes, at least one coded block flag (CBF) is coded in the bitstream. The CBF for the parent node indicates whether non-zero residual transform coefficients are coded in the bitstream for any block that is a descendent node of the respective parent node.

[0009] In some such methods, at least one of the CBFs is a chroma CBF associated with a chroma component. The chroma CBF indicates whether non-zero residual transform coefficients are coded in the bitstream for the associated chroma component in any block that is a descendent node of the respective parent node. Chroma CBFs may be signaled at a plurality of levels of parent nodes. However, in some embodiments, a chroma CBF is coded at a given descendent node only if a chroma CBF of a parent node of that descendent node indicates that non-zero residual transform coefficients of the respective chroma component are coded in the bitstream for at least one block that is a descendent node of the parent node.

[0010] Separate chroma CBFs may be signaled for separate chroma components (e.g. a first and a second chroma component).

[0011] In some embodiments, for each picture coded using inter prediction, a root CBF is coded for a root node of a plurality of the QTBT structures in the picture. This root CBF indicates whether non-zero

residual transform coefficients are coded in the bitstream for any component of any block in the respective root node. In some embodiments, a root CBF is not signaled for the root node in which all blocks are coded in skip mode, and the use of skip mode itself indicates that no residual transform coefficients are coded in the bitstream for blocks that are coded in skip mode. In some embodiments, a root CBF is not signaled for QTBT structures in which at least one block is coded in merge mode, and the use of merge mode itself indicates that residual transform coefficients are coded in the bitstream at least for the block that is coded in merge mode.

[0012] In a further exemplary embodiment, a method is provided for decoding a video from a bitstream, where the video includes a plurality of pictures. Each picture is encoded as a plurality of blocks arranged as leaf nodes in at least one hierarchical QTBT structure. Each leaf node is a descendent node of a respective parent node in at least one level. For each of a plurality of the parent nodes, a decoder parses from the bitstream a CBF flag. The CBF indicates whether any corresponding descendent leaf nodes have non-zero residual transform coefficients. The decoder parses residual transform coefficients from the bitstream only for leaf nodes that are not descendent nodes of a parent node with a CBF indicating that no non-zero residual transform coefficients are present.

[0013] In some embodiments, systems using a processor and a non-transitory computer-readable medium are provided for storing and executing instructions to perform the operations described herein. In further exemplary embodiments, a non-transitory computer-readable storage medium stores a bitstream representing a video encoded using techniques described herein.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] A more detailed understanding may be had from the following description, presented by way of example in conjunction with the accompanying drawings.

[0015] FIG. 1 is a block diagram illustrating an example of a block-based video encoder.

[0016] FIG. 2 is a block diagram illustrating an example of a block-based video decoder.

[0017] FIG. 3 is a diagram of an example of eight directional prediction modes.

[0018] FIG. 4 is a diagram illustrating an example of 33 directional prediction modes and two non-directional prediction modes.

[0019] FIG. 5 is a diagram of an example of horizontal prediction.

[0020] FIG. 6 is a diagram of an example of the planar mode.

[0021] FIG. 7 is a diagram illustrating an example of motion prediction.

[0022] FIG. 8 is a diagram illustrating an example of block-level movement within a picture.

[0023] FIG. 9 is a diagram illustrating an example of a coded bitstream structure.

- [0024] FIG. 10 is a diagram illustrating an example communication system.
- [0025] FIG. 11 is a diagram illustrating an example wireless transmit/receive unit (WTRU).
- [0026] FIG. 12 illustrates an example of Quad-Tree Plus Binary-Tree (QTBT) block partitioning.
- [0027] FIG. 13 illustrates an example of coded block flag (CBF) signaling for QTBT block partitioning.

DETAILED DESCRIPTION

Block-Based Video Coding.

[0028] Exemplary embodiments disclosed herein operate using a block-based hybrid video coding framework such as that employed in HEVC and JEM. FIG. 1 is a block diagram of a generic block-based hybrid video encoding system. The input video signal 102 is processed block by block. In HEVC, extended block sizes (called a “coding unit” or CU) are used to efficiently compress high resolution (1080p and beyond) video signals. In HEVC, a CU can be up to 64x64 pixels. A CU can be further partitioned into prediction units (PU), for which separate prediction methods are applied. For each input video block (MB or CU), spatial prediction (160) and/or temporal prediction (162) may be performed. Spatial prediction (or “intra prediction”) uses pixels from the samples of already coded neighboring blocks (which are called reference samples) in the same video picture/slice to predict the current video block. Spatial prediction reduces spatial redundancy inherent in the video signal. Temporal prediction (also referred to as “inter prediction” or “motion compensated prediction”) uses reconstructed pixels from the already coded video pictures to predict the current video block. Temporal prediction reduces temporal redundancy inherent in the video signal. A temporal prediction signal for a given video block is usually signaled by one or more motion vectors which indicate the amount and the direction of motion between the current block and its reference block. Also, if multiple reference pictures are supported (as is the case for the recent video coding standards such as H.264/AVC or HEVC), then for each video block, its reference picture index is sent additionally; and the reference index is used to identify from which reference picture in the reference picture store (164) the temporal prediction signal comes. After spatial and/or temporal prediction, the mode decision block (180) in the encoder chooses the best prediction mode, for example based on a rate-distortion optimization method. The prediction block is then subtracted from the current video block (116); and the prediction residual is de-correlated using transform (104) and quantized (106). The quantized residual coefficients are inverse quantized (110) and inverse transformed (112) to form the reconstructed residual, which is then added back to the prediction block (126) to form the reconstructed video block. Further in-loop filtering such as de-blocking filter and Adaptive Loop Filters may be applied (166) on the reconstructed video block before it is put in the reference picture store (164) and used to code future video blocks. To form the output video bit-stream 120, coding mode (inter or intra), prediction mode information, motion information, and quantized residual coefficients are all sent to the entropy coding unit (108) to be further compressed and packed to form the bit-stream.

[0029] FIG. 2 is a block diagram of a block-based video decoder. The video bit-stream 202 is unpacked and entropy decoded at entropy decoding unit 208. The coding mode and prediction information are sent to either the spatial prediction unit 260 (if intra coded) or the temporal prediction unit 262 (if inter coded) to form the prediction block. The residual transform coefficients are sent to inverse quantization unit 210 and inverse transform unit 212 to reconstruct the residual block. The prediction block and the residual block are then added together at 226. The reconstructed block may further go through in-loop filtering before it is stored in reference picture store 264. The reconstructed video in reference picture store is then sent out to drive a display device, as well as used to predict future video blocks.

[0030] A video encoder and/or decoder (e.g., video encoder 100 or video decoder 200) may perform spatial prediction (e.g., which may be referred to as intra prediction). Spatial prediction may be performed by predicting from already coded neighboring pixels following one of a plurality of prediction directions (e.g., which may be referred to as directional intra prediction).

[0031] FIG. 3 is a diagram of an example of eight directional prediction modes. The eight directional prediction modes of FIG. 3 may be supported in H.264/AVC. As shown generally at 300 in FIG. 3, the nine modes (including DC mode 2) are:

- Mode 0: Vertical Prediction
- Mode 1: Horizontal prediction
- Mode 2: DC prediction
- Mode 3: Diagonal down-left prediction
- Mode 4: Diagonal down-right prediction
- Mode 5: Vertical-right prediction
- Mode 6: Horizontal-down prediction
- Mode 7: Vertical-left prediction
- Mode 8: Horizontal-up prediction

[0032] Spatial prediction may be performed on a video block of various sizes and/or shapes. Spatial prediction of a luma component of a video signal may be performed, for example, for block sizes of 4x4, 8x8, and 16x16 pixels (e.g., in H.264/AVC). Spatial prediction of a chroma component of a video signal may be performed, for example, for block size of 8x8 (e.g., in H.264/AVC). For a luma block of size 4x4 or 8x8, a total of nine prediction modes may be supported, for example, eight directional prediction modes and the DC mode (e.g., in H.264/AVC). Four prediction modes may be supported; horizontal, vertical, DC, and planar prediction, for example, for a luma block of size 16x16.

[0033] Furthermore, directional intra prediction modes and non-directional prediction modes may be supported.

[0034] FIG. 4 is a diagram illustrating an example of 33 directional prediction modes and two non-directional prediction modes. The 33 directional prediction modes and two non-directional prediction modes, shown generally at 400 in FIG. 4, may be supported by HEVC. Spatial prediction using larger block sizes may be supported. For example, spatial prediction may be performed on a block of any size, for example, of square block sizes of 4x4, 8x8, 16x16, 32x32, or 64x64. Directional intra prediction (e.g., in HEVC) may be performed with 1/32-pixel precision.

[0035] Non-directional intra prediction modes may be supported (e.g., in H.264/AVC, HEVC, or the like), for example, in addition to directional intra prediction. Non-directional intra prediction modes may include the DC mode and/or the planar mode. For the DC mode, a prediction value may be obtained by averaging the available neighboring pixels and the prediction value may be applied to the entire block uniformly. For the planar mode, linear interpolation may be used to predict smooth regions with slow transitions. H.264/AVC may allow for use of the planar mode for 16x16 luma blocks and chroma blocks.

[0036] An encoder (e.g., the encoder 100) may perform a mode decision (e.g., at block 180 in FIG. 1) to determine the best coding mode for a video block. When the encoder determines to apply intra prediction (e.g., instead of inter prediction), the encoder may determine an optimal intra prediction mode from the set of available modes. The selected directional intra prediction mode may offer strong hints as to the direction of any texture, edge, and/or structure in the input video block.

[0037] FIG. 5 is a diagram of an example of horizontal prediction (e.g., for a 4x4 block), as shown generally at 500 in FIG. 5. Already reconstructed pixels P0, P1, P2 and P3 (*i.e.*, the shaded boxes) may be used to predict the pixels in the current 4x4 video block. In horizontal prediction, a reconstructed pixel, for example, pixels P0, P1, P2 and/or P3, may be propagated horizontally along the direction of a corresponding row to predict the 4x4 block. For example, prediction may be performed according to Equation (1) below, where $L(x, y)$ may be the pixel to be predicted at (x, y) , $x, y = 0 \dots 3$.

$$\begin{aligned}
 L(x, 0) &= P0 \\
 L(x, 1) &= P1 \\
 L(x, 2) &= P2 \\
 L(x, 3) &= P3
 \end{aligned}
 \tag{1}$$

[0038] FIG. 6 is a diagram of an example of the planar mode, as shown generally at 600 in FIG. 6. The planar mode may be performed accordingly: the rightmost pixel in the top row (marked by a T) may be replicated to predict pixels in the rightmost column. The bottom pixel in the left column (marked by an L) may be replicated to predict pixels in the bottom row. Bilinear interpolation in the horizontal direction (as shown in the left block) may be performed to produce a first prediction $H(x, y)$ of center pixels. Bilinear interpolation in the vertical direction (e.g., as shown in the right block) may be performed to produce a second prediction

$V(x,y)$ of center pixels. An averaging between the horizontal prediction and the vertical prediction may be performed to obtain a final prediction $L(x,y)$, using $L(x,y) = ((H(x,y)+V(x,y))>>1)$.

