



(51) International Patent Classification:

H04N 19/513 (2014.01) H04N 19/503 (2014.01)

H04N 19/593 (2014.01)

(21) International Application Number:

PCT/CN2019/086654

(22) International Filing Date:

13 May 2019 (13.05.2019)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicant: ALIBABA GROUP HOLDING LIMITED

[—/CN]; Fourth Floor, One Capital Place, P.O. Box 847, George Town, Grand Cayman, Cayman Islands (KY).

(72) Inventors: AN, Jicheng; Wangjing Block A, Alibaba Center No. 9th Wangjing East Garden 4th Area, Chaoyang District, Beijing, China, Beijing 100102 (CN). LOU, Jian;

10270 NE 12th St F305, Bellevue, WA 98004 (US).

(74) Agent: BEIJING TSINGYUANHUI INTELLECTUAL PROPERTY LAW FIRM; Room 362, 3rd Floor, Suzhou Street No.1, Haidian District, Beijing 100080 (CN).

(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: NON-LOCAL INTRA BLOCK COPY AND PARALLEL CODING FOR MOTION PREDICTION

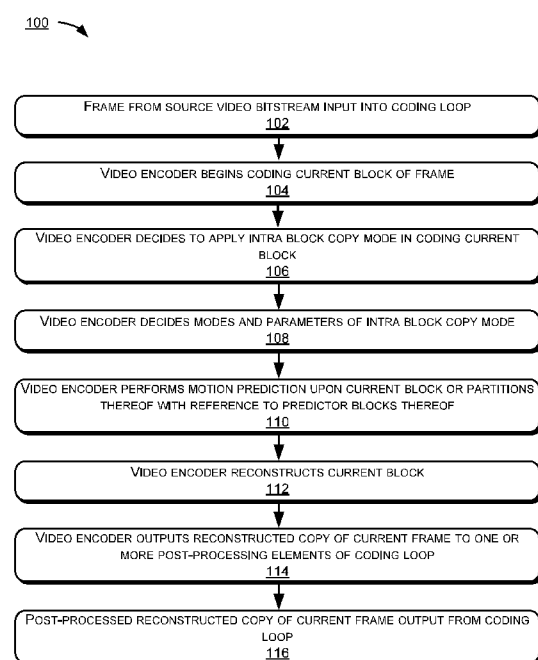


FIG. 1

(57) Abstract: Systems and methods are provided for implementing non-local intra block copy and parallel coding for motion prediction by steps including, while coding a current block of a frame, a video encoder performing motion prediction upon at least a partition of the current block with reference to a non-local predictor block in the frame; and the video encoder reconstructing the current block, thereby enabling predictor block searches for different partitions of the current block to be executed in parallel, reducing processing overhead during video encoding and increasing the speed and responsiveness of video content delivery.



NON-LOCAL INTRA BLOCK COPY AND PARALLEL CODING FOR MOTION PREDICTION

BACKGROUND

[0001] The development of the High-Efficiency Video Coding (HEVC) standard as a successor video coding format to the H.264/AVC (Advanced Video Coding) standard, has, as an objective, implementing new coding features to reduce bandwidth of transporting a video signal. Among various proposed features of the HEVC standard, additional extended features have also been proposed, referred to as the HEVC Extensions on Screen Content Coding (HEVC SCC).

[0002] One aspect of HEVC SCC is intra block copy (IBC), a motion prediction method which operates within inter frame prediction provided by HEVC. In inter frame prediction, to predict motion information of a current frame being encoded, another frame from the source video is selected as a reference, and motion information of a prediction unit (PU) of the current frame is derived from motion information of the reference. IBC utilizes the current frame itself as the reference frame, wherein another block of the current frame is selected as a reference block for predicting motion information of a current block. IBC may be applied in frames whose pictorial content includes a pattern or texture that repeats over the frame.

[0003] However, in accordance with the IBC proposal, the reference block is “local” to the current block; that is, the reference block is in proximity to a same coding tree unit (CTU) to which the current block belongs, in such a way that the reference block includes at least some pixels of the CTU to which the current block belongs (such as at least some pixels of another block of the same CTU). The reference block overlapping with the CTU imposes limitations upon the encoding of frames according to IBC, which may lead to increased overhead and slower processing in encoding videos due to the creation of dependencies on encoding and reconstruction of pixels occupied by the reference block, and new techniques are required to overcome these limitations.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] The detailed description is set forth with reference to the accompanying figures. In the figures, the left-most digit(s) of a reference number identifies the figure in which the reference number first appears. The use of the same reference numbers in different figures indicates similar or identical items or features.

[0005] FIG. 1 illustrates a flowchart of a method of applying IBC to an inter motion prediction process.

[0006] FIG. 2 illustrates a result of a video encoder searching for a corresponding predictor block for a partition of a partition mode according to an existing implementation of IBC with unrestricted predictor block search.

[0007] FIG. 3 illustrates a result of a video encoder searching for a predictor block for a partition of a partition mode according to an implementation of IBC with non-local predictor block search.

[0008] FIG. 4 illustrates deciding an optimal partition mode for coding the current block under an existing implementation of IBC with unrestricted predictor block search and reconstructing the current block.

[0009] FIG. 5 illustrates a flowchart of deciding an optimal partition mode for reconstructing the current block under an implementation of IBC with non-local predictor block search, and reconstructing the current block.

[0010] FIG. 6 illustrates an example block diagram of a video coding process according to an example embodiment of the present disclosure.

[0011] FIG. 7 illustrates an example system for implementing the processes and methods described above for implementing non-local intra block copy and parallel coding for motion prediction.

DETAILED DESCRIPTION

[0012] Systems and methods discussed herein are directed to implementing non-local intra block copy and parallel coding for motion prediction, and more specifically to an intra block copy scheme for various video encoding standards wherein a reference block is not local to a current block, and parallel coding of blocks in a block is enabled.

[0013] According to example embodiments of the present disclosure, intra block copy may refer to a process implemented during motion prediction by a video encoder which processes a block of a frame from a bitstream and outputs a coded block having motion prediction information which may be reconstructed by a video encoder.

[0014] According to example embodiments of the present disclosure implemented to be compatible with the H.264 standard, a frame may be subdivided into macroblocks (MBs) each having dimensions of 16x16 pixels, which may be further subdivided into partitions. According to example embodiments of the present disclosure implemented to be compatible with the HEVC standard, a frame may be subdivided into coding tree units (CTUs), which may be further subdivided into coding units (CUs). According to example embodiments of the present disclosure implemented as other standards, a frame may be subdivided into units of NxN pixels, which may then be further subdivided into subunits. Each of these largest subdivided units of a frame may generally be referred to as a “block” for the purpose of this disclosure.

[0015] According to example embodiments of the present disclosure, a block may be subdivided into partitions having dimensions in multiples of 4x4 pixels. For example, a partition of a block may have dimensions of 8x4 pixels, 4x8 pixels, 8x8 pixels, 16x8 pixels, or 8x16 pixels. Consequently, a block may be subdivided into two partitions, four partitions, eight partitions, and the like, without limitation, in accordance with dimensions of the partitions thereof. Intra block copy may be applied to the entirety of a block, may be applied to a partition of a block, or may be applied separately to each partition of a block, without limitation.

[0016] According to example embodiments of the present disclosure, a block may be partitioned into multiple possible modes of partitions, where some modes may be mutually exclusive. For example, a block may be partitioned into two 16x8 partitions. Alternatively, a block may be partitioned into two 8x16 partitions.

[0017] According to example embodiments of the present disclosure, coding and motion prediction are applied to a current block of a frame having at least some blocks already coded and/or reconstructed, wherein a reconstructed block of the frame is selected as a reference, and a displacement vector indicates a relative shift between the

reference block and the current block. Pixels of the current block are predicted by the already reconstructed pixels of the reference block and the vector.

[0018] According to example embodiments of the present disclosure, video encoding standards implementing inter prediction modes may refer to data formats wherein frames are encoded with motion vector information and prediction information of a frame by the inclusion of one or more references to motion information and prediction units (PUs) of one or more other frames. Motion information may refer to data describing motion of a block structure of a frame or a unit or subunit thereof, such as motion vectors and references to blocks of a current frame or of another frame. PUs may refer to a unit or multiple subunits corresponding to a block structure among multiple block structures of a frame, such as an MB or a CTU, wherein blocks are partitioned based on the frame data and are coded according to established video codecs. Motion information corresponding to a PU may describe motion prediction as encoded by any motion vector coding tool, including, but not limited to, those described herein.

[0019] According to example embodiments of the present disclosure, motion prediction information may include a block vector (BV) identifying a predictor block. A BV may be a displacement vector representing a displacement between a current block and a predictor block that is referenced for coding of the current block. Displacement may be measured in pixels in a horizontal direction and a vertical direction over a current frame. The displacement vector may represent a displacement between a pixel of the current block and a corresponding pixel of the predictor block at the same positions within the respective blocks. For example, the displacement vector may represent a displacement from a pixel at an upper-left corner of the current block to a pixel at an upper-left corner of the predictor block.

[0020] According to example embodiments of the present disclosure, a predictor block corresponding to a current block may be a block, or a predictor block corresponding to a partition of a current block may be a block of same dimensions as the partition.

[0021] A block may occupy a range of pixels over a horizontal direction and a vertical direction of a frame. The range of a block may be defined by a relative position

of a pixel of the block to the frame (such as a coordinate of a pixel at an upper-left corner of the block relative to the frame), a width of the block over a first direction, and a length of the block over a second direction. For example, given coordinates representing a first pixel at an upper-left corner of a block, a second pixel at an upper-right corner of the block may be found by adding a width of the block to a horizontal coordinate of the first pixel, a third pixel at a lower-left corner of the block may be found by adding a length of the block to a vertical coordinate of the first pixel, and a fourth pixel at a lower-right corner of the block may be found by adding a width of the block to a horizontal coordinate of the first pixel and adding a length of the block to a vertical coordinate of the first pixel. A block may occupy all pixels circumscribed by a geometric boundary running through the first pixel, the second pixel, the fourth pixel, and the third pixel. A block may be a whole block, any partition of any dimensions of a block, and any arbitrarily bounded block occupying multiple blocks, multiple partitions of a block, and/or partitions of multiple blocks.

[0022] FIG. 1 illustrates a flowchart of a method 100 of applying IBC to a motion prediction process.

[0023] At step 102, a frame from a source video bitstream is input into a coding loop. A video encoder, filters, and other elements of a coding loop established by video encoding standards may be part of the coding loop. These elements may include a set of computer-readable program instructions executable by one or more processors to implement a video coding loop according to a video encoding standard supporting motion prediction, such as H.264 or HEVC.