[0039] FIG. 7 and FIG. 8 are diagrams illustrating, as shown generally at 700 and 800, an example of motion prediction of video blocks (e.g., using temporal prediction unit 162 of FIG. 1). FIG. 8, which illustrates an example of block-level movement within a picture, is a diagram illustrating an example decoded picture buffer including, for example, reference pictures “Ref pic 0,” “Ref pic 1,” and “Ref pic2.” The blocks B0, B1, and B2 in a current picture may be predicted from blocks in reference pictures “Ref pic 0,” “Ref pic 1,” and “Ref pic2” respectively. Motion prediction may use video blocks from neighboring video frames to predict the current video block. Motion prediction may exploit temporal correlation and/or remove temporal redundancy inherent in the video signal. For example, in H.264/AVC and HEVC, temporal prediction may be performed on video blocks of various sizes (e.g., for the luma component, temporal prediction block sizes may vary from 16x16 to 4x4 in H.264/AVC, and from 64x64 to 4x4 in HEVC). With a motion vector of (mvx, mvy), temporal prediction may be performed as provided by equation (2):

$$P(x, y) = ref(x - mvx, y - mvy) \quad (2)$$

where $ref(x,y)$ may be pixel value at location (x, y) in the reference picture, and $P(x,y)$ may be the predicted block. A video coding system may support inter-prediction with fractional pixel precision. When a motion vector (mvx, mvy) has fractional pixel value, one or more interpolation filters may be applied to obtain the pixel values at fractional pixel positions. Block-based video coding systems may use multi-hypothesis prediction to improve temporal prediction, for example, where a prediction signal may be formed by combining a number of prediction signals from different reference pictures. For example, H.264/AVC and/or HEVC may use bi-prediction that may combine two prediction signals. Bi-prediction may combine two prediction signals, each from a reference picture, to form a prediction, such as the following equation (3):

$$P(x, y) = \frac{P_0(x, y) + P_1(x, y)}{2} = \frac{ref_0(x - mvx_0, y - mvy_0) + ref_1(x - mvx_1, y - mvy_1)}{2} \quad (3)$$

where $P_0(x, y)$ and $P_1(x, y)$ may be the first and the second prediction block, respectively. As illustrated in equation (3), the two prediction blocks may be obtained by performing motion-compensated prediction from two reference pictures $ref_0(x, y)$ and $ref_1(x, y)$, with two motion vectors (mvx_0, mvy_0) and (mvx_1, mvy_1) , respectively. The prediction block $P(x,y)$ may be subtracted from the source video block (e.g., at 116) to form a prediction residual block. The prediction residual block may be transformed (e.g., at transform unit 104) and/or quantized (e.g., at quantization unit 106). The quantized residual transform coefficient blocks may be sent to an entropy coding unit (e.g., entropy coding unit 108) to be entropy coded to reduce bit rate. The

entropy coded residual coefficients may be packed to form part of an output video bitstream (e.g., bitstream 120).

[0040] A single layer video encoder may take a single video sequence input and generate a single compressed bit stream transmitted to the single layer decoder. A video codec may be designed for digital video services (e.g., such as but not limited to sending TV signals over satellite, cable and terrestrial transmission channels). With video centric applications deployed in heterogeneous environments, multi-layer video coding technologies may be developed as an extension of the video coding standards to enable various applications. For example, multiple layer video coding technologies, such as scalable video coding and/or multi-view video coding, may be designed to handle more than one video layer where each layer may be decoded to reconstruct a video signal of a particular spatial resolution, temporal resolution, fidelity, and/or view. Although a single layer encoder and decoder are described with reference to FIG. 1 and FIG. 2, the concepts described herein may utilize a multiple layer encoder and/or decoder, for example, for multi-view and/or scalable coding technologies.

[0041] FIG. 9 is a diagram illustrating an example of a coded bitstream structure. A coded bitstream 1300 consists of a number of NAL (Network Abstraction layer) units 1301. A NAL unit may contain coded sample data such as coded slice 1306, or high level syntax metadata such as parameter set data, slice header data 1305 or supplemental enhancement information data 1307 (which may be referred to as an SEI message). Parameter sets are high level syntax structures containing essential syntax elements that may apply to multiple bitstream layers (e.g. video parameter set 1302 (VPS)), or may apply to a coded video sequence within one layer (e.g. sequence parameter set 1303 (SPS)), or may apply to a number of coded pictures within one coded video sequence (e.g. picture parameter set 1304 (PPS)). The parameter sets can be either sent together with the coded pictures of the video bit stream, or sent through other means (including out-of-band transmission using reliable channels, hard coding, etc.). Slice header 1305 is also a high level syntax structure that may contain some picture-related information that is relatively small or relevant only for certain slice or picture types. SEI messages 1307 carry the information that may not be needed by the decoding process but can be used for various other purposes such as picture output timing or display as well as loss detection and concealment.

[0042] FIG. 10 is a diagram illustrating an example of a communication system. The communication system 1400 may comprise an encoder 1402, a communication network 1404, and a decoder 1406. The encoder 1402 may be in communication with the network 1404 via a connection 1408, which may be a wireline connection or a wireless connection. The encoder 1402 may be similar to the block-based video encoder of FIG. 1. The encoder 1402 may include a single layer codec (e.g., FIG. 1) or a multilayer codec. For example, the encoder 1402 may be a multi-layer (e.g., two-layer) scalable coding system with picture-level ILP support. The decoder 1406 may be in communication with the network 1404 via a connection 1410,

which may be a wireline connection or a wireless connection. The decoder 1406 may be similar to the block-based video decoder of FIG. 2. The decoder 1406 may include a single layer codec (e.g., FIG. 2) or a multilayer codec. For example, the decoder 1406 may be a multi-layer (e.g., two-layer) scalable decoding system with picture-level ILP support.

[0043] The encoder 1402 and/or the decoder 1406 may be incorporated into a wide variety of wired communication devices and/or wireless transmit/receive units (WTRUs), such as, but not limited to, digital televisions, wireless broadcast systems, a network element/terminal, servers, such as content or web servers (e.g., such as a Hypertext Transfer Protocol (HTTP) server), personal digital assistants (PDAs), laptop or desktop computers, tablet computers, digital cameras, digital recording devices, video gaming devices, video game consoles, cellular or satellite radio telephones, digital media players, and/or the like.

[0044] The communications network 1404 may be a suitable type of communication network. For example, the communications network 1404 may be a multiple access system that provides content, such as voice, data, video, messaging, broadcast, etc., to multiple wireless users. The communications network 1404 may enable multiple wireless users to access such content through the sharing of system resources, including wireless bandwidth. For example, the communications network 1404 may employ one or more channel access methods, such as code division multiple access (CDMA), time division multiple access (TDMA), frequency division multiple access (FDMA), orthogonal FDMA (OFDMA), single-carrier FDMA (SC-FDMA), and/or the like. The communication network 1404 may include multiple connected communication networks. The communication network 1404 may include the Internet and/or one or more private commercial networks such as cellular networks, WiFi hotspots, Internet Service Provider (ISP) networks, and/or the like.

[0045] FIG. 11 is a system diagram of an example WTRU. As shown the example WTRU 1500 may include a processor 1518, a transceiver 1520, a transmit/receive element 1522, a speaker/microphone 1524, a keypad or keyboard 1526, a display/touchpad 1528, non-removable memory 1530, removable memory 1532, a power source 1534, a global positioning system (GPS) chipset 1536, and/or other peripherals 1538. It will be appreciated that the WTRU 1500 may include any sub-combination of the foregoing elements while remaining consistent with an embodiment. Further, a terminal in which an encoder (e.g., encoder 100) and/or a decoder (e.g., decoder 200) is incorporated may include some or all of the elements depicted in and described herein with reference to the WTRU 1500 of FIG. 11.

[0046] The processor 1518 may be a general purpose processor, a special purpose processor, a conventional processor, a digital signal processor (DSP), a graphics processing unit (GPU), a plurality of microprocessors, one or more microprocessors in association with a DSP core, a controller, a microcontroller, Application Specific Integrated Circuits (ASICs), Field Programmable Gate Array (FPGAs) circuits, any other type of integrated circuit (IC), a state machine, and the like. The processor 1518 may perform signal coding, data processing, power control, input/output processing, and/or any other functionality that enables the

WTRU 1500 to operate in a wired and/or wireless environment. The processor 1518 may be coupled to the transceiver 1520, which may be coupled to the transmit/receive element 1522. While FIG. 11 depicts the processor 1518 and the transceiver 1520 as separate components, it will be appreciated that the processor 1518 and the transceiver 1520 may be integrated together in an electronic package and/or chip.

[0047] The transmit/receive element 1522 may be configured to transmit signals to, and/or receive signals from, another terminal over an air interface 1515. For example, in one or more embodiments, the transmit/receive element 1522 may be an antenna configured to transmit and/or receive RF signals. In one or more embodiments, the transmit/receive element 1522 may be an emitter/detector configured to transmit and/or receive IR, UV, or visible light signals, for example. In one or more embodiments, the transmit/receive element 1522 may be configured to transmit and/or receive both RF and light signals. It will be appreciated that the transmit/receive element 1522 may be configured to transmit and/or receive any combination of wireless signals.

[0048] In addition, although the transmit/receive element 1522 is depicted in FIG. 15 as a single element, the WTRU 1500 may include any number of transmit/receive elements 1522. More specifically, the WTRU 1500 may employ MIMO technology. Thus, in one embodiment, the WTRU 1500 may include two or more transmit/receive elements 1522 (e.g., multiple antennas) for transmitting and receiving wireless signals over the air interface 1515.

[0049] The transceiver 1520 may be configured to modulate the signals that are to be transmitted by the transmit/receive element 1522 and/or to demodulate the signals that are received by the transmit/receive element 1522. As noted above, the WTRU 1500 may have multi-mode capabilities. Thus, the transceiver 1520 may include multiple transceivers for enabling the WTRU 1500 to communicate via multiple RATs, such as UTRA and IEEE 802.11, for example.

[0050] The processor 1518 of the WTRU 1500 may be coupled to, and may receive user input data from, the speaker/microphone 1524, the keypad 1526, and/or the display/touchpad 1528 (e.g., a liquid crystal display (LCD) display unit or organic light-emitting diode (OLED) display unit). The processor 1518 may also output user data to the speaker/microphone 1524, the keypad 1526, and/or the display/touchpad 1528. In addition, the processor 1518 may access information from, and store data in, any type of suitable memory, such as the non-removable memory 1530 and/or the removable memory 1532. The non-removable memory 1530 may include random-access memory (RAM), read-only memory (ROM), a hard disk, or any other type of memory storage device. The removable memory 1532 may include a subscriber identity module (SIM) card, a memory stick, a secure digital (SD) memory card, and the like. In one or more embodiments, the processor 1518 may access information from, and store data in, memory that is not physically located on the WTRU 1500, such as on a server or a home computer (not shown).