[0024] At step 104, a video encoder begins coding a current block of the frame. Coding the current block may take place before, concurrent with, or after coding and/or reconstructing any number of other blocks of the frame, without limitation as to which other blocks of the frame have been, are being, or have yet to be coded or reconstructed, except that for the purpose of performing the subsequent step of searching for predictor blocks, at least one other block of the frame has already been reconstructed prior to step 104.

[0025] Coding of the current block and reconstruction of partitions thereof may proceed according to techniques established for video encoders such as according to the H.264 standard, the HEVC standard, or other implementations. For example, the current block or partitions thereof may undergo transformation and quantization prior to being entropy coded.

[0026] At step 106, the video encoder decides to apply an intra block copy mode in coding the current block. Although other prediction modes may be available to the video encoder in deciding a prediction mode in coding the block, and the video encoder, as of the time of step 106, may have coded, may be coding, or may subsequently code other blocks of the frame in either intra block copy mode or other prediction modes, herein it is assumed that an intra block copy mode is applied in coding the current block.

[0027] At step 108, the video encoder decides modes and parameters of the intra block copy mode. For example, the video encoder may decide an optimal partition mode for coding the current block. An optimal partition mode may be a partition mode yielding a minimal value for a suitable cost function among partition modes of a current block. Suitable cost functions may include, for example, Sum of Absolute Differences (SAD). To compute a cost function for a partition mode, for each partition in a partition mode, the video encoder may search for a predictor block of same dimensions as the partition in the current frame.

[0028] For example, a first partition mode for the current block includes a first 16x8 partition and a second 16x8 partition; for the first 16x8 partition a first 16x8 predictor block may be searched for in the current frame, and for the second 16x8 partition a second 16x8 predictor block may be searched for in the current frame. A second partition mode for the current block includes a first 8x16 partition and a second 8x16 partition; for the first 8x16 partition a first 8x16 predictor block may be searched for in the current frame, and for the second 8x16 partition a second 8x16 predictor block may be searched for in the current frame. If the first partition mode yields a lower value for a cost function than the second partition mode does, the first partition mode is more optimal than the second partition mode. If the second partition mode yields a lower value for a cost function than the first partition mode does, the second partition mode is

more optimal than the first partition mode. At the same time, yet other partition modes for the same block may be more optimal or less optimal than the first partition mode and/or the second partition mode.

[0029] At step 110, the video encoder performs motion prediction upon the current block or partitions thereof with reference to predictor blocks thereof. A predictor block of the current block or a partition thereof may be a result of searching during the previous step as mentioned above. Motion prediction may be performed in accordance with a prediction mode as implemented according to a video encoding standard supporting motion prediction, such as H.264 or HEVC. Motion prediction may be performed in accordance with modes and parameters decided in step 108 as described above. For example, motion prediction may be performed in accordance with the optimal partition mode for coding the current block.

[0030] At step 112, the video encoder reconstructs the current block. The video encoder may reconstruct the current block as a whole or on a per-partition basis in accordance with a partition mode. The video encoder may reconstruct the current block as a whole in the case that no dependencies as described subsequently with regard to FIG. 4 prevent the current block from being reconstructed as a whole, in which case steps 110 and 112 may be performed after step 108. The video encoder may reconstruct the current block on a per-partition basis in the case that dependencies as described subsequently with regard to FIG. 4 prevent the current block from being reconstructed as a whole, in which case parts of steps 110 and 112 may be performed concurrently or interspersed with parts of step 108.

[0031] Reconstruction of the current block and reconstruction of partitions thereof may proceed according to techniques established for video encoders such as according to the H.264 standard, the HEVC standard, or other implementations. For example, the current block or partitions thereof may undergo dequantization and inverse transformation prior to being reconstructed.

[0032] At step 114, the video encoder outputs a reconstructed copy of the current frame to one or more post-processing elements of the coding loop. For example, while, for the purpose of performing motion prediction during the above step 110, another

reconstructed copy of the current frame which has not been output into a filter may be maintained to provide a sample source of reconstructed pixels for predictor blocks, the reconstructed copy of the current frame may instead be output into an in-loop filter for processing prior to outputting the reconstructed frame from the coding loop. Other post-processing elements may perform operations upon the reconstructed frame such as scaling, deinterlacing, deblocking, and such post-processing techniques established in video encoding; the present disclosure shall not be limiting as to post-processing operations that may be performed herein.

[0033] At step 116, the post-processed reconstructed copy of the current frame is output from the coding loop. The post-processed reconstructed copy of the current frame may be, for example, output to a display buffer for display, or output to a reference frame buffer to be made available as a reference frame in motion prediction subsequent frames of the video source. A reconstructed copy of the current frame which has not been output into post-processing elements of the coding loop, in contrast, may not be output from the coding loop and may be discarded.

[0034] FIG. 2 illustrates a result of a video encoder searching for a corresponding predictor block for a partition of a partition mode according to an existing implementation of IBC with unrestricted predictor block search. As illustrated by FIG. 2, a current frame 200 contains a current block 202 as referenced in the above-mentioned step 104, a reconstructed region 204 containing one or more reconstructed blocks, and a non-reconstructed region 206 containing one or more to-be reconstructed blocks. The reconstructed region 204 includes one or more local reconstructed blocks 208, which are reconstructed blocks immediately above and/or left of the current block 202. An unrestricted predictor block search for a partition 210 of the current block 202 has resulted in a predictor block 212 corresponding thereto, which occupies at least some pixels of one or more local reconstructed blocks 208 and at least some pixels of the current block 202. A BV 214 represents a displacement between the partition 210 and the predictor block 212 corresponding thereto.

[0035] Criteria for unrestricted predictor block search may be in accordance with those described in existing intra block copy implementations and are not reiterated

herein, except to note that criteria for searching for a predictor block according to such proposals result in, or at least do not prevent, a predictor block 212 from occupying pixels of the current block 202.

[0036] FIG. 3 illustrates a result of a video encoder searching for a predictor block for a partition of a partition mode according to an implementation of IBC with non-local predictor block search. As illustrated by FIG. 3, a current frame 300 contains a current block 302 as referenced in the above-mentioned step 104, a reconstructed region 304 containing one or more reconstructed blocks, and a non-reconstructed region 306 containing one or more to-be reconstructed blocks. The reconstructed region 304 includes one or more local reconstructed blocks 308, which are reconstructed blocks immediately above and/or left of the current block 302. A non-local predictor block search for a partition 310 of the current block 302 has resulted in a predictor block 312 corresponding thereto, which occupies at least some pixels of one or more local reconstructed blocks 308 and at least some pixels of the current block 302. A BV 314 represents a displacement between the partition 310 and the predictor block 312 corresponding thereto.

[0037] Criteria for non-local predictor block search are in accordance with those described in existing intra block copy proposals and are not reiterated herein, with the additional restriction that a predictor block 312 corresponding to a partition 310 of the current block 302 other than a first-coded partition of the current block 302 shall not occupy pixels of the current block 302. This additional restriction may not be enforced for the first-coded partition of the current block 302. A first-coded partition of the current block 302 may be a partition among each partition of the current block 302 which is encoded earlier and decoded earlier than each other partition according to a suitable order, such as a raster scan order wherein a first-coded partition is an uppermost and leftmost partition of the current block 302, according to video encoding standards. For example, assuming that all blocks of the reconstructed region 304 are above and/or left in the frame relative to the current block 302, to prevent the predictor block 312 from occupying pixels of the current block 302, given coordinates representing a first pixel at an upper-left corner of the predictor block 312 and coordinates representing a

second pixel at an upper-left corner of the current block 302, (a) a horizontal coordinate derived from adding a width of the predictor block to a horizontal coordinate of the first pixel should be less than a horizontal coordinate of the second pixel, and (b) a vertical coordinate derived from adding a length of the predictor block to a vertical coordinate of the first pixel should be less than a vertical coordinate of the second pixel. If the search for a predictor block would occur at a location in the current frame where either or both of (a) and (b) would not be satisfied, the search may skip that location.

[0038] Thus, example embodiments of the present disclosure may both expand and limit the parameters of a search for a predictor block of a current block or a partition thereof. The parameters of the search may be expanded to include pixels of each block of the current frame already reconstructed at the time of coding the current block of the frame. At the same time, the parameters of the search may be limited to, while searching for a predictor block corresponding to any partition other than a first-coded partition, skip search locations where a found predictor block would occupy any pixels of the current block or pixels of any partitions thereof. Therefore, the search may be performed over all pixels of the current frame already reconstructed, except that search locations sufficiently proximate to the current block such that a found predictor block would overlap with the current block shall be skipped while searching for a predictor block corresponding to any partition other than a first-coded partition.

[0039] FIG. 4 illustrates a flowchart of deciding an optimal partition mode for coding the current block under an existing implementation of IBC with predictor blocks being local and reconstructing the current block.

[0040] To decide an optimal partition mode, respective values of a cost function may be computed for possible partition modes of the current block. As illustrated by FIG. 4, cost_0 represents a value of the SAD cost function for a first partition mode including a first 16x8 partition and a second 16x8 partition, and cost_1 represents a value of the SAD cost function for a second partition mode including a first 8x16 partition and a second 8x16 partition.

[0041] According to example embodiments of the present disclosure, the value of cost_0 may be computed from at least a first difference derived from comparing the

first 16x8 partition with a predictor block corresponding thereto and a second difference derived from comparing the second 16x8 partition with a predictor block corresponding thereto. According to example embodiments of the present disclosure, the value of cost₁ may be computed from at least a first difference derived from comparing the first 8x16 partition with a predictor block corresponding thereto and a second difference derived from comparing the second 8x16 partition with a predictor block corresponding thereto. An optimal partition mode may be decided as being the partition mode of the current block having the lowest value of the cost function among all partition modes of the current block.

[0042] As illustrated by FIG. 4, and further in reference to FIG. 2, suppose that the current block is the current block 202 as illustrated, the second 16x8 partition is the partition 210 as illustrated, the first 16x8 partition occupies the remainder of the current block 202, and the predictor block corresponding to the second 16x8 partition is the predictor block 212 as illustrated. Then, searching for a predictor block corresponding to the second 16x8 partition may be dependent on first reconstructing the 16x8 partition, because, in accordance with predictor blocks being local, the pixels occupied by the first 16x8 partition cannot be searched until they have first been reconstructed. Furthermore, when computing the value of cost₀, computing the value of the cost function for the second 16x8 partition may be dependent on first reconstructing the first 16x8 partition, because, in accordance with predictor blocks being local, the predictor block corresponding to the second 16x8 partition occupies at least some pixels of the first 16x8 partition, which must first be reconstructed before the second 16x8 partition may be compared with the predictor block corresponding thereto.