[0051] The processor 1518 may receive power from the power source 1534, and may be configured to distribute and/or control the power to the other components in the WTRU 1500. The power source 1534 may be any suitable device for powering the WTRU 1500. For example, the power source 1534 may include one or more dry cell batteries (e.g., nickel-cadmium (NiCd), nickel-zinc (NiZn), nickel metal hydride (NiMH), lithium-ion (Li-ion), etc.), solar cells, fuel cells, and the like.

[0052] The processor 1518 may be coupled to the GPS chipset 1536, which may be configured to provide location information (e.g., longitude and latitude) regarding the current location of the WTRU 1500. In addition to, or in lieu of, the information from the GPS chipset 1536, the WTRU 1500 may receive location information over the air interface 1515 from a terminal (e.g., a base station) and/or determine its location based on the timing of the signals being received from two or more nearby base stations. It will be appreciated that the WTRU 1500 may acquire location information by way of any suitable location-determination method while remaining consistent with an embodiment.

[0053] The processor 1518 may further be coupled to other peripherals 1538, which may include one or more software and/or hardware modules that provide additional features, functionality and/or wired or wireless connectivity. For example, the peripherals 1538 may include an accelerometer, orientation sensors, motion sensors, a proximity sensor, an e-compass, a satellite transceiver, a digital camera and/or video recorder (e.g., for photographs and/or video), a universal serial bus (USB) port, a vibration device, a television transceiver, a hands free headset, a Bluetooth® module, a frequency modulated (FM) radio unit, and software modules such as a digital music player, a media player, a video game player module, an Internet browser, and the like.

[0054] By way of example, the WTRU 1500 may be configured to transmit and/or receive wireless signals and may include user equipment (UE), a mobile station, a fixed or mobile subscriber unit, a pager, a cellular telephone, a personal digital assistant (PDA), a smartphone, a laptop, a netbook, a tablet computer, a personal computer, a wireless sensor, consumer electronics, or any other terminal capable of receiving and processing compressed video communications.

[0055] The WTRU 1500 and/or a communication network (e.g., communication network 804) may implement a radio technology such as Universal Mobile Telecommunications System (UMTS) Terrestrial Radio Access (UTRA), which may establish the air interface 1515 using wideband CDMA (WCDMA). WCDMA may include communication protocols such as High-Speed Packet Access (HSPA) and/or Evolved HSPA (HSPA+). HSPA may include High-Speed Downlink Packet Access (HSDPA) and/or High-Speed Uplink Packet Access (HSUPA). The WTRU 1500 and/or a communication network (e.g., communication network 804) may implement a radio technology such as Evolved UMTS Terrestrial Radio Access (E-UTRA), which may establish the air interface 1515 using Long Term Evolution (LTE) and/or LTE-Advanced (LTE-A).

[0056] The WTRU 1500 and/or a communication network (e.g., communication network 804) may implement radio technologies such as IEEE 802.16 (e.g., Worldwide Interoperability for Microwave Access (WiMAX)), CDMA2000, CDMA2000 1X, CDMA2000 EV-DO, Interim Standard 2000 (IS-2000), Interim Standard 95 (IS-95), Interim Standard 856 (IS-856), Global System for Mobile communications (GSM), Enhanced Data rates for GSM Evolution (EDGE), GSM EDGE (GERAN), and the like. The WTRU 1500 and/or a communication network (e.g., communication network 804) may implement a radio technology such as IEEE 802.11, IEEE 802.15, or the like.

Quad-Tree Plus Binary-Tree (QTBT) Partitioning.

[0057] In HEVC, a picture is split into CUs based on a quad-tree structure that allows for splitting the CUs into an appropriate size based on the signal characteristics of the region. The CU represents the basic quad-tree split region that is used to differentiate intra and inter coded blocks. Inside a CU, multiple non-overlapping PUs can be defined, each of which specifies a region with individual prediction parameters (e.g., intra prediction mode, motion vector, reference picture index and so forth). After obtaining the residuals by applying the prediction process to the PUs, the CU is further split to TUs based on another quad-tree, each TU specifying the block of applying residual coding with transform size equal to the TU size.

[0058] Although the above block partitioning structure of HEVC provides significant coding gain over the previous video coding standards, there is still space for further improvement for various reasons. First, CU partitions with the minimum granularity for switching between intra and inter coding are square and follow a quad-tree structure. The use of square blocks in a quad-tree structure may not be flexible enough to adapt to the various local characteristics in a picture. Second, PU partitions only have a limited number of types which may be inefficient to capture the geometric structure of 2D data. Third, the multiple concepts of CU, PU and TU may be redundant in certain regions in a picture which may introduce unnecessary signaling overhead and increase encoding/decoding complexity.

[0059] To address at least some of the abovementioned issues, a quad-tree plus binary-tree (QTBT) block partitioning structure has been proposed in J. An, Y.-W. Chen, K. Zhang, H. Huang, Y.-W. Huang, S. Lei, "Block partitioning structure for next generation video coding", COM16-C966R3-E, Sept., 2015. In an exemplary QTBT structure, each coding tree unit (CTU) which is the root node of quad-tree is firstly partitioned in the quad-tree manner, where the quad-tree splitting of one node can be iterated until the node reaches the minimum of the allowed quad-tree size (MinQTSIZE). If the quad-tree node size is no larger than the maximum of the allowed binary tree size (MaxBTSIZE), it can be further partitioned by a binary tree in either horizontal or vertical direction. The splitting of the binary tree can be iterated until the binary tree node reaches the minimum of the allowed binary tree node size (MinBTSIZE) or the maximum of the allowed binary tree depth. The binary tree node is used as the basic unit of both prediction and transform without any further partitioning (such that the concepts of PU and TU are not employed).

[0060] To help understand a QTBT partitioning structure, as one example, consider a case in which the CTU size is 128x128, MinQTSIZE is 16x16, MaxBTSIZE is 64x64 and MinBTSIZE is 4. The quad-tree partitioning is firstly applied to the CTU to generate quad-tree leaf nodes. The quad-tree leaf node size may range from 128x128 to 16x16. If the quad-tree node is 128x128, then it will not be split by the binary tree as it exceeds the maximum binary tree size (MaxBTSIZE). Otherwise, the quad-tree node will be further partitioned by the binary tree. As the quad-tree node is also the root node of the binary tree, its binary tree depth is equal to 0. The binary tree partitioning can be iterated until it reaches the binary tree depth reaches MaxBTDepth or the binary tree node has width or height equal to MinBTSIZE. FIG. 12 illustrates one example of QTBT block partitioning where the solid lines represent quad-tree splitting and the dotted lines represent binary tree splitting. As shown in FIG. 12, if one binary tree node is further split, a flag is signaled to indicate whether horizontal or vertical splitting is used. For quad-tree splitting, no overhead needs to be signaled as it always partitions a block into four sub-blocks with an equal size. In July, 2016, the QTBT was adopted as the basic coding structure of JEM-3.0.

Limitations of QTBT.

[0061] In in JEM-3.0, the QTBT structure is applied to represent both prediction and transform information of a CTU. The QTBT for each CTU comprises a set of nodes. The node at the highest level is referred to as a “root node” which corresponds to a QT node that is further partitioned into multiple sub-blocks. The node at the lowest level which is not further split (e.g., one QT/BT without additional BT partitions) is referred to as a “leaf node”. If one BT node at lower level is split from another QT/BT node at higher level, the QT/BT node at the higher level is referred to as a “parent node” or “parent” of the BT at the lower level while the BT at the lower level is referred to as a “descendent node” or a “descendant” of the QT/BT node at the higher level.

[0062] In QTBT, as in HEVC, a coded block flag (CBF) signals the significance of each leaf node, indicating whether the QTBT leaf node contains nonzero transform coefficients. There are separate CBFs for luma and each of the two chroma components. If the CBF of one color component is equal to 0, no transform coefficient information is signaled for the component and all the transform coefficients are set to 0; otherwise, the transform coefficients are coded into bit-stream. FIG. 13 illustrates an example of a QTBT block partition with the same structure illustrated in FIG. 12. As illustrated in FIG. 13, leaf nodes 1350 and 1352 have nonzero transform coefficients, while the remaining leaf nodes in the QTBT have no transform coefficients. One way to signal whether the leaf nodes have nonzero transform coefficients is to signal a CBF for each of the twelve leaf nodes. The CBFs signaled using such a technique may be represented as follows, with four sets of bins corresponding to the BTs in the four QT root nodes:

0, 0;

0, 0, 1;

1;

0, 0, 0, 0, 0, 0

[0063] The CBF bits are signaled in a raster-scan order of BTs, in the order of upper BT to lower BT, or left BT to right BT.

[0064] In JEM-3.0, different CBF signaling methods are applied to intra-coded and inter-coded blocks. For QTBT leaf nodes using motion-compensated prediction, one single CBF (`root_cbf_flag`) is signaled to indicate whether at least one nonzero coefficient exists in any of the color components of the respective leaf node (considering three color components together). When `root_cbf_flag` is equal to 0, no further transform coefficient is transmitted and all the residuals are set to 0; otherwise, additional CBF flags are transmitted for luma and two chroma components separately. For QTBT leaf nodes using intra prediction, the `root_cbf_flag` is inferred to be 1.

[0065] As can be seen from the above, known techniques for CBF signaling in QTBT are capable of reducing the bits of residual coding by generating one flag which indicates that the whole QT/BT includes no non-zero coefficients rather than signal each of the zero-valued coefficients of the block. However, there are still several problems with such techniques.

[0066] First, since the human vision system is much more sensitive to variations in brightness than color, a video coding system usually allocates more bits to the luma component than chroma components, e.g., by adjusting the quantization parameter (QP) offset value between luma and chroma components. Moreover, since chroma components usually have smaller dynamic range, more chroma coefficients become zeros after quantization. Consequently, it is highly possible that all the sub-partitions, which are leaf nodes under one QTBT node have no non-zero coefficients for one chroma component. Based on the existing design, one CBF flag (which is equal to 0) is signaled for each of those BTs for each chroma component. However, it may be more efficient to generate only one flag that indicates all the chroma coefficients under a given QTBT node include no non-zero coefficients.

[0067] Second, in the current CBF design of the QTBT, for each inter-coded BT, one `root_cbf_flag` is signaled to indicate whether at least one non-zero coefficient is transmitted for the BT. Given that the current QTBT allows more flexible block partitions for motion-compensated prediction, this can increase the precision of motion-compensated prediction and therefore reduce the energy of the prediction errors for inter-coded blocks. Therefore, it may frequently happen that all the QT/BT leaf nodes under a given QT/BT node contains no non-zero coefficients. In this case, it may be more efficient to signal only one flag that indicates all the luma and chroma coefficients under the QT/BT node are all zeros than to signal one `root_cbf_flag` as 0 for each of the corresponding leaf nodes.

Overview of Exemplary Embodiments.

[0068] To address issues identified above, systems and methods are proposed herein to improve CBF signaling and thus enhance the overall coding efficiency of the QTBT block partitioning structure. Exemplary embodiments include one or more of the following features.

[0069] In some embodiments, a hierarchical signaling method is used to signal the CBFs of chroma components for the QTBT structure. Specifically, one CBF flag is signaled at each QT/BT node level for a particular chroma component, indicating whether any descendent QT/BT leaf node under the current level is associated with a non-zero coefficient.

[0070] In some embodiments, for pictures using motion-compensated prediction (e.g., P/B-slices), a signal is provided at the QT/BT root node indicating whether there are significant (non-zero) transform coefficients present in the descendent leaf nodes that originate from the current root node if that QT/BT node has any descendent nodes. When the flag is equal to 1, the coefficients of the descendent leaf nodes under the current node may be signaled using the existing CBF signaling as described above; otherwise, no further residual information is transmitted and all the transform coefficients are inferred to be 0.

[0071] In some embodiments, redundancy removal methods are also employed to reduce the overhead of CBF signaling under certain circumstances where the CBF values can be inferred.

Hierarchical Signaling of Chroma Components in QTBT.

[0072] As discussed above, in the current QTBT structure, one CBF flag is signaled for each of two chroma components for each BT. However, as the dynamic range of chroma residuals are usually much smaller than that of luma component, a large number of chroma transform coefficients will be quantized to 0 before being sent to the bit-stream, especially at low bit-rate (e.g. high QPs). If all the descendent leaf nodes under a certain parent node of a QTBT structure do not contain non-zero coefficients at all, it may be more efficient in terms of signaling overhead to use only a flag indicating that all the blocks originating from the current node only include zero coefficients rather than to signal one CBF flag equal to 0 for each respective individual BT.

[0073] To improve efficiency of signaling, in at least one embodiment disclosed herein, a hierarchical signaling method is used to signal the chroma CBFs for the QTBT structure, where the chroma CBFs are signaled at each descendent node level that originates from the same certain QT/BT node. In such embodiments, the signaling of chroma CBFs are performed not only for the leaf nodes but also for each parent node level of the same QTBT tree. When the chroma CBF flag at a given QTBT level is 0, this indicates that all the chroma residuals of the leaf nodes of the current level are equal to 0 and no other transform coefficient information is transmitted; otherwise (if the chroma CBF at the current level is 1), depending on whether the current node is further partitioned by a quad-tree or a binary-tree, four or two additional CBFs are further signaled for the chroma component, each indicating whether one of sub-block partitions for the

current node has any non-zero transform coefficient. The above hierarchical chroma CBF signaling is iterated until a leaf node is reached for the QTBT structure.

[0074] With reference to the exemplary QTBT partitions of FIG. 13, an exemplary embodiment provides for hierarchical CBF signaling for chroma components. In the exemplary embodiment, CBF bits of “0110” are signaled at the QT/BT root node level (at depth 0) to indicate whether each of the four BTs contains a non-zero coefficient. Further, since the upper-right BT contains non-zero coefficients and is further split by a horizontal binary-tree, additional CBF bits of “01” are signaled at the next level (depth 1) to indicate whether two sub-BT partitions comprise non-zero coefficients. For the lower-left BT, although the CBF of its BT root node is 1, no additional CBF bits are signaled as it is not further split. Whereas a known signaling technique uses twelve bins for CBF signaling of the example in FIG. 13, the proposed hierarchical CBF signaling only consumes 6 bins to indicate the CBFs of the chroma components for the same QTBT structure:

0; 1; 1; 0;

and

0, 1.

[0075] In the exemplary embodiment described above, the proposed hierarchical CBF signaling is only used for chroma components while the luma CBF signaling is unchanged (such that the CBF flag of the luma component is only signaled at a QTBT leaf node). This may complicate the design as different CBF signaling methods are used for luma and chroma components separately. To simplify the CBF signaling design, in another embodiment, luma and chroma CBF signaling are unified by extending the hierarchical CBF signaling of the chroma component to signal luma CBF. Therefore, in such embodiments, the luma and chroma CBFs are both signaled for each node level of one QTBT tree.

Signaling the CBF of QT/BT Root Node for Inter Picture/Slice.

[0076] In the current CBF signaling of the QTBT, for each QT/BT leaf node that uses motion-compensated prediction, a single flag `root_cbf_flag` signals whether the transform coefficients need to be transmitted for that leaf node. When `root_cbf_flag` is equal to 1, the transform coefficients are normally signaled; otherwise (when `root_cbf_flag` is equal to 0), no further residual information is transmitted and all the transform coefficients of the QT/BT are set to 0. Although `root_cbf_flag` is very useful for the coding of QT/BT blocks which can be precisely predicted by motion compensated prediction (especially at low bit-rate), the existing design may not maximize the coding benefit that the syntax element can provide. The current QTBT allows more flexible block partitions (quad-partition plus binary-partition) for motion-compensated prediction, this can significantly improve the quality of temporal prediction and therefore reduce the energy of the prediction errors for residual coding. Correspondingly, it is often the case that all the QT/BT leaf nodes under a certain parent node contain no non-zero coefficients. In this case, it is more efficient to signal only

one flag indicating that all the luma and chroma coefficients under the current node are all zeros rather than to signal one root_cbf_flag equal to 0 for each of the corresponding leaf nodes.

[0077] In exemplary embodiments, for inter-coded pictures/slices, one flag is signaled at each QT/BT root node (each root node of a quad-tree and each root node of a binary tree) to indicate whether there are significant (non-zero) transform coefficients present in the descendent leaf nodes of the root node. In an embodiment, a single flag qtbt_root_node_cbf is provided at each root node to signal whether at least one non-zero coefficient exists for any of the color components the current QT/BT root node as a whole. When qtbt_node_root_cbf is equal to 0, no residual information is transmitted further and all the transform coefficients are inferred to be 0. Otherwise, the coefficients of the blocks under the current QT/BT root node are signaled based on the existing CBF signaling method. Specifically, when qtbt_node_root_cbf is equal to 1, the signaling process will go to each descendent leaf node of the root node and signal one root_cbf_flag (if the descendent leaf node is inter-coded) and specific CBF flags for each color component. If the CBF of one color component is equal to 1, the coefficients of the descendent leaf node are then coded into bit-stream.

[0078] An exemplary embodiment of root CBF signaling may be described with reference to the QTBT partition structure illustrated in FIG. 13, applied to both luma and chroma components. The numbers “0” and “1” in FIG. 13 represent the value of root_cbf_flag for each QT/BT, where “0” indicates that there is no non-zero coefficient for the QT/BT and “1” indicates that there is at least one non-zero coefficient for the QT/BT. As described above, with the use of a known root CBF signaling method, twelve bins are signaled to represent the root_cbf_flags of the twelve QT/BT leaf nodes in the QTBT structure, each being generated for one respective QT/BT leaf node. Alternatively, when an exemplary embodiment of the proposed root CBF signaling is applied, only seven bins are signaled:

0; 1; 1; 0;

and

0, 0, 1.

Specifically, the root CBF bits of “0110” are firstly signaled, which correspond to the values of qtbt_root_cbf for the four QT root nodes. For the upper-right QT, three additional root CBF bits “001” are further signaled, which correspond to the values of root_cbf_flag of the three BTs under the QT root node. For the lower-left QT, as it is not further split, no root_cbf_flag needs to be signaled.

[0079] In at least one embodiment, the proposed qtbt_root_cbf is always signaled for the QT/BT root node in inter-coded pictures/slices regardless of the prediction mode of each specific QT/BT leaf node under the same QT/BT root node (either intra-coded or inter-coded). In another embodiment of the disclosure, the proposed qtbt_root_cbf is only signaled for the QT/BT root node that contains at least one QT/BT leaf node which is predicted using motion compensated prediction. Otherwise (if all the QT/BT leaf nodes under the

root node are intra-coded), the value of `qtb_t_root_cbf` is inferred to be 1. Further, in another embodiment of the disclosure, the proposed `qtb_t_root_cbf` is only signaled for the QT/BT root node that contains only QT/BT leaf nodes that are predicted using motion compensated prediction. Otherwise (if at least one QT/BT leaf node under the root node is intra-coded), the value of the `qtb_t_root_cbf` is inferred to be 1. In the last two cases, the decision whether to signal or infer the value of `qtb_t_root_cbf` for the current QT/BT root node is dependent on the prediction modes of all the leaf nodes that starts from the current QT/BT.

[0080] In the existing signaling design of QTBT-rated syntax elements, the signaling of the QTBT partition flags and the signaling of prediction mode and transform coefficients of each QTBT leaf node are interleaved. In such systems, the prediction modes of QT/BT leaf nodes are unknown before the parsing process proceeds to that QT/BT node. In other words, the decoder has no access to the prediction mode information of each QT/BT leaf node when one QT root node is just parsed from the bit-stream. Such a coder may be incapable of determining if the `qtb_t_root_cbf` should be parsed next. In order to address this parsing dependency problem, it is proposed in exemplary embodiments to separate the parsing of the QTBT partitions and the prediction modes of each QT/BT leaf node from the parsing of transform coefficients. Specifically, in the first parsing loop, all the syntax elements except for CBF flags and the ones that are related with transform coefficient signaling are signaled/parsed. Then, in the second parsing loop, the CBF flags and the transform coefficients are parsed based on the QTBT partitions and the prediction modes that are obtained from the first pass. This change only modifies the signaling order of the syntax elements and may have no impact on coding performance.

[0081] In exemplary methods discussed above, the `qtb_t_root_cbf` flag is used to indicate whether there are non-zero transform coefficients present in the descendent leaf nodes of one root node. In one embodiment of the disclosure, the proposed `qtb_t_root_cbf` flag is only used to indicate whether there are non-zero transform coefficients in the descendent leaf nodes which are coded using motion-compensated prediction (inter-coded).

CBF Signaling Redundancy Removal.

[0082] Under certain circumstances, the proposed CBF signaling as discussed in the section “Hierarchical signaling of chroma components in QTBT” for hierarchical chroma CBFs and in the section “Signaling the CBF of QT/BT Root Node for Inter Picture/Slice” for `qtb_t_root_cbf` are redundant and can be inferred instead of being explicitly signaled. Specifically, in the following circumstances, CBF signaling may be redundant:

1. In JEM-3.0, if one QT/BT is coded in SKIP mode, this indicates that no transform coefficients need to be transmitted and all the residuals are inferred to zeros. Thus, if all the QT/BT leaf nodes under a QT/BT root node are coded in SKIP mode, there is no need to transmit the `qtb_t_root_cbf` flag and the chroma CBF for the root node as they are always 0.