[0043] Likewise, suppose that the second 8x16 partition is an 8x16 partition occupying a right half of the current block 202, the first 8x16 partition occupies a left half of the current block 202, and an 8x16 predictor block corresponding to the second 8x16 partition partially overlaps with the left half of the current block 202. Then, searching for a predictor block corresponding to the second 8x16 partition may be dependent on first reconstructing the 8x16 partition, because, in accordance with predictor blocks being local, the pixels occupied by the first 8x16 partition cannot be

searched until they have first been reconstructed. Furthermore, when computing the value of cost_1 , computing the value of the cost function for the second 8x16 partition may be dependent on first reconstructing the first 8x16 partition, because, in accordance with predictor blocks being local, the predictor block corresponding to the second 8x16 partition occupies at least some pixels of the first 8x16 partition, which must first be reconstructed before the second 8x16 partition may be compared with the predictor block corresponding thereto.

[0044] Therefore, as illustrated by FIG. 4, the following steps may be performed during a process of deciding an optimal partition mode for coding the current block:

[0045] A method 400 of a video encoder computing a value of a cost function for a first partition mode include at least steps 402, 404, 406, 408, and 410.

[0046] In step 402, the video encoder searches for a predictor block corresponding to a first 16x8 partition.

[0047] In step 404, based on the predictor block corresponding thereto, the video encoder codes the first 16x8 partition in accordance with steps 104 to 110 as mentioned above, and the video encoder reconstructs the first 16x8 partition in accordance with step 112 as mentioned above.

[0048] In step 406, the video encoder searches for a predictor block corresponding to a second 16x8 partition. Step 406 has dependency upon step 404 because, in accordance with predictor blocks being local, the pixels occupied by the first 16x8 partition cannot be searched until they have first been reconstructed. Therefore, step 406 must be performed after step 404.

[0049] In step 408, based on the predictor block corresponding thereto, the video encoder codes the second 16x8 partition in accordance with steps 104 to 110 as mentioned above, and the video encoder reconstructs the second 16x8 partition in accordance with step 112 as mentioned above.

[0050] In step 410, the video encoder computes a value cost_0 of a cost function for the first partition mode.

[0051] A method 420 of a video encoder computing a value of a cost function for a second partition mode include at least steps 422, 424, 426, 428, and 430.

[0052] In step 422, the video encoder searches for a predictor block corresponding to a first 8x16 partition.

[0053] In step 424, based on the predictor block corresponding thereto, the video encoder codes the first 8x16 partition in accordance with steps 104 to 110 as mentioned above, and the video encoder reconstructs the first 8x16 partition in accordance with step 112 as mentioned above.

[0054] In step 426, the video encoder searches for a predictor block corresponding to a second 8x16 partition. Step 426 has dependency upon step 424 because, in accordance with predictor blocks being local, the pixels occupied by the first 8x16 partition cannot be searched until they have first been reconstructed. Therefore, step 426 must be performed after step 424.

[0055] In step 428, based on the predictor block corresponding thereto, the video encoder codes the second 8x16 partition in accordance with steps 104 to 110 as mentioned above, and the video encoder reconstructs the second 8x16 partition in accordance with step 112 as mentioned above.

[0056] In step 430, the video encoder computes a value cost₁ of a cost function for the second partition mode.

[0057] Furthermore, additional steps similar to those above may be performed wherein the video encoder computes values of the same cost function for any number of other partition modes for the current block.

[0058] Following the completion of steps 410 and 430 and any additional steps similar thereto, in step 440, the video encoder decides an optimal partition mode for coding the current block by finding the lowest value among all computed values of the cost function. Computed values of the cost function may include those computed in step 410, step 430, and additional steps similar thereto.

[0059] The optimal partition mode for coding the current block may be considered to perform motion prediction upon the current block as described in step 110 above.

[0060] FIG. 5 illustrates a flowchart of deciding an optimal partition mode for reconstructing the current block under an implementation of IBC with non-local predictor block search, and reconstructing the current block.

[0061] As illustrated by FIG. 5, and further in reference to FIG. 3, suppose that the current block is the current block 302 as illustrated, the second 16x8 partition is the partition 310 as illustrated, the first 16x8 partition occupies the remainder of the current block 302, and the predictor block corresponding to the second 16x8 partition is the predictor block 312 as illustrated. The first 16x8 partition may be a first-coded partition of the current block 302. Then, searching for a predictor block corresponding to the second 16x8 partition may be independent from first reconstructing the 16x8 partition, because, in accordance with non-local predictor block search, although the pixels occupied by the first 16x8 partition cannot be searched until they have first been reconstructed, search locations sufficiently proximate to the current block such that a found predictor block would overlap with the current block shall be skipped. Furthermore, when computing the value of cost₀, computing the value of the cost function for the second 16x8 partition may be independent from first reconstructing the first 16x8 partition, because, in accordance with non-local predictor block search, the predictor block corresponding to the second 16x8 partition does not occupy pixels of the first 16x8 partition, so the second 16x8 partition may be compared with the predictor block corresponding thereto regardless of whether the first 16x8 partition has been reconstructed.

[0062] Likewise, suppose that the second 8x16 partition is an 8x16 partition occupying a right half of the current block 302, the first 8x16 partition occupies a left half of the current block 302, and an 8x16 predictor block corresponding to the second 8x16 partition partially overlaps with the left half of the current block 302. The first 8x16 partition may be a first-coded partition of the current block 302. Then, searching for a predictor block corresponding to the second 8x16 partition may be independent from first reconstructing the 8x16 partition, because, in accordance with non-local predictor block search, although the pixels occupied by the first 8x16 partition cannot be searched until they have first been reconstructed, search locations sufficiently proximate to the current block such that a found predictor block would overlap with the current block shall be skipped. Furthermore, when computing the value of cost₁, computing the value of the cost function for the second 8x16 partition may be

independent from first reconstructing the first 8x16 partition, because, in accordance with non-local predictor block search, the predictor block corresponding to the second 8x16 partition does not occupy pixels of the first 8x16 partition, so the second 8x16 partition may be compared with the predictor block corresponding thereto regardless of whether the first 8x16 partition has been reconstructed.

[0063] Therefore, as illustrated by FIG. 5, the following steps may be performed during a process of deciding an optimal partition mode for coding the current block:

[0064] A method 500 of a video encoder computing a value of a cost function for a first partition mode include at least steps 502, 504, and 506.

[0065] In step 502, the video encoder searches for a predictor block corresponding to a first 16x8 partition.

[0066] In step 504, the video encoder searches for a predictor block corresponding to a second 16x8 partition. Step 504 has no dependency upon step 502 because, in accordance with non-local predictor block search, although the pixels occupied by the first 16x8 partition cannot be searched until they have first been reconstructed, search locations sufficiently proximate to the current block such that a found predictor block would overlap with the current block shall be skipped. Therefore, steps 502 and 504 may be performed with any timing relative to each other, including step 502 starting first, step 504 starting first, step 502 completing first, and step 504 completing first, and steps 502 and 504 may be performed wholly or partially concurrently by a processor capable of parallel execution of these steps.

[0067] In step 506, the video encoder computes a value `cost_0` of a cost function for the first partition mode. Step 506 may be performed after steps 502 and 504 have both completed.

[0068] A method 520 of a video encoder computing a value of a cost function for a first partition mode include at least steps 522, 524, and 526.

[0069] In step 522, the video encoder searches for a predictor block corresponding to a first 8x16 partition.

[0070] In step 524, the video encoder searches for a predictor block corresponding to a second 8x16 partition. Step 524 has no dependency upon step 522 because, in

accordance with non-local predictor block search, although the pixels occupied by the first 8x16 partition cannot be searched until they have first been reconstructed, search locations sufficiently proximate to the current block such that a found predictor block would overlap with the current block shall be skipped. Therefore, steps 522 and 524 may be performed with any timing relative to each other, including step 522 starting first, step 524 starting first, step 522 completing first, and step 524 completing first, and steps 522 and 524 may be performed wholly or partially concurrently by a processor capable of parallel execution of these steps.

[0071] In step 526, the video encoder computes a value $cost\_1$ of a cost function for the first partition mode. Step 526 may be performed after steps 522 and 524 have both completed.

[0072] Furthermore, additional steps similar to those above may be performed wherein the video encoder computes values of the same cost function for any number of other partition modes for the current block.

[0073] Following the completion of steps 506 and 526 and any additional steps similar thereto, in step 530, the video encoder decides an optimal partition mode for coding the current block by finding the lowest value among all computed values of the cost function. Computed values of the cost function may include those computed in step 506, step 526, and additional steps similar thereto.

[0074] The optimal partition mode for coding the current block may be considered to perform motion prediction upon the current block as described in step 110 above.

[0075] In step 540, based on the predictor block(s) corresponding to the partitions of the optimal partition mode, the video encoder codes the current block in accordance with steps 104 to 110 as mentioned above, and the video encoder reconstructs the current block in accordance with step 112 as mentioned above.

[0076] According to example embodiments of the present disclosure, non-local predictor block search causes the dependency of predictor block searches for a partition of the current block to be decoupled from reconstruction of other partitions of the current block, and therefore decoupled from predictor block searches for any other partition of the current block. This enables predictor block searches for different

partitions of the current block to be executed in parallel, reducing processing overhead during video encoding and increasing the speed and responsiveness of video content delivery.

[0077] According to example embodiments of the present disclosure, decoupling the dependency of predictor block searches for a partition of the current block from reconstructing the current block also decouples the dependency of computing values of a cost function from reconstructing the current block, and decouples deciding an optimal partition mode from reconstructing the current block. Furthermore, since pixels of a block are coded before they are reconstructed, this also decouples the dependency of predictor block searches for a partition of the current block, computing values of a cost function, and deciding an optimal partition mode from coding the current block. Thus, this enables the video encoder to code any partitions of, or the entirety of, the current block. and the video encoder to reconstruct any partitions of, or the entirety of, the current block after all predictor block searches for each partition mode of the current block; after all cost function value computation for each partition mode of the current block; and after deciding an optimal partition mode for coding the current block. Thus, the pixels of the current block need not be redundantly reconstructed more than once under multiple partition modes, as illustrated by FIG. 4; the entirety of the current block need only be reconstructed once, as illustrated by FIG. 5.

[0078] FIG. 6 illustrates an example block diagram of a video coding process 600 according to an example embodiment of the present disclosure.

[0079] From a video source such as a bitstream 610, a current frame 612 may be obtained and input into a coding loop 614. In the coding loop 614, a video encoder 616 may code the frame on a per-block basis in accordance with steps 104 to 110 as mentioned above to generate residues and motion prediction information, and the video encoder 616 may reconstruct the residues and motion prediction information on a per-block basis in accordance with step 112 as mentioned above to generate a reconstructed frame 620.