2. In JEM-3.0 if one QT/BT is coded in MERGE mode, this indicates that there is at least one non-zero coefficient transmitted (otherwise, the QT/BT block should be coded in SKIP mode). Therefore, if there are one or more QT/BT leaf nodes under a QT/BT root node which are coded in MERGE mode, there is no need to transmit the `qtb_t_root_cbf` flag as it is always 1.
3. Consider a case where there are N QT/BT leaf nodes under a QT root node and the corresponding `qtb_t_root_cbf` is equal to 1. If the `root_cbf_flag` flags of the first $N-1$ QT/BT leaf nodes are zeros, there is no need to signal the `root_cbf_flag` for the last QT/BT leaf node as it is constrained to be 1.
4. Similar to item 3, for chroma hierarchical chroma CBF signaling, if the CBF of the parent node level is 1 and the CBFs of the first $N-1$ sibling nodes at the same level are zeros, there is no need to signal the chroma CBF for the last node at the current level as it is constrained to be 1.

[0083] For item 2, the decision on whether one QT/BT is coded by MERGE mode may be utilized to avoid the redundant signaling of `qtb_t_root_cbf`. This redundancy removal method may also be implemented in an alternative way by making the signaling of the MERGE mode (indicated by the `merge_mode_flag`) dependent on the value of `qtb_t_root_cbf`. Specifically, when the value of the `qtb_t_root_cbf` for one QT/BT root node is equal to 0, there is no need to signal the value of the `merge_mode_flag` for each QT/BT leaf node under the QT/BT as they have to be 0 (i.e., non-MERGE mode).

Additional Embodiments of CBF Signaling.

[0084] Although embodiments of the hierarchical CBF signaling the `qtb_t_root_cbf` signaling as discussed above are carried out from the QT/BT root level, signaling methods according to the present disclosure can be performed at various coding levels, such as CTU level, arbitrary QT level, or arbitrary BT level. Each signaling level may provide a different trade-off between coding efficiency and encoding/decoding complexity. For example, for QTs under which all the descendent BTs have no non-zero coefficients, it is more efficient to put the `qtb_t_root_cbf` flag at the QT level given that one single flag can be used to represent the transform coefficients in the whole QT. However, for QTs which comprise some BTs that have non-zero coefficients and some BTs that have no non-zero coefficients, it may be more beneficial to put the `qtb_t_root_cbf` flag at a certain BT level.

[0085] To improve the coding gain in embodiments using the proposed CBF signaling, an encoder may conduct additional rate-distortion (RD) tests by setting the proposed CBF flag to 0 (to force all the transform coefficients to be 0). This can also increase the encoding complexity which could become severe when the proposed CBF syntax element is placed at a higher QTBT level. In one embodiment of the disclosure, it is proposed to carry out the proposed CBF signaling from the first BT level under one QT root node. Specifically, for BT leaf nodes that are generated from a corresponding QT root node by two or more BT partitions, the corresponding CBFs are signaled by the proposed CBF signaling methods; otherwise (BT leaf nodes with

only one BT partition or QT leaf node without BT partition), the default CBF signaling in current QTBT may be applied.

[0086] In another embodiment of this disclosure, region-based selection of the coding level for the proposed CBF signaling may be applied. For example, high coding level (e.g., CTU level) may be selected for flat regions or the regions with slow motion which are more likely to be precisely predicted and therefore has more all-zeros coding blocks; alternatively, a lower coding level (e.g. a given BT level) may be selected for the regions with more textures or high motion, which usually lead to more non-zero coefficients due to the reduced prediction quality.

Reduction of Encoding Complexity.

[0087] As discussed above, to improve the coding gain in embodiments using the proposed CBF signaling, an encoder may perform additional RD testing by setting the proposed CBF flag (e.g., `qtb_t_root_cbf`) to 0, forcing all the transform coefficients to be 0. This could introduce a non-negligible increase in encoding complexity. In order to address this encoding complexity issue, one or more of the following techniques may be used.

1. It is proposed in some embodiments to firstly check the RD cost without forcibly setting the CBF flag to 0 and then to check the RD cost with setting the CBF flag to 0. In addition, the calculation of the second RD cost is only conducted if there is at least one non-zero coefficient for the first RD test.
2. In order to reduce the number of tested coding modes, it is proposed in some embodiments to use the same coding modes for the RD test without setting CBF to 0 and the RD test with setting CBF to 0. More specifically, for intra mode, the selected luma and chroma intra predictions are shared for two RD tests; for inter mode, the selected motion vector, reference picture and motion vector predictor are shared between two RD tests.
3. The quality of inter prediction, which uses temporal reference samples, is often better than the quality of intra prediction, which uses the reference samples from spatial neighbors. As a result, the coefficients of inter-coded blocks are more likely to be zero than those of intra-coded blocks. Therefore, in order to reduce the encoding complexity, when the CBF flag is forced to be 0, it is proposed in some embodiments to only calculate the RD costs for inter modes while the RD calculation of intra modes is skipped.

[0088] Although features and elements are described above in particular combinations, one of ordinary skill in the art will appreciate that each feature or element can be used alone or in any combination with the other features and elements. In addition, the methods described herein may be implemented in a computer program, software, or firmware incorporated in a computer-readable medium for execution by a computer or processor. Examples of computer-readable storage media include, but are not limited to, a read only memory (ROM), a random access memory (RAM), a register, cache memory, semiconductor memory devices,

magnetic media such as internal hard disks and removable disks, magneto-optical media, and optical media such as CD-ROM disks, and digital versatile disks (DVDs). A processor in association with software may be used to implement a radio frequency transceiver for use in a WTRU, UE, terminal, base station, RNC, or any host computer.

CLAIMS

What is claimed:

1. A method of decoding a video, the method comprising:

receiving a bitstream encoding the video, wherein the video comprises a plurality of pictures, each picture encoded as a plurality of blocks arranged as leaf nodes in at least one hierarchical quad-tree plus binary-tree (QTBT) structure, each leaf node being a descendent node of a respective parent node in at least one level;

for each of a plurality of the parent nodes, parsing from the bitstream a coded block flag (CBF) indicating whether any corresponding descendent blocks have non-zero residual transform coefficients; and

for a plurality of blocks, determining, based at least in part on the CBF of at least one parent of the respective block, whether residual transform coefficients for the block are coded in the bitstream.

2. The method of claim 1, further comprising parsing residual transform coefficients from the bitstream only for blocks that are not descendent nodes of a parent node with a CBF indicating that no non-zero residual transform coefficients are present.

3. The method of claim 1 or 2, wherein at least one of the CBFs is a chroma CBF associated with a chroma component, the chroma CBF indicating whether non-zero residual transform coefficients are coded in the bitstream for the associated chroma component in any block that is a descendent node of the respective parent node.

4. The method of claim 3, wherein chroma CBFs are coded at a plurality of levels of parent nodes.

5. The method of claim 3, wherein a chroma CBF is coded at a descendent node only if the chroma CBF of the respective parent node of the descendent node indicates that non-zero residual transform coefficients are coded in the bitstream for at least one block that is a descendent node of the parent node.

6. The method of claim 3, wherein (i) a first chroma CBF of a parent node indicates whether non-zero residual transform coefficients of a first chroma component are signaled for any block that is a descendent node of the parent node; and (ii) a second chroma CBF of the parent node indicates whether non-zero residual transform coefficients of a second chroma component are signaled for any block that is a descendent node of the parent node.

7. The method of claim 1 or 2, wherein, for each of the pictures coded using inter prediction, a root CBF is parsed from the bitstream for a QT/BT root node of a plurality of the QTBT structures in the picture, the root

CBF indicating whether non-zero residual transform coefficients are coded in the bitstream for any component of any block that is a descendent node of the QT/BT root node.

8. The method of claim 7, wherein a root CBF is not parsed from the bitstream for QTBT structures in which all blocks are coded in skip mode, and wherein residual transform coefficients are not coded in the bitstream for blocks that are coded in skip mode.

9. The method of claim 7, wherein a root CBF is not parsed from the bitstream for QTBT structures in which at least one block is coded in merge mode, and wherein residual transform coefficients are coded in the bitstream at least for the block that is coded in merge mode.

10. A method of encoding a video in a bitstream, wherein the video comprises a plurality of pictures, the method comprising:

encoding each picture as a plurality of blocks arranged as leaf nodes in at least one hierarchical quad-tree plus binary-tree (QTBT) structure, each leaf node being a descendent node of a respective parent node in at least one level; and

for a plurality of the parent nodes, encoding at least one coded block flag (CBF) in the bitstream indicating whether non-zero residual transform coefficients are encoded in the bitstream for any block that is a descendent node of the respective parent node.

11. The method of claim 10, wherein at least one of the CBFs is a chroma CBF associated with a chroma component, the chroma CBF indicating whether non-zero residual transform coefficients are encoded in the bitstream for the associated chroma component in any block that is a descendent node of the respective parent node.

12. The method of claim 11, wherein chroma CBFs are signaled at a plurality of levels of parent nodes.

13. The method of claim 11, wherein a chroma CBF is encoded for a descendent node only if the chroma CBF of the respective parent node of the descendent node indicates that non-zero residual transform coefficients are encoded in the bitstream for at least one block that is a descendent node of the parent node.

14. The method of claim 11, wherein (i) a first chroma CBF of a parent node indicates whether non-zero residual transform coefficients of a first chroma component are encoded for any block that is a descendent node of the parent node; and (ii) a second chroma CBF of a parent node indicates whether non-zero residual transform coefficients of a second chroma component are encoded for any block that is a descendent node of the parent node.

15. The method of claim 10, wherein, for each picture encoded using inter prediction, a root CBF is encoded for a QT/BT root node of a plurality of the QTBT structures in the picture, the root CBF indicating whether non-zero residual transform coefficients are encoded in the bitstream for any component of any block that is a descendent node of the QT/BT root node.
16. The method of claim 15, wherein a root CBF is not encoded for QTBT structures in which all blocks are encoded in skip mode, and wherein residual transform coefficients are not encoded in the bitstream for blocks that are encoded in skip mode.
17. The method of claim 15, wherein a root CBF is not encoded for QTBT structures in which at least one block is encoded in merge mode, and wherein residual transform coefficients are encoded in the bitstream at least for the block that is encoded in merge mode.
18. A system comprising a processor and a non-transitory computer-readable storage medium storing instructions operative, when executed on the processor, to perform functions comprising:
- receiving a bitstream encoding a video, wherein the video comprises a plurality of pictures, each picture encoded as a plurality of blocks arranged as leaf nodes in at least one hierarchical quad-tree plus binary-tree (QTBT) structure, each leaf node being a descendent node of a respective parent node in at least one level;
 - for each of a plurality of the parent nodes, parsing from the bitstream a coded block flag (CBF) indicating whether any corresponding descendent leaf nodes have non-zero residual transform coefficients;
 - and
 - for each block, determining, based at least in part on the CBF of at least one parent of the respective block, whether residual transform coefficients for the block are coded in the bitstream.

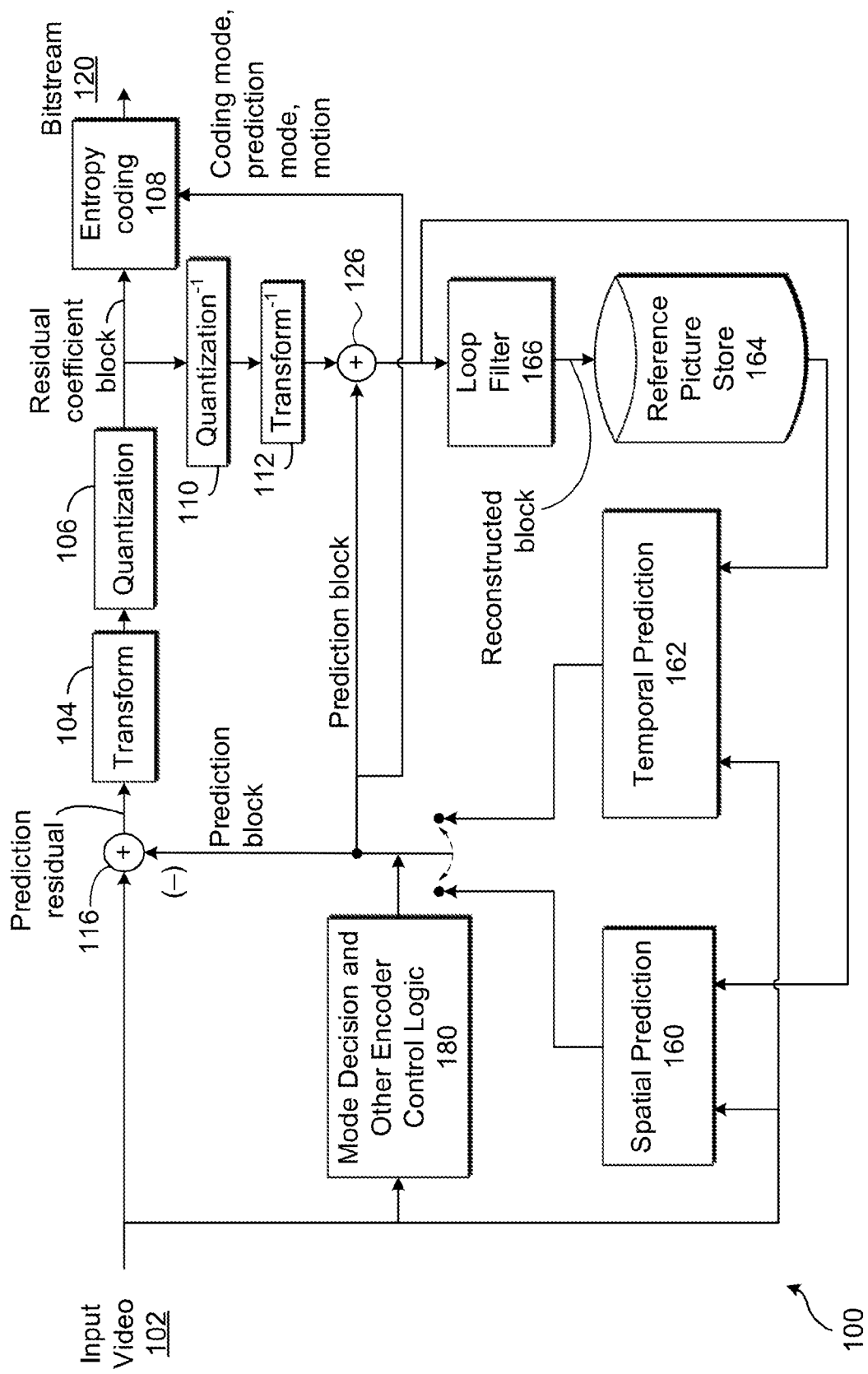


FIG. 1

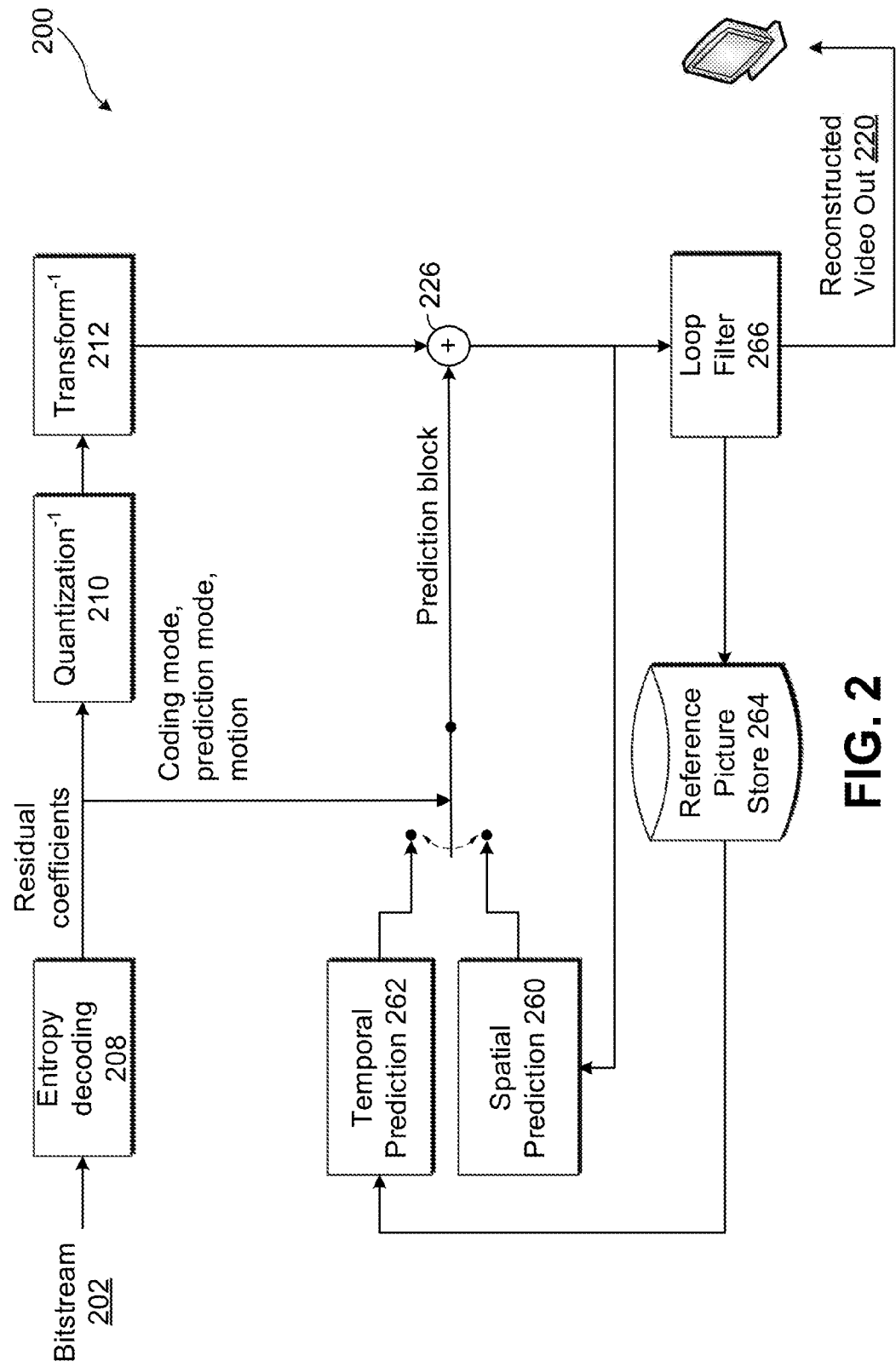


FIG. 2

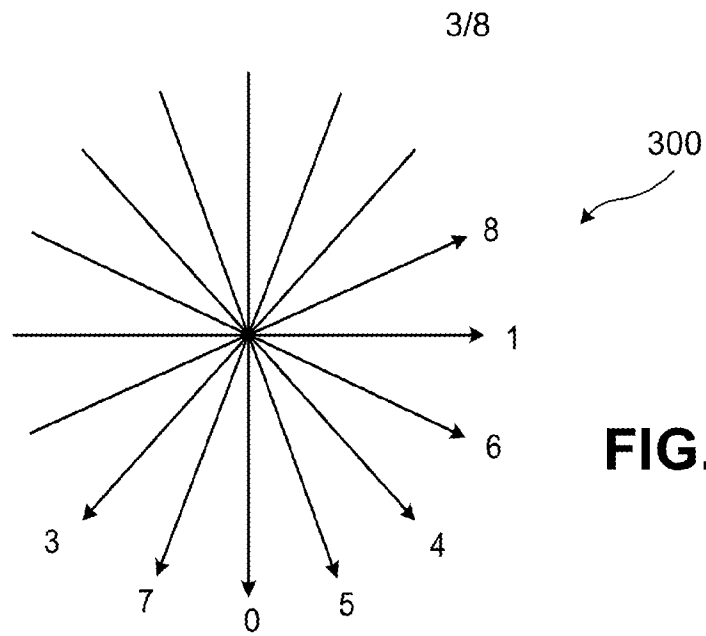


FIG. 3

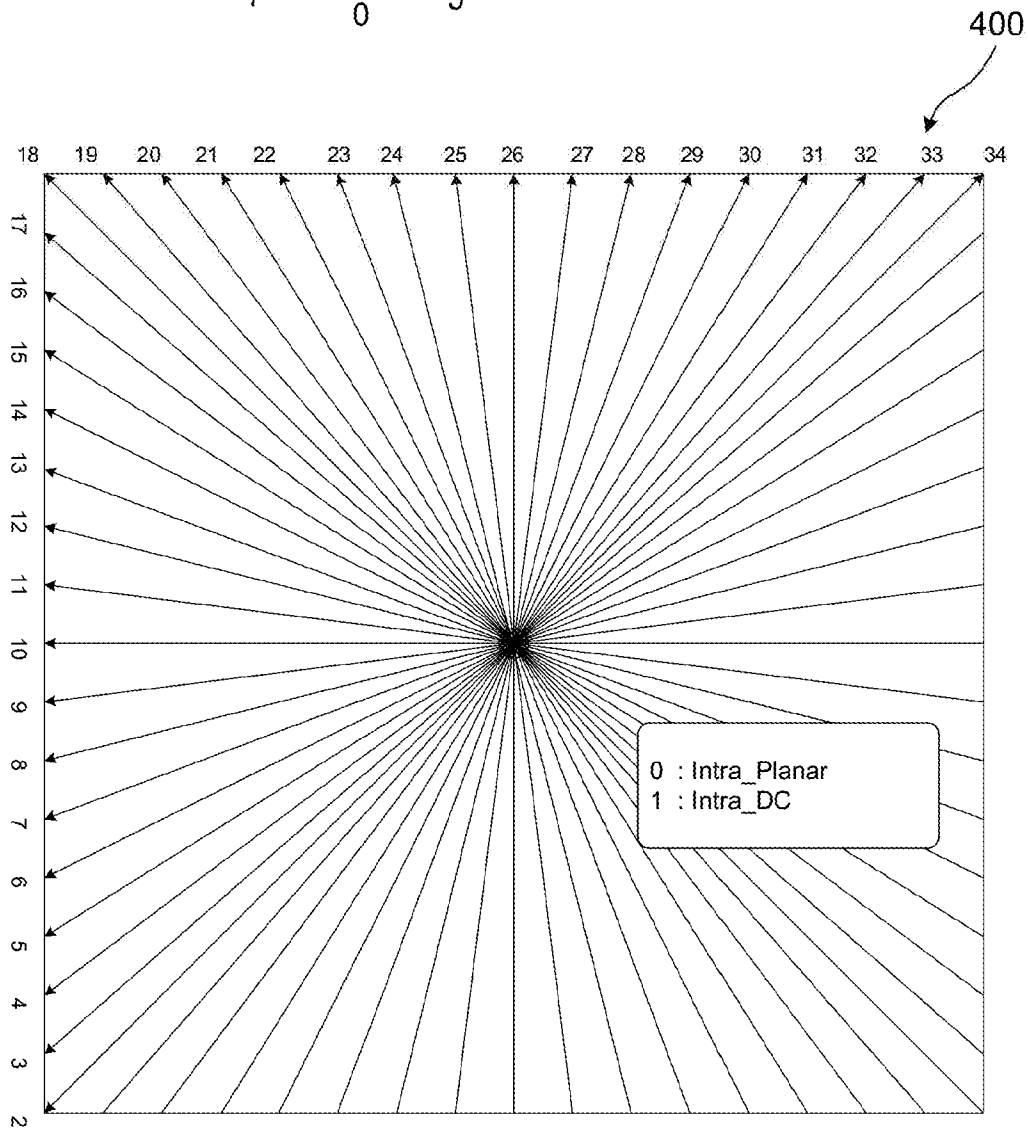
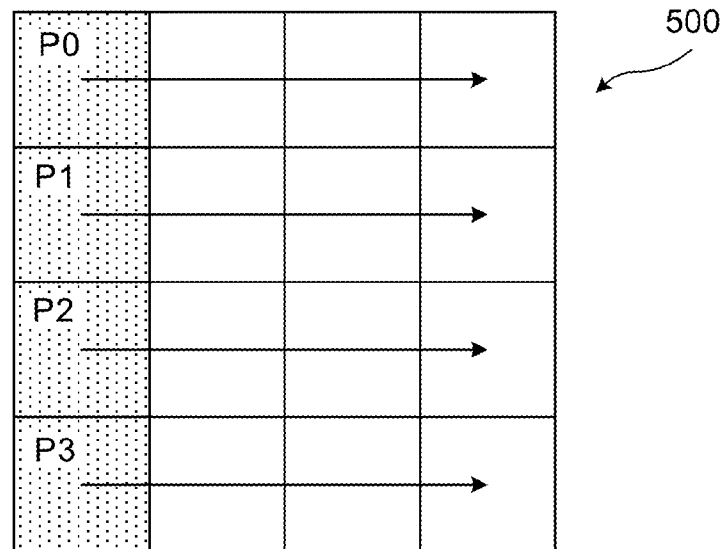