[0080] The video encoder 616 may output the reconstructed frame 620 to one or more post-processing elements 622 of the coding loop, which may perform operations

upon the reconstructed frame 620 such as scaling, deinterlacing, deblocking, and such post-processing techniques established in video encoding. After performance of post-processing operations, the reconstructed frame 620 may be output from the coding loop at a destination such as a reference frame buffer 624 or a display buffer 626.

[0081] The video encoder 616 may also store a copy of the reconstructed frame 630 which is not output to the post-processing elements of the coding loop. The copy of the reconstructed frame 630 may be maintained to provide a sample source of reconstructed pixels for predictor blocks. The copy of the reconstructed frame 630 may be discarded after the reconstructed frame 620 is output from the coding loop.

[0082] According to example embodiments of the present disclosure, the video encoder 616 may be any encoder implementing a coding format, including, but not limited to, those coding formats described herein. Generating a reconstructed frame based on motion prediction information may include intra block copy as described herein, and may include inter prediction based on a previous reconstructed frame serving as a reference picture as established in related technology, which shall not be reiterated herein.

[0083] FIG. 7 illustrates an example system 700 for implementing the processes and methods described above for implementing non-local intra block copy and parallel coding for motion prediction.

[0084] The techniques and mechanisms described herein may be implemented by multiple instances of the system 700 as well as by any other computing device, system, and/or environment. The system 700 shown in FIG. 7 is only one example of a system and is not intended to suggest any limitation as to the scope of use or functionality of any computing device utilized to perform the processes and/or procedures described above. Other well-known computing devices, systems, environments and/or configurations that may be suitable for use with the embodiments include, but are not limited to, personal computers, server computers, hand-held or laptop devices, multiprocessor systems, microprocessor-based systems, set top boxes, game consoles, programmable consumer electronics, network PCs, minicomputers, mainframe computers, distributed computing environments that include any of the above systems

or devices, implementations using field programmable gate arrays (“FPGAs”) and application specific integrated circuits (“ASICs”), and/or the like.

[085] The system 700 may include one or more processors 702 and system memory 704 communicatively coupled to the processor(s) 702. The processor(s) 702 may execute one or more modules and/or processes to cause the processor(s) 702 to perform a variety of functions. In some embodiments, the processor(s) 702 may include a central processing unit (CPU), a graphics processing unit (GPU), both CPU and GPU, or other processing units or components known in the art. Additionally, each of the processor(s) 702 may possess its own local memory, which also may store program modules, program data, and/or one or more operating systems.

[086] Depending on the exact configuration and type of the system 700, the system memory 704 may be volatile, such as RAM, non-volatile, such as ROM, flash memory, miniature hard drive, memory card, and the like, or some combination thereof. The system memory 704 may include one or more computer-executable modules 706 that are executable by the processor(s) 702.

[087] The modules 706 may include, but are not limited to, a frame inputting module 708, an encoder module 710, one or more post-processing modules 714, and a frame outputting module 716. The encoder module 710 may include a block coding submodule 718, a block prediction mode deciding submodule 720, a predictor block searching submodule 722, a cost function computing submodule 724, an optimal partition mode deciding submodule 726, and a block motion predicting submodule 728. The encoder module 710 may include a reconstructing submodule 730 and a reconstructed copy output submodule 732. The one or more post-processing modules 714 may include various modules not illustrated herein such as a scaling module, a deinterlacing module, a deblocking module, and other modules configured to perform such post-processing techniques established in video encoding; the present disclosure shall not be limiting as to post-processing modules that may be included herein.

[088] The frame inputting module 708 may be configured to input a frame from a source video bitstream into a coding loop.

[089] The block coding submodule 718 of the encoder module 710 may be configured to code a current block of the frame. The block coding submodule 718 coding the current block may take place before, concurrent with, or after the block coding submodule 718 coding and/or the reconstructing submodule 730 reconstructing any number of other blocks of the frame, without limitation as to which other blocks of the frame have been, are being, or have yet to be coded by the block coding submodule 718 or reconstructed by the reconstructing submodule 730, except that for the purpose of the block coding submodule 718 performing the subsequent step of searching for predictor blocks, at least one other block of the frame has already been reconstructed by the reconstructing submodule 730 prior to coding the current block.

[090] The block coding submodule 718 may code the current block and reconstruction of partitions thereof according to techniques established for video encoders such as according to the H.264 standard, the HEVC standard, or other implementations. For example, the current block or partitions thereof may undergo transformation and quantization prior to being entropy coded.

[091] The block prediction mode deciding submodule 720 may be configured to decide to apply an intra block copy mode in coding the current block. Although other prediction modes may be available to the block prediction mode deciding submodule 720 in deciding a prediction mode in coding the block, and the block coding submodule 718 may have coded, may be coding, or may subsequently code other blocks of the frame in either intra block copy mode or other prediction modes, herein it is assumed that an intra block copy mode is applied in coding the current block.

[092] The optimal partition mode deciding submodule 726 may be configured to decide an optimal partition mode for coding the current block. An optimal partition mode may be a partition mode yielding a minimal value for a suitable cost function among partition modes of a current block. Suitable cost functions may include, for example, Sum of Absolute Differences (SAD). The cost function computing submodule 724 may be configured to compute a cost function for a partition mode, wherein for each partition in a partition mode, a predictor block of same dimensions as the partition may be searched for in the current frame. The predictor block searching

submodule 722 may be configured to search for the predictor block corresponding to the partition. Details of the steps performed by the predictor block searching submodule 722, the cost function computing submodule 724, and the optimal partition mode deciding submodule 726 may be as described above with reference to FIG. 1.

[093] The block motion predicting submodule 728 may be configured to perform motion prediction upon the current block or partitions thereof with reference to predictor blocks thereof. A predictor block of the current block or a partition thereof may be a result of searching by the predictor block searching submodule as mentioned above. The block motion predicting submodule 728 may perform motion prediction in accordance with an intra block copy mode as implemented according to a video encoding standard supporting intra block copy, such as H.264 or HEVC. The block motion predicting submodule 728 may perform motion prediction in accordance with the optimal partition mode for coding the current block as decided by the optimal partition mode deciding submodule 726.

[0094] The reconstructing submodule 730 may be configured to reconstruct the current block. The reconstructing submodule 730 may reconstruct the current block as a whole or on a per-partition basis in accordance with a partition mode. The reconstructing submodule 730 may reconstruct the current block as a whole in the case that no dependencies as described above with regard to FIG. 4 prevent the current block from being reconstructed as a whole, in which case the reconstructing submodule 730 may reconstruct the current block after the optimal partition mode deciding submodule 726 decides an optimal partition mode. The reconstructing submodule 730 may reconstruct the current block on a per-partition basis in the case that dependencies as described subsequently with regard to FIG. 4 prevent the current block from being reconstructed as a whole, in which case the reconstructing submodule 730 may reconstruct parts of the current block concurrently or interspersed with the predictor block searching submodule 722 searching for predictor blocks, the cost function computing submodule 724 computing a cost function for a partition mode, and the optimal partition mode deciding submodule 726 deciding an optimal partition mode for coding the current block.

[095] The reconstructing submodule 730 may reconstruct the current block and reconstruct partitions thereof according to techniques established for video encoders such as according to the H.264 standard, the HEVC standard, or other implementations. For example, the current block or partitions thereof may undergo dequantization and inverse transformation prior to being reconstructed.

[096] The reconstructed copy output submodule 732 may be configured to output a reconstructed copy of the current frame to one or more post-processing elements of the coding loop. For example, while, for the purpose of the block motion prediction submodule 728 performing motion prediction as described above, another reconstructed copy of the current frame which has not been output into a filter may be maintained to provide a sample source of reconstructed pixels for predictor blocks, the reconstructed copy of the current frame may instead be output into an in-loop filter for processing prior to outputting the reconstructed frame from the coding loop. Other post-processing modules 714 may perform operations upon the reconstructed frame such as scaling, deinterlacing, deblocking, and such post-processing techniques established in video encoding; the present disclosure shall not be limiting as to post-processing operations that may be performed herein.

[097] The frame outputting module 716 may be configured to output the post-processed reconstructed copy of the current frame is output from the coding loop. The post-processed reconstructed copy of the current frame may be, for example, output to a display buffer for display, or output to a reference frame buffer to be made available as a reference frame in motion prediction subsequent frames of the video source. A reconstructed copy of the current frame which has not been output into post-processing elements of the coding loop, in contrast, may not be output from the coding loop and may be discarded.

[098] The system 700 may additionally include an input/output (I/O) interface 740 for receiving bitstream data to be processed, and for outputting reconstructed frames into a reference frame buffer and/or a display buffer. The system 700 may also include a communication module 750 allowing the system 700 to communicate with other devices (not shown) over a network (not shown). The network may include the Internet,

wired media such as a wired network or direct-wired connections, and wireless media such as acoustic, radio frequency (RF), infrared, and other wireless media.

[0099] Some or all operations of the methods described above can be performed by execution of computer-readable instructions stored on a computer-readable storage medium, as defined below. The term “computer-readable instructions” as used in the description and claims, include routines, applications, application modules, program modules, programs, components, data structures, algorithms, and the like. Computer-readable instructions can be implemented on various system configurations, including single-processor or multiprocessor systems, minicomputers, mainframe computers, personal computers, hand-held computing devices, microprocessor-based, programmable consumer electronics, combinations thereof, and the like.

[0100] The computer-readable storage media may include volatile memory (such as random-access memory (RAM)) and/or non-volatile memory (such as read-only memory (ROM), flash memory, etc.). The computer-readable storage media may also include additional removable storage and/or non-removable storage including, but not limited to, flash memory, magnetic storage, optical storage, and/or tape storage that may provide non-volatile storage of computer-readable instructions, data structures, program modules, and the like.

[0101] A non-transient computer-readable storage medium is an example of computer-readable media. Computer-readable media includes at least two types of computer-readable media, namely computer-readable storage media and communications media. Computer-readable storage media includes volatile and non-volatile, removable and non-removable media implemented in any process or technology for storage of information such as computer-readable instructions, data structures, program modules, or other data. Computer-readable storage media includes, but is not limited to, phase change memory (PRAM), static random-access memory (SRAM), dynamic random-access memory (DRAM), other types of random-access memory (RAM), read-only memory (ROM), electrically erasable programmable read-only memory (EEPROM), flash memory or other memory technology, compact disk read-only memory (CD-ROM), digital versatile disks (DVD) or other optical storage,

magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other non-transmission medium that can be used to store information for access by a computing device. In contrast, communication media may embody computer-readable instructions, data structures, program modules, or other data in a modulated data signal, such as a carrier wave, or other transmission mechanism. As defined herein, computer-readable storage media do not include communication media.