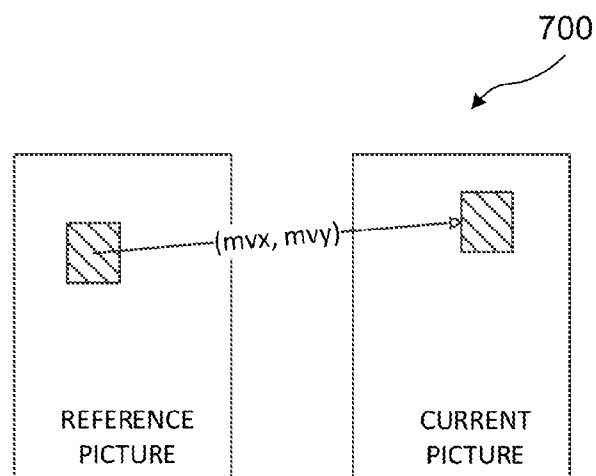
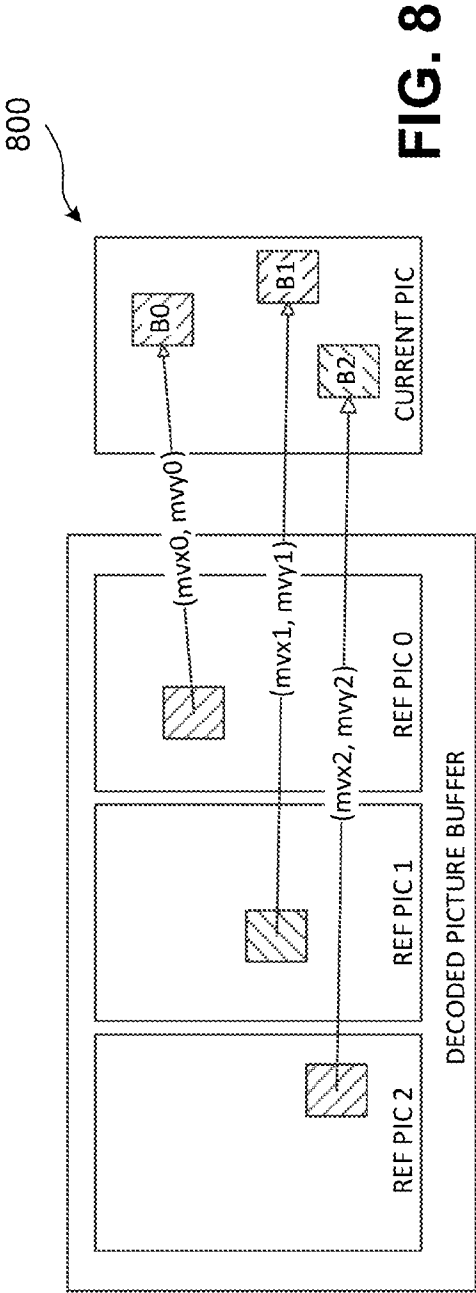
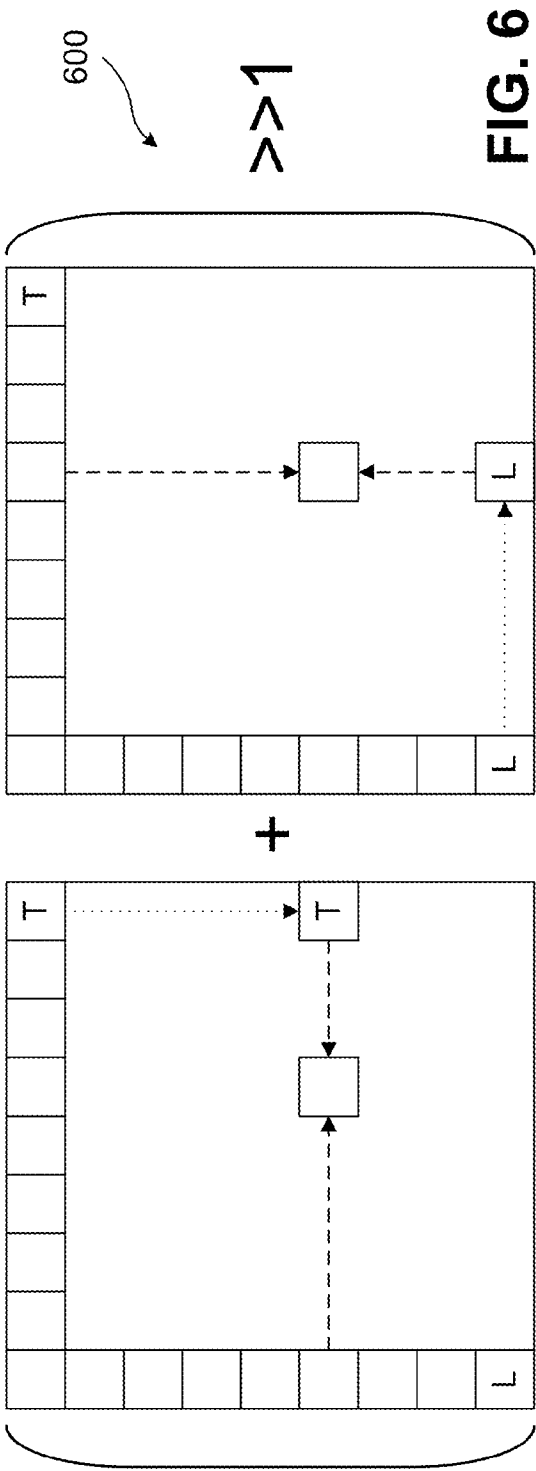


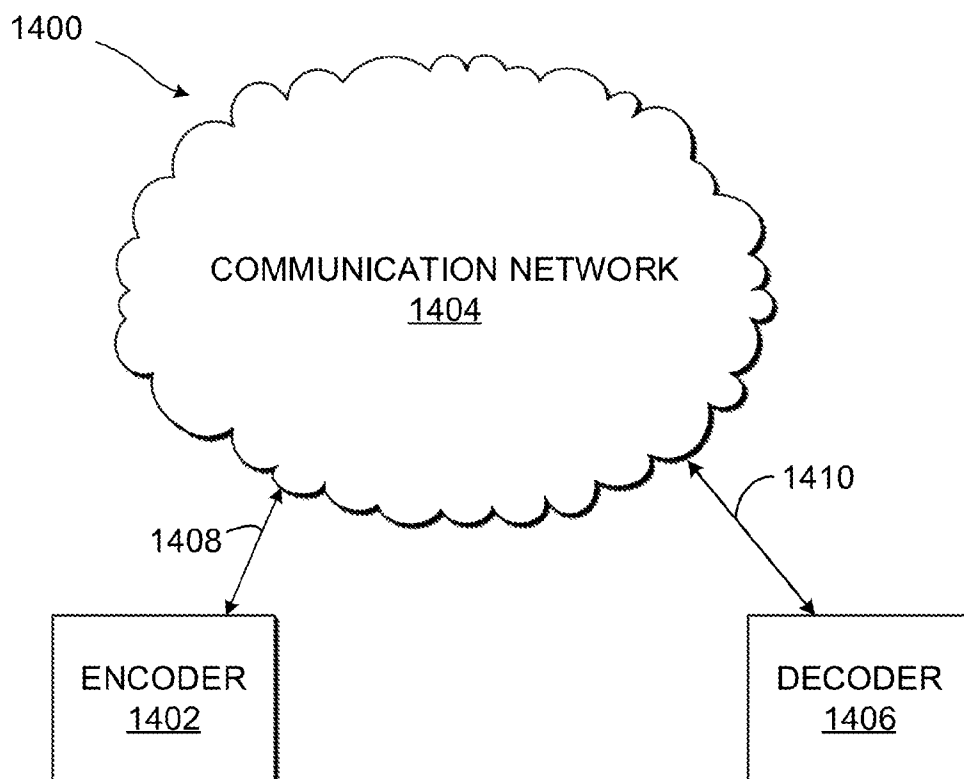
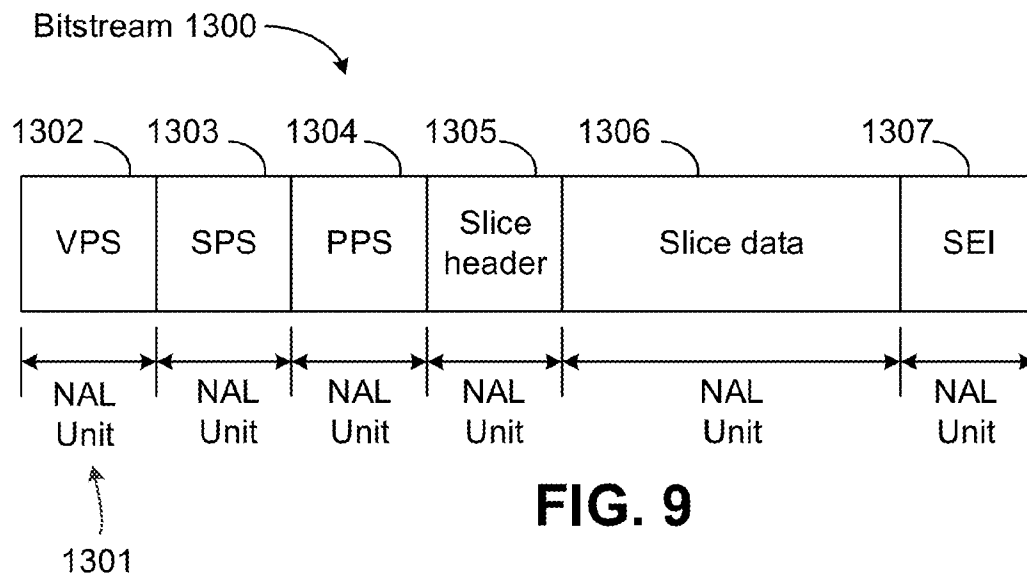
FIG. 4