[0102] The computer-readable instructions stored on one or more non-transitory computer-readable storage media that, when executed by one or more processors, may perform operations described above with reference to FIGS. 1-7. Generally, computer-readable instructions include routines, programs, objects, components, data structures, and the like that perform particular functions or implement particular abstract data types. The order in which the operations are described is not intended to be construed as a limitation, and any number of the described operations can be combined in any order and/or in parallel to implement the processes.

[0103] By the abovementioned technical solutions, the present disclosure provides inter-coded resolution-adaptive video coding supported by motion prediction coding formats, improving the video coding process under multiple motion prediction coding formats by enabling resolution changes between frames to be coded while allowing motion vectors to reference previous frames. Thus, the bandwidth savings of inter-coding are maintained, the bandwidth savings of motion prediction coding are realized allowing reference frames to be used to predict motion vectors of subsequent frames, and the bandwidth savings of adaptively down-sampling and up-sampling according to bandwidth availability are also realized, all at the same time, achieving substantial improvement of network costs during video coding and content delivery while reducing the transport of additional data that would offset or compromise these savings.

EXAMPLE CLAUSES

[0104] A. A method comprising: while coding a current block of a frame, by a video encoder: performing motion prediction upon at least a partition of the current block other than a first-coded partition of the current block with reference to a predictor block

in the frame, the predictor block resulting from a non-local predictor block search; and reconstructing the current block by the video encoder.

[0105] B. The method as paragraph A recites, further comprising: deciding, by the video encoder, an optimal partition mode for coding the current block.

[0106] C. The method as paragraph B recites, wherein deciding, by the video encoder, an optimal partition mode for coding the current block comprises: performing predictor block search for a predictor block corresponding to a respective first-coded partition for each partition mode of a plurality of partition modes of the current block; performing non-local predictor block search for a predictor block corresponding to each respective partition other than a first-coded partition for each partition mode of a plurality of partition modes of the current block; computing for each partition mode, a value of a cost function based on each partition and the predictor block corresponding thereto; and deciding a partition mode of the current block having the lowest value of the cost function among all partition modes of the current block as an optimal partition mode.

[0107] D. The method as paragraph C recites, wherein searching for a predictor block corresponding to a partition is performed over all pixels of the current frame already reconstructed at the time of searching.

[0108] E. The method as paragraph C recites, wherein performing non-local predictor block search for predictor blocks corresponding to at least two partitions of at least one partition mode is executable at least partially in parallel.

[0109] F. The method as paragraph E recites, wherein reconstructing the current block by the video encoder comprises reconstructing the entirety of the current block after searching for a predictor block corresponding to each partition for each partition mode of a plurality of partition modes of the current block has been completed for each partition mode of the plurality of partition modes.

[0110] G. The method as paragraph B recites, further comprising performing, by the video encoder, motion prediction in accordance with the optimal partition mode for coding the current block.

[0111] H. The method as paragraph A recites, wherein reconstructing the current block is not performed while coding the current block; and reconstructing the current block does not comprise reconstructing any pixels of the current block more than once.

[0112] I. The method as paragraph A recites, wherein a block is one of: a macroblock (MB), a coding tree unit (CTU), or a block having width and length of a same arbitrary number of pixels.

[0113] J. A computer-readable storage medium storing computer-readable instructions executable by one or more processors, that when executed by the one or more processors, cause the one or more processors to perform operations comprising: while coding a current block of a frame, by a video encoder: performing motion prediction upon at least a partition of the current block other than a first-coded partition of the current block with reference to a predictor block in the frame, the predictor block resulting from a non-local predictor block search; and reconstructing the current block by the video encoder.

[0114] K. The computer-readable storage medium as paragraph J recites, wherein the operations further comprise: deciding, by the video encoder, an optimal partition mode for coding the current block.

[0115] L. The computer-readable storage medium as paragraph K recites, wherein deciding, by the video encoder, an optimal partition mode for coding the current block comprises: performing predictor block search for a predictor block corresponding to a respective first-coded partition for each partition mode of a plurality of partition modes of the current block; performing non-local predictor block search for a predictor block corresponding to each respective partition other than a first-coded partition for each partition mode of a plurality of partition modes of the current block; computing for each partition mode, a value of a cost function based on each partition and the predictor block corresponding thereto; and deciding a partition mode of the current block having the lowest value of the cost function among all partition modes of the current block as an optimal partition mode.

[0116] M. The computer-readable storage medium as paragraph L recites, wherein searching for a predictor block corresponding to a partition is performed over all pixels of the current frame already reconstructed at the time of searching.

[0117] N. The computer-readable storage medium as paragraph L recites, wherein performing non-local predictor block search for predictor blocks corresponding to at least two partitions of at least one partition mode is executable at least partially in parallel.

[0118] O. The computer-readable storage medium as paragraph K recites, wherein the operations further comprise performing, by the video encoder, motion prediction in accordance with the optimal partition mode for coding the current block.

[0119] P. The computer-readable storage medium as paragraph J recites, wherein reconstructing the current block is not performed while coding the current block; and reconstructing the current block does not comprise reconstructing any pixels of the current block more than once.

[0120] Q. The computer-readable storage medium as paragraph J recites, wherein a block is one of: a macroblock (MB), a coding tree unit (CTU), or a block having width and length of a same arbitrary number of pixels.

[0121] R. A system comprising: one or more processors; and memory communicatively coupled to the one or more processors, the memory storing computer-executable modules executable by the one or more processors that, when executed by the one or more processors, perform associated operations, the computer-executable modules comprising: an encoder module; a block coding submodule of the encoder module configured to code a current block of a frame; a block motion predicting submodule of the encoder module configured to perform motion prediction upon at least a partition of the current block other than a first-coded partition of the current block with reference to a predictor block in the frame, the predictor block resulting from a non-local predictor block search; and a reconstructing submodule of the encoder module configured to reconstruct the current block.

[0122] S. The system as paragraph R recites, further comprising: an optimal partition mode deciding submodule of the encoder module configured to decide an optimal partition mode for coding the current block.

[0123] T. The system as paragraph S recites, further comprising: a predictor block searching submodule configured to perform predictor block search for a predictor block corresponding to a respective first-coded partition for each partition mode of a plurality of partition modes of the current block; perform non-local predictor block search for a predictor block corresponding to each respective partition other than a first-coded partition for each partition mode of a plurality of partition modes of the current block; and a cost function computing submodule configured to compute, for each partition mode, a value of a cost function based on each partition and the predictor block corresponding thereto; wherein the optimal partition deciding submodule decides a partition mode of the current block having the lowest value of the cost function among all partition modes of the current block as an optimal partition mode.

[0124] U. The system as paragraph T recites, wherein the predictor block searching submodule searches for a predictor block corresponding to a partition over all pixels of the current frame already reconstructed at the time of searching.

[0125] V. The system as paragraph T recites, wherein the predictor block searching submodule performing non-local predictor block search for predictor blocks corresponding to at least two partitions of at least one partition mode is executable at least partially in parallel.

[0126] W. The system as paragraph V recites, wherein the reconstructing submodule reconstructs the entirety of the current block after the predictor block searching submodule searching for a predictor block corresponding to each partition for each partition mode of a plurality of partition modes of the current block has been completed for each partition mode of the plurality of partition modes.

[0127] X. The system as paragraph S recites, further comprising a block motion predicting submodule configured to perform motion prediction in accordance with the optimal partition mode for coding the current block.

[0128] Y. The system as paragraph R recites, wherein the reconstructing submodule does not reconstruct the current block while the block coding submodule coding the current block; and the reconstructing submodule does not, in reconstructing the current block, reconstruct any pixels of the current block more than once.

[0129] Z. The system as paragraph R recites, wherein a block is one of: a macroblock (MB), a coding tree unit (CTU), or a block having width and length of a same arbitrary number of pixels.

[0130] Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described. Rather, the specific features and acts are disclosed as exemplary forms of implementing the claims.

[0131]

CLAIMS

WHAT IS CLAIMED IS:

1. A method comprising:
while coding a current block of a frame, by a video encoder:
performing motion prediction upon at least a partition of the current block other than a first-coded partition of the current block with reference to a predictor block in the frame, the predictor block resulting from a non-local predictor block search; and
reconstructing the current block by the video encoder.
2. The method of claim 1, further comprising:
deciding, by the video encoder, an optimal partition mode for coding the current block.
3. The method of claim 2, wherein deciding, by the video encoder, an optimal partition mode for coding the current block comprises:
performing predictor block search for a predictor block corresponding to a respective first-coded partition for each partition mode of a plurality of partition modes of the current block;
performing non-local predictor block search for a predictor block corresponding to each respective partition other than a first-coded partition for each partition mode of a plurality of partition modes of the current block;
computing for each partition mode, a value of a cost function based on each partition and the predictor block corresponding thereto; and
deciding a partition mode of the current block having the lowest value of the cost function among all partition modes of the current block as an optimal partition mode.

4. The method of claim 3, wherein searching for a predictor block corresponding to a partition is performed over all pixels of the current frame already reconstructed at the time of searching.

5. The method of claim 3, wherein performing non-local predictor block search for predictor blocks corresponding to at least two partitions of at least one partition mode is executable at least partially in parallel.

6. The method of claim 5, wherein reconstructing the current block by the video encoder comprises reconstructing the entirety of the current block after searching for a predictor block corresponding to each partition for each partition mode of a plurality of partition modes of the current block has been completed for each partition mode of the plurality of partition modes.

7. The method of claim 2, further comprising performing, by the video encoder, motion prediction in accordance with the optimal partition mode for coding the current block.

8. The method of claim 1, wherein reconstructing the current block is not performed while coding the current block; and

reconstructing the current block does not comprise reconstructing any pixels of the current block more than once.

9. A computer-readable storage medium storing computer-readable instructions executable by one or more processors, that when executed by the one or more processors, cause the one or more processors to perform operations comprising:

while coding a current block of a frame, by a video encoder:

performing motion prediction upon at least a partition of the current block other than a first-coded partition of the current block with reference to a predictor

block in the frame, the predictor block resulting from a non-local predictor block search; and

reconstructing the current block by the video encoder.

10. The computer-readable storage medium of claim 9, wherein the operations further comprise:

deciding, by the video encoder, an optimal partition mode for coding the current block.

11. The computer-readable storage medium of claim 10, wherein deciding, by the video encoder, an optimal partition mode for coding the current block comprises:

performing predictor block search for a predictor block corresponding to a respective first-coded partition for each partition mode of a plurality of partition modes of the current block;

performing non-local predictor block search for a predictor block corresponding to each respective partition other than a first-coded partition for each partition mode of a plurality of partition modes of the current block;

computing for each partition mode, a value of a cost function based on each partition and the predictor block corresponding thereto; and

deciding a partition mode of the current block having the lowest value of the cost function among all partition modes of the current block as an optimal partition mode.

12. The computer-readable storage medium of claim 11, wherein searching for a predictor block corresponding to a partition is performed over all pixels of the current frame already reconstructed at the time of searching.

13. The computer-readable storage medium of claim 11, wherein performing non-local predictor block search for predictor blocks corresponding to at least two partitions of at least one partition mode is executable at least partially in parallel.

14. The computer-readable storage medium of claim 13, wherein reconstructing the current block by the video encoder comprises reconstructing the entirety of the current block after searching for a predictor block corresponding to each partition for each partition mode of a plurality of partition modes of the current block has been completed for each partition mode of the plurality of partition modes.

15. A system comprising:
one or more processors; and
memory communicatively coupled to the one or more processors, the memory storing computer-executable modules executable by the one or more processors that, when executed by the one or more processors, perform associated operations, the computer-executable modules comprising:

an encoder module;

a block coding submodule of the encoder module configured to code a current block of a frame;

a block motion predicting submodule of the encoder module configured to perform motion prediction upon at least a partition of the current block other than a first-coded partition of the current block with reference to a predictor block in the frame, the predictor block resulting from a non-local predictor block search; and

a reconstructing submodule of the encoder module configured to reconstruct the current block.

16. The system of claim 15, further comprising:
an optimal partition mode deciding submodule of the encoder module configured to decide an optimal partition mode for coding the current block.

17. The system of claim 16, further comprising:

a predictor block searching submodule configured to perform predictor block search for a predictor block corresponding to a respective first-coded partition for each partition mode of a plurality of partition modes of the current block; perform non-local predictor block search for a predictor block corresponding to each respective partition other than a first-coded partition for each partition mode of a plurality of partition modes of the current block; and

a cost function computing submodule configured to compute, for each partition mode, a value of a cost function based on each partition and the predictor block corresponding thereto;

wherein the optimal partition deciding submodule decides a partition mode of the current block having the lowest value of the cost function among all partition modes of the current block as an optimal partition mode.

18. The system of claim 17, wherein the predictor block searching submodule searches for a predictor block corresponding to a partition over all pixels of the current frame already reconstructed at the time of searching.

19. The system of claim 17, wherein the predictor block searching submodule performing non-local predictor block search for predictor blocks corresponding to at least two partitions of at least one partition mode is executable at least partially in parallel.

20. The system of claim 19, wherein the reconstructing submodule reconstructs the entirety of the current block after the predictor block searching submodule searching for a predictor block corresponding to each partition for each partition mode of a plurality of partition modes of the current block has been completed for each partition mode of the plurality of partition modes.

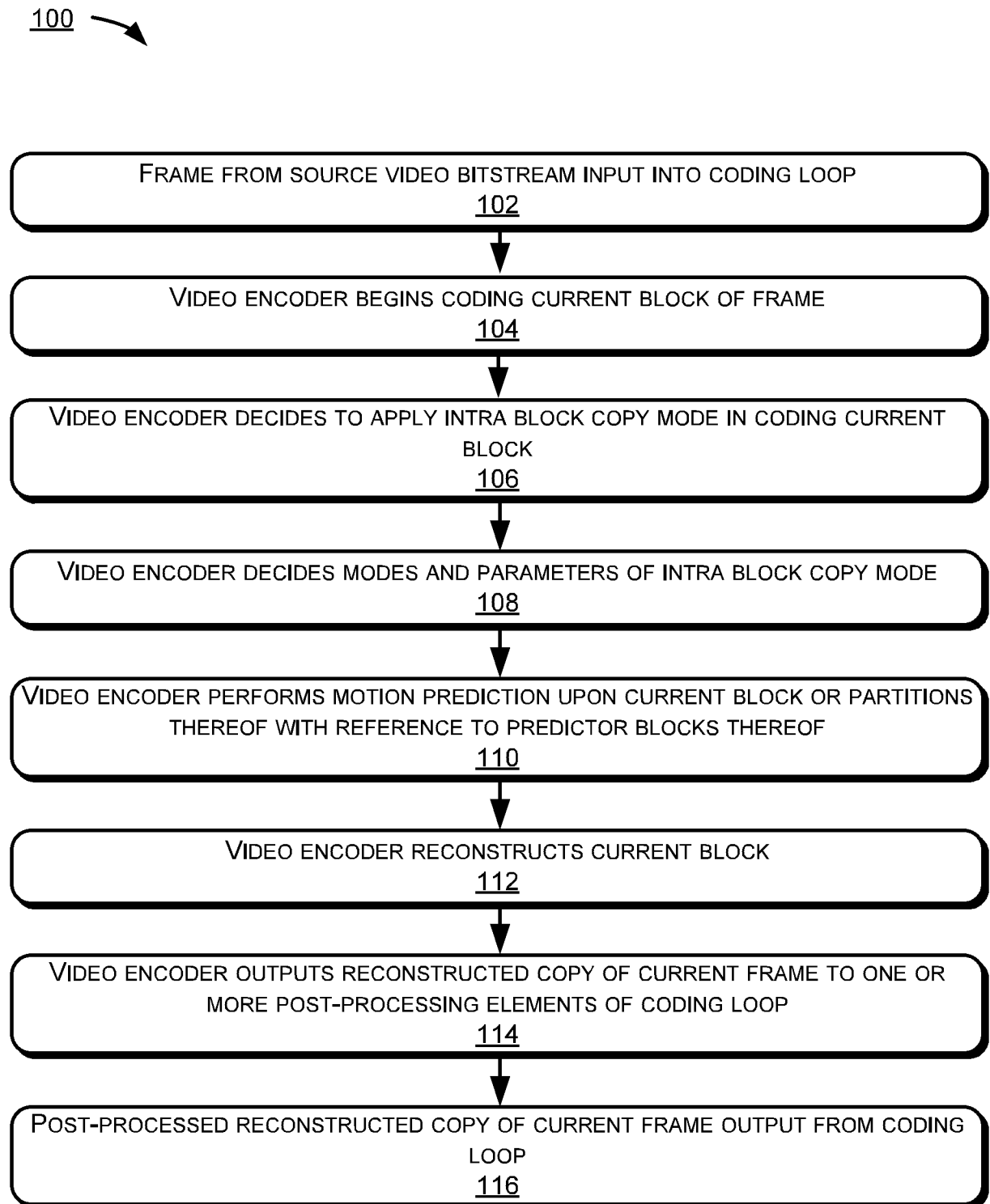
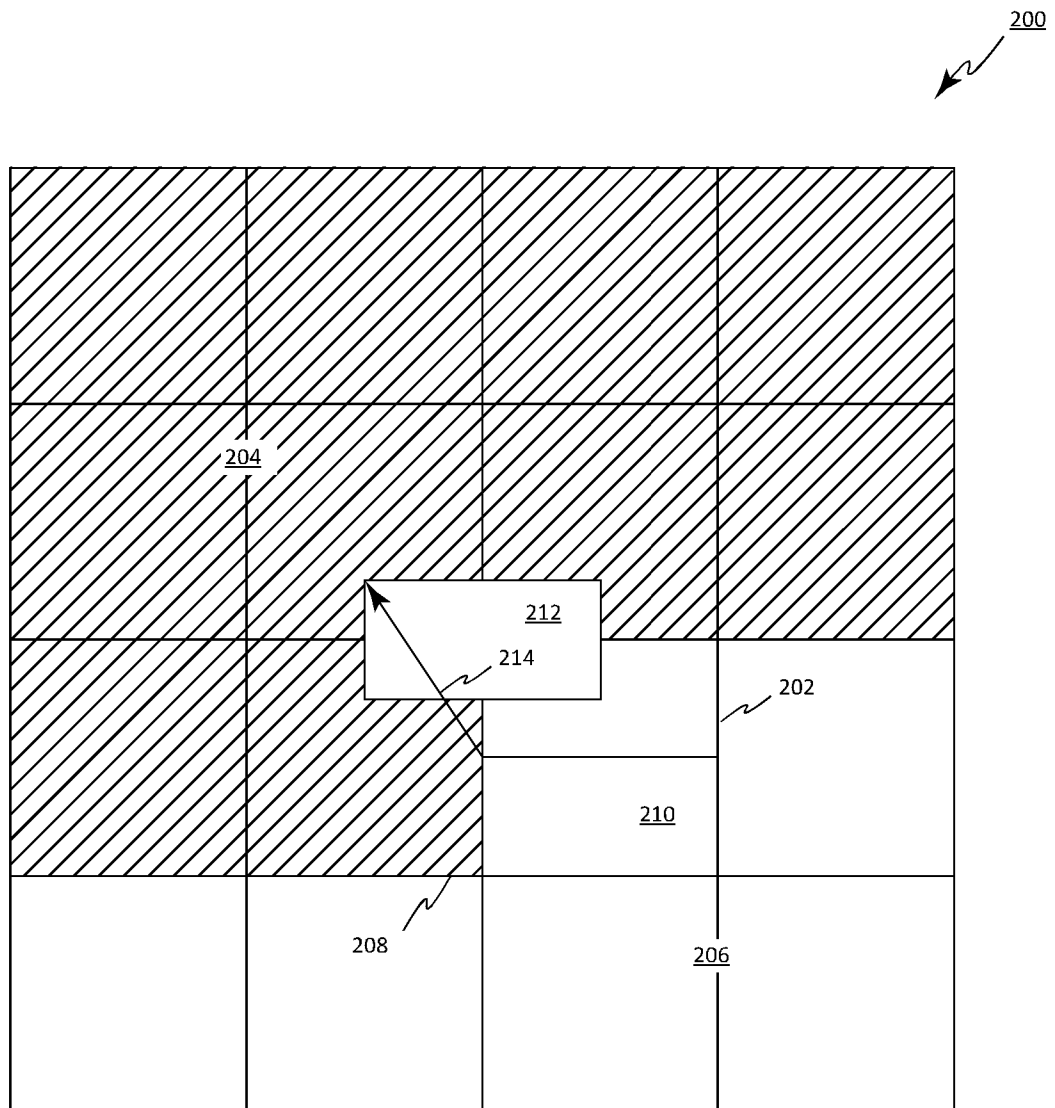