4/8

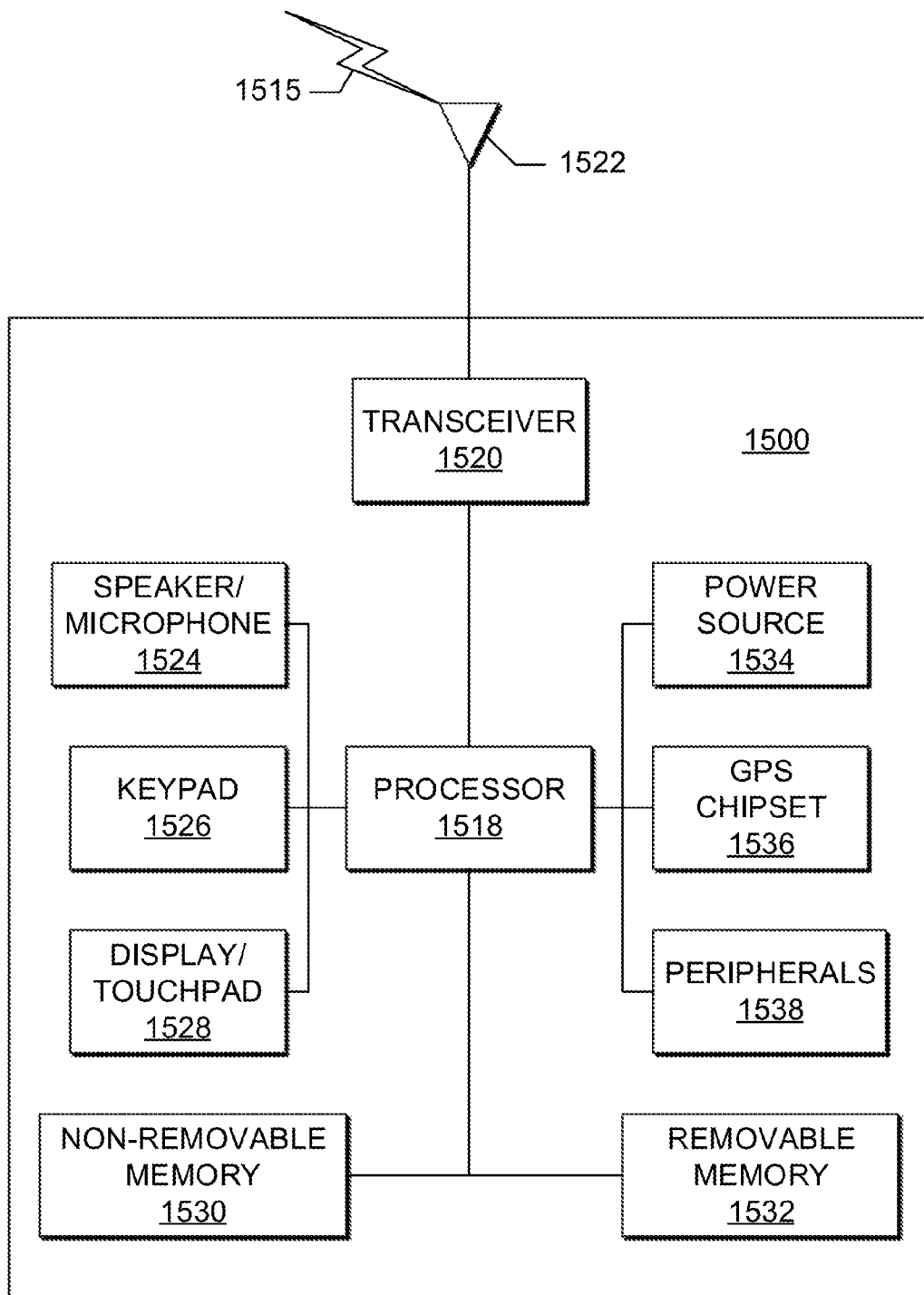
**FIG. 5****FIG. 7**



6/8



7/8

**FIG. 11**

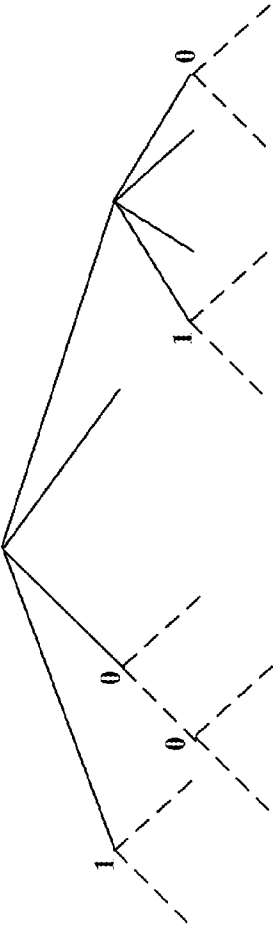
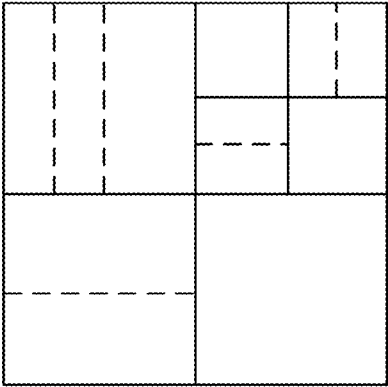


FIG. 12

- LEAF NODE WITH NONZERO TRANSFORM COEFFICIENTS
- LEAF NODE WITH NO TRANSFORM COEFFICIENTS

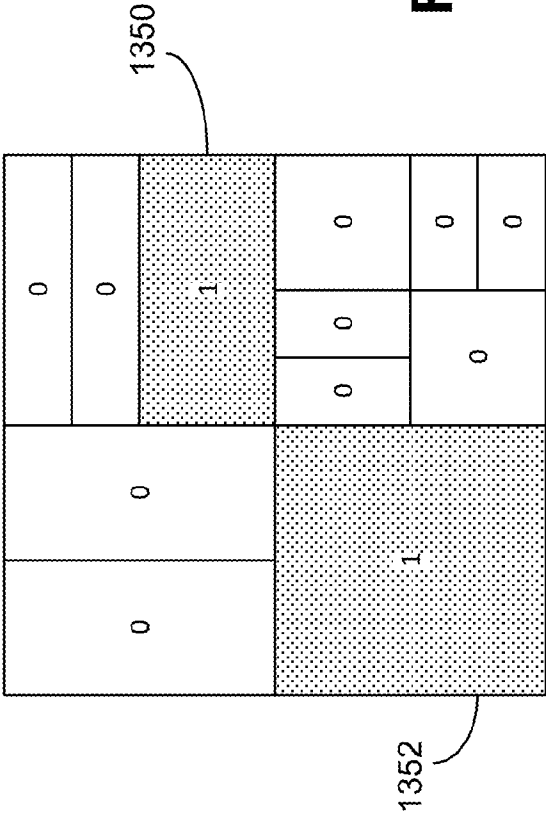


FIG. 13

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2017/049937

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N19/70 H04N19/96
ADD.

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

B. FIELDS SEARCHED

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

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

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

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2014/092965 A1 (JOSHI RAJAN LAXMAN [US] ET AL) 3 April 2014 (2014-04-03) paragraphs [0074], [0083], [0110], [0143] - [0147], [0149], [0151], [0153] - [0154], [0160]; figures 3,7 -----	1-18
Y	JICHENG AN ET AL: "Block partitioning structure for next generation video coding", 113. MPEG MEETING; 19-10-2015 - 23-10-2015; GENEVA; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. m37524, 26 October 2015 (2015-10-26), XP030065891, page 1 - page 3 ----- -/-	1-6, 10-14,18



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

13 October 2017

Date of mailing of the international search report

23/10/2017

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Seeger, Alexander

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2017/049937

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2013/251026 A1 (GUO LIWEI [US] ET AL) 26 September 2013 (2013-09-26) paragraphs [0027] - [0031], [0069] - [0074] -----	1-6, 10-14, 18

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2017/049937

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2014092965	A1	03-04-2014	
		CN 104685875 A	03-06-2015
		CN 104685876 A	03-06-2015
		DK 2904790 T3	16-01-2017
		EP 2904788 A1	12-08-2015
		EP 2904790 A1	12-08-2015
		ES 2611349 T3	08-05-2017
		HU E030994 T2	28-06-2017
		JP 2015534779 A	03-12-2015
		JP 2015534780 A	03-12-2015
		KR 20150064165 A	10-06-2015
		KR 20150067254 A	17-06-2015
		MA 37964 A1	31-03-2016
		PL 2904790 T3	31-05-2017
		PT 2904790 T	27-01-2017
		US 2014092965 A1	03-04-2014
		US 2014092983 A1	03-04-2014
		WO 2014055344 A1	10-04-2014
		WO 2014055345 A1	10-04-2014

US 2013251026	A1	26-09-2013	
		CN 104205837 A	10-12-2014
		EP 2829063 A2	28-01-2015
		JP 6046235 B2	14-12-2016
		JP 2015511105 A	13-04-2015
		US 2013251026 A1	26-09-2013
		WO 2013142784 A2	26-09-2013