FIG. 1



PRIOR ART

FIG. 2

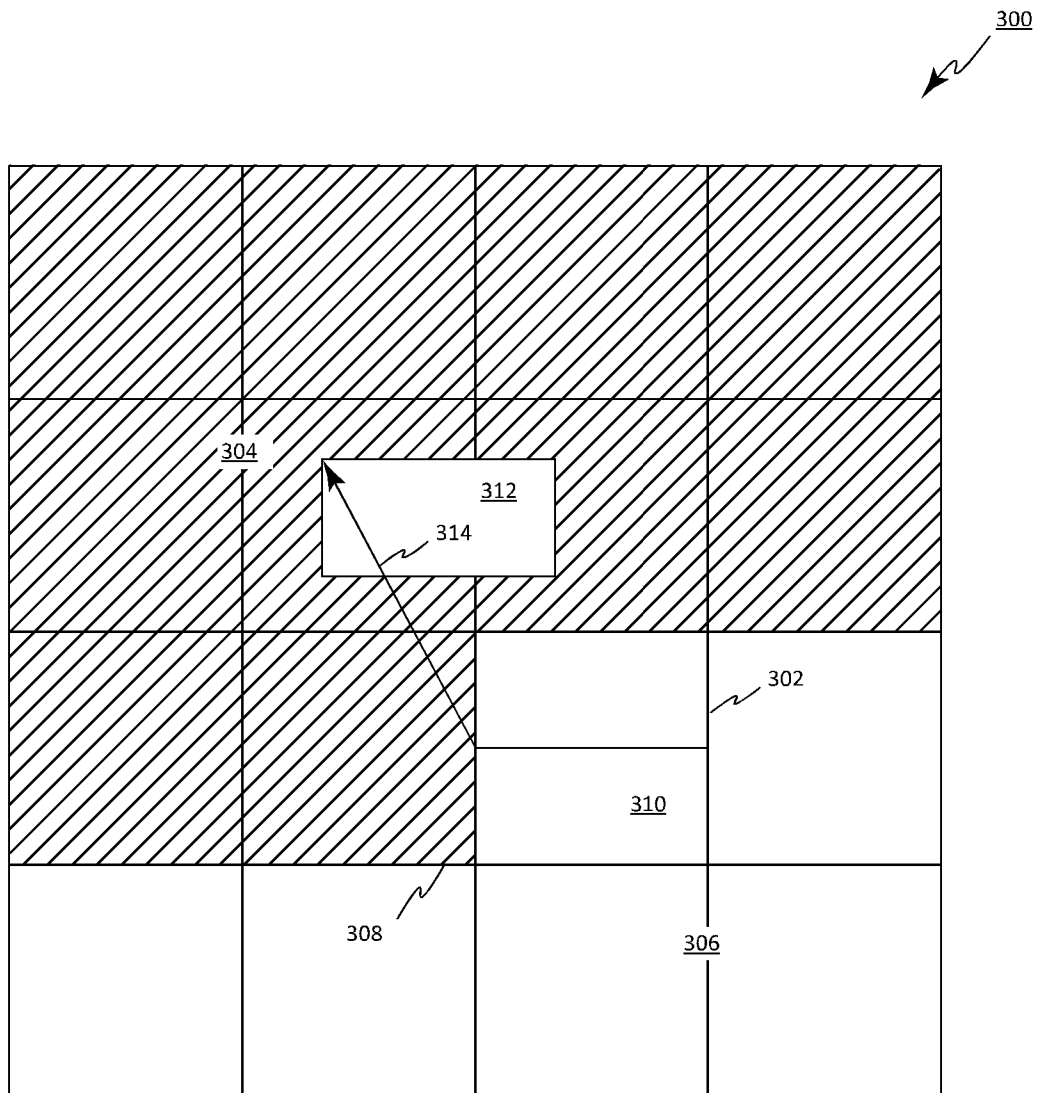
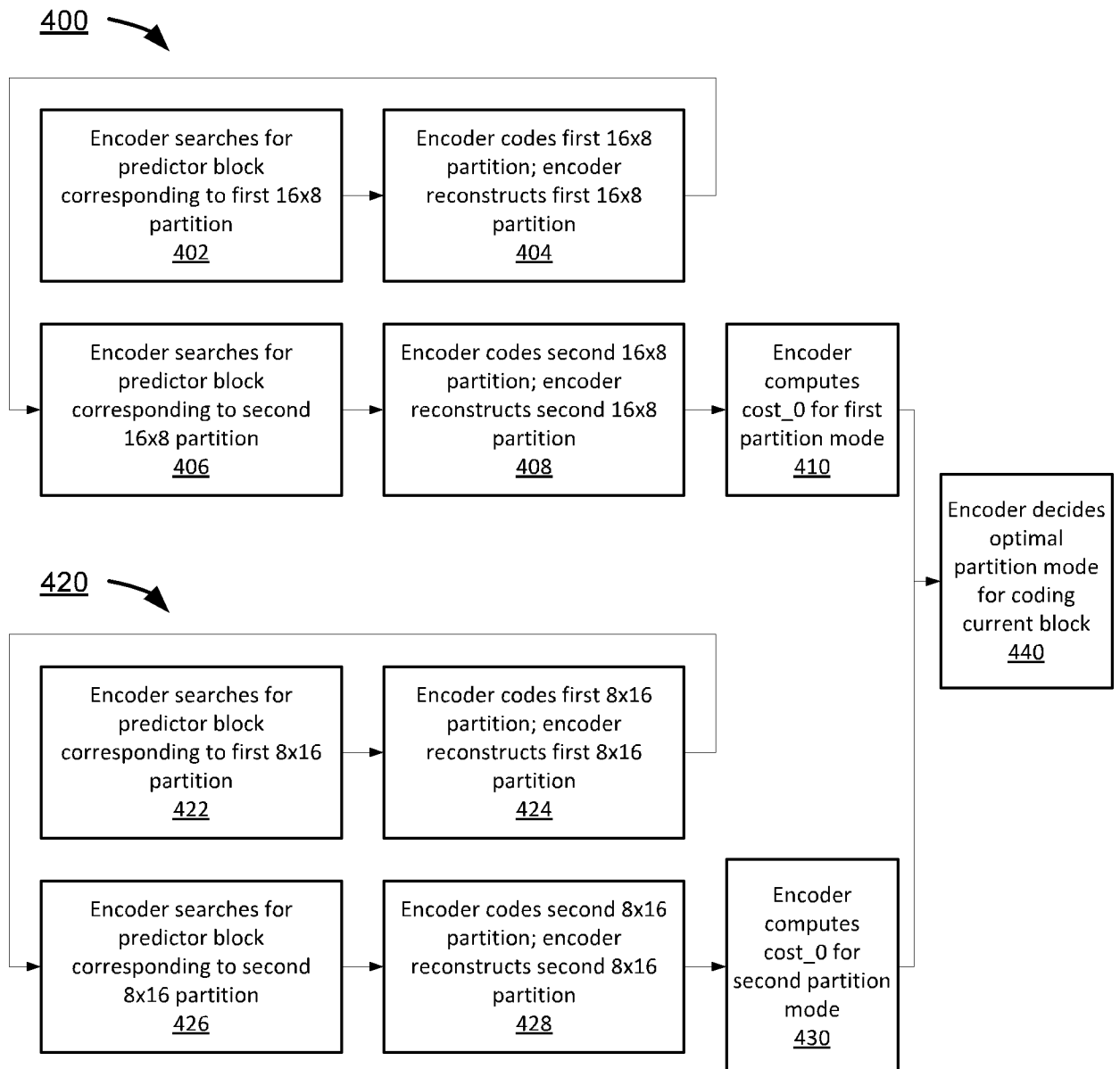


FIG. 3



PRIOR ART

FIG. 4

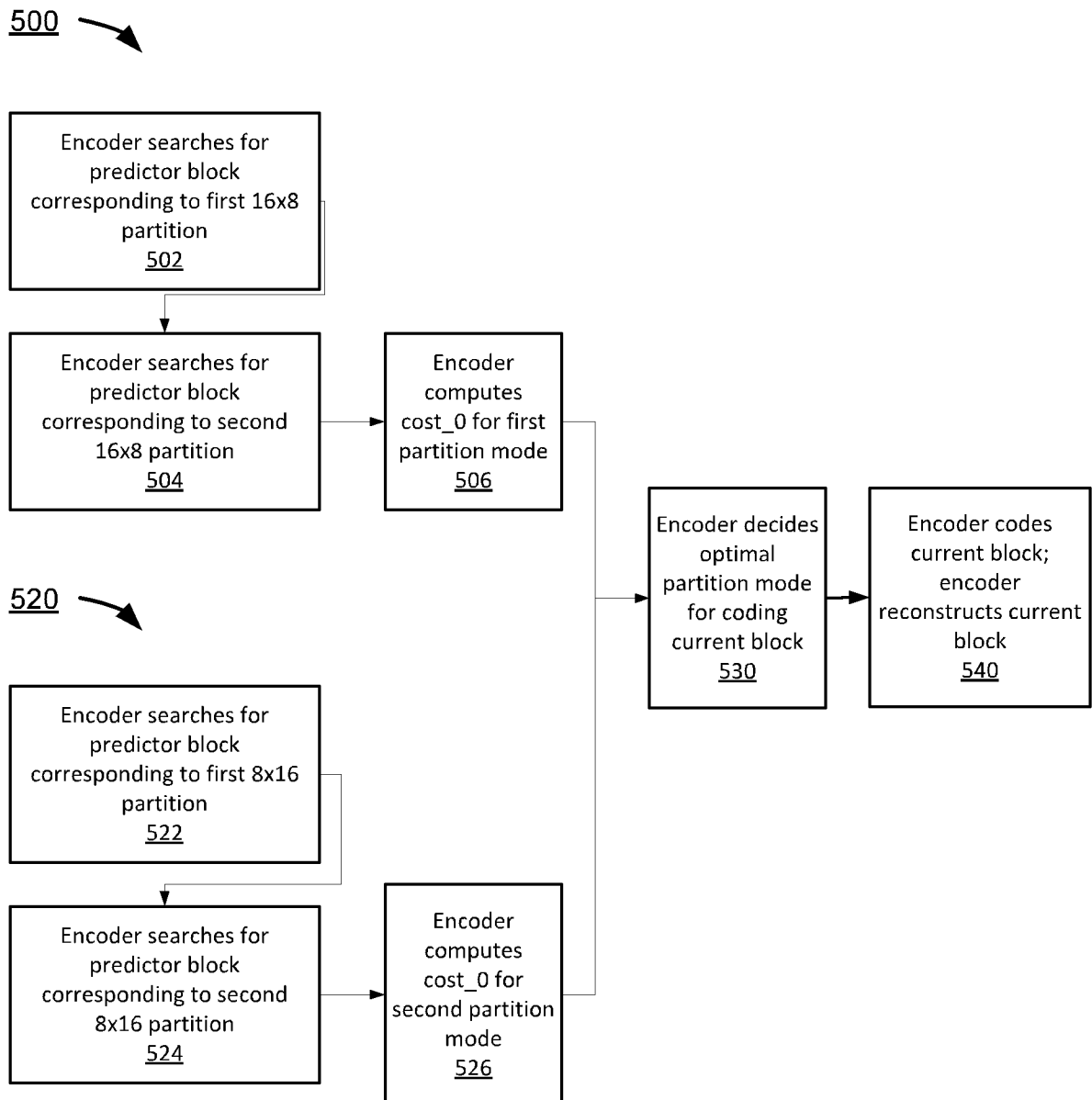


FIG. 5

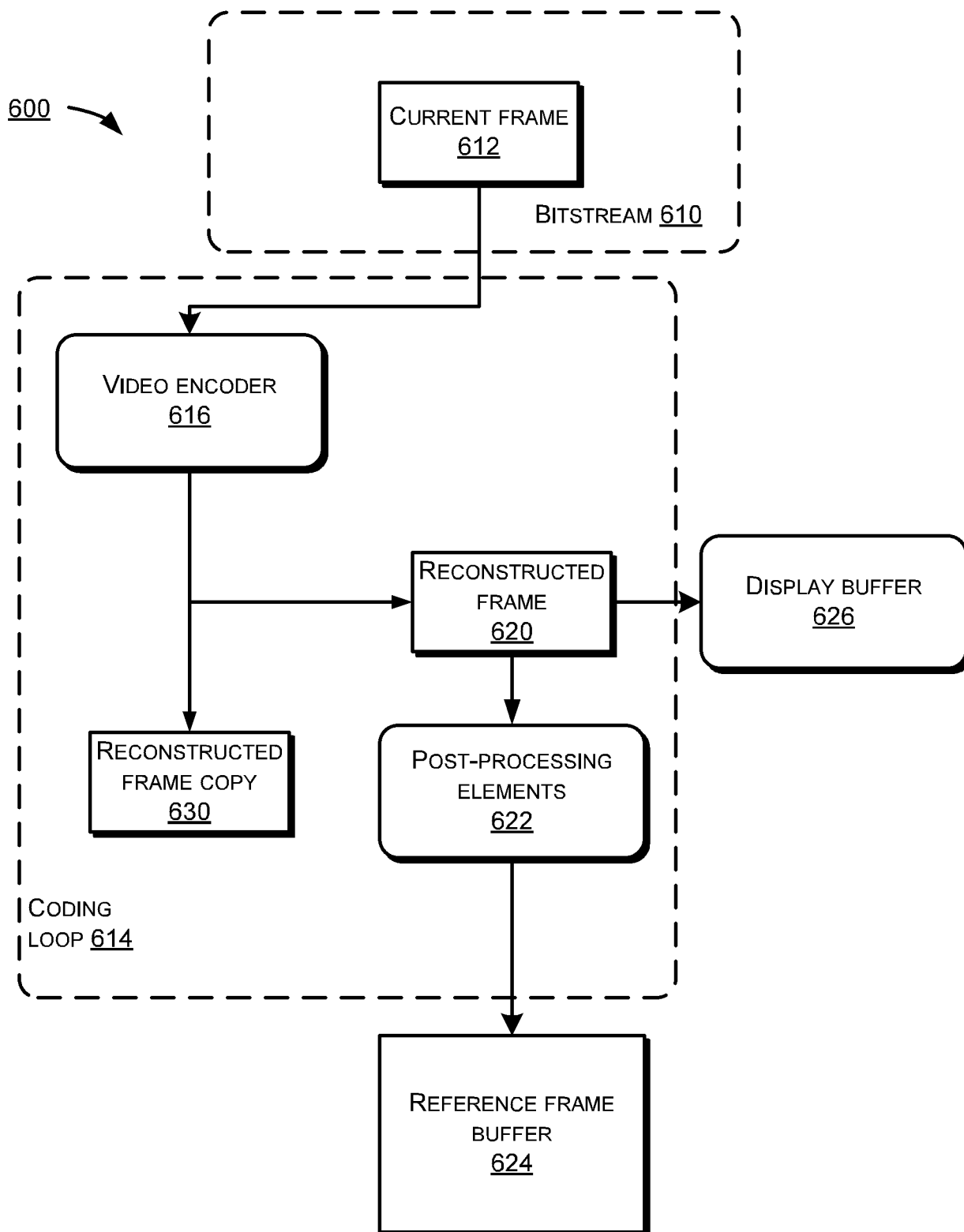
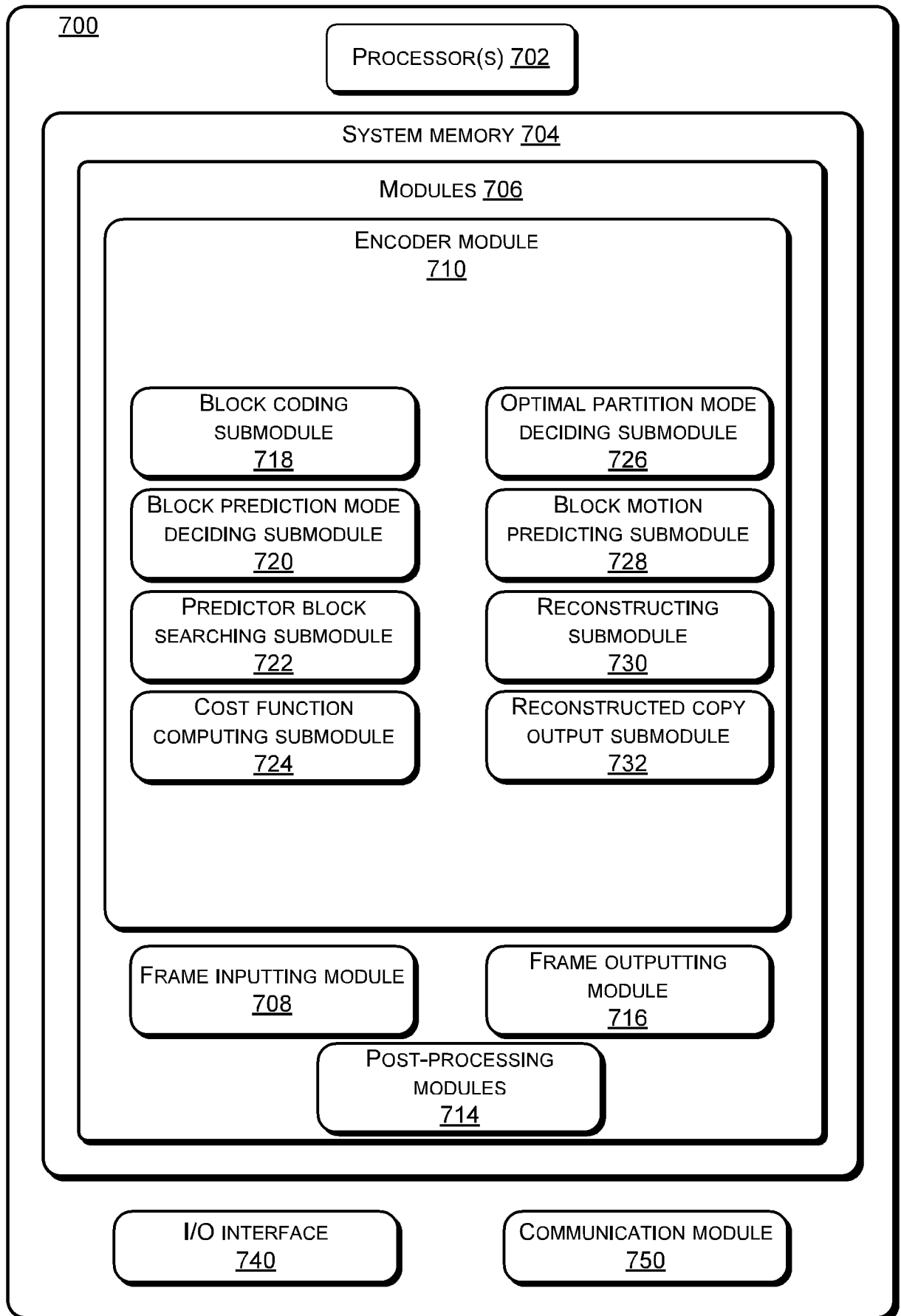


FIG. 6

**FIG. 7**

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2019/086654

A. CLASSIFICATION OF SUBJECT MATTER

H04N 19/513(2014.01)i; H04N 19/593(2014.01)i; H04N 19/503(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)

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)

CNPAT, CNKI, WPI, EPODOC, IEEE: predict, frame, block, partition, encode, motion, non-local, reconstruct, video, current, HEVC, IBC, CTU, pixel, BV, vector

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2013272410 A1 (QUALCOMM INCORPORATED) 17 October 2013 (2013-10-17) description, paragraphs [0024]-[0035]	1-20
A	CN 102196270 A (HUAWEI TECHNOLOGIES CO., LTD. et al.) 21 September 2011 (2011-09-21) the whole document	1-20
A	WO 2017058633 A1 (QUALCOMM INCORPORATED) 06 April 2017 (2017-04-06) the whole document	1-20
A	US 2014078250 A1 (QUALCOMM INCORPORATED) 20 March 2014 (2014-03-20) the whole document	1-20
A	CN 105430417 A (ZTE CORPORATION) 23 March 2016 (2016-03-23) the whole document	1-20



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

17 January 2020

Date of mailing of the international search report

12 February 2020

Name and mailing address of the ISA/CN

National Intellectual Property Administration, PRC
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing
100088
China

Authorized officer

WANG,Jian

Facsimile No. (86-10)62019451

Telephone No. 86-(10)-53961733

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2019/086654

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2013272410	A1	17 October 2013	KR	20150008394	A	22 January 2015
				EP	2837191	A2	18 February 2015
				WO	2013155267	A2	17 October 2013
				CN	104335586	A	04 February 2015
CN	102196270	A	21 September 2011	None			
WO	2017058633	A1	06 April 2017	KR	20180063094	A	11 June 2018
				EP	3357246	A1	08 August 2018
				CN	108141605	A	08 June 2018
				BR	112018006627	A2	23 October 2018
				JP	2018530249	A	11 October 2018
				AU	2016332326	A1	29 March 2018
				US	2017099495	A1	06 April 2017
				TW	201714455	A	16 April 2017
				IN	201847008321	A	06 April 2018
US	2014078250	A1	20 March 2014	CN	104704833	A	10 June 2015
				EP	2898692	A1	29 July 2015
				WO	2014047188	A1	27 March 2014
				IN	201501051	P4	01 July 2016
CN	105430417	A	23 March 2016	WO	2016045581	A1	31 March 2016
				US	2017302958	A1	19 October 2017
				EP	3200459	A1	02 August 2017