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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 April 2012 (05.04.2012)

PCT

(10) International Publication Number
WO 2012/044707 A1

(51) International Patent Classification:
H04N 7/26 (2006.01)

(21) International Application Number:
PCT/US2011/053761

(22) International Filing Date:
28 September 2011 (28.09.2011)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

61/388,895	1 October 2010 (01.10.2010)	US
61/388,741	1 October 2010 (01.10.2010)	US
13/247,190	28 September 2011 (28.09.2011)	US

(71) Applicant *(for all designated States except US)*: **GENERAL INSTRUMENT CORPORATION** [US/US]; 101 Tournament Drive, Horsham, Pennsylvania 19044 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **PANUSOPONE, Krit** [TH/US]; 10725 Calle Mar De Mariposa, San Diego, California 92130 (US). **FANG, Xue** [CN/US]; 16354 Pinto Ridge Drive, San Diego, California 92127 (US). **WANG, Limin** [US/US]; 13641 Shoal Summit Drive, San Diego, California 92128 (US).

(74) Agents: **PAYNE, Susan** et al.; 600 North US Highway 45, Libertyville, Illinois 60048 (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, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, 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, ML, MR, NE, SN, TD, TG).

Published:

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

(54) **Title**: CODING AND DECODING UTILIZING PICTURE BOUNDARY VARIABILITY IN FLEXIBLE PARTITIONING

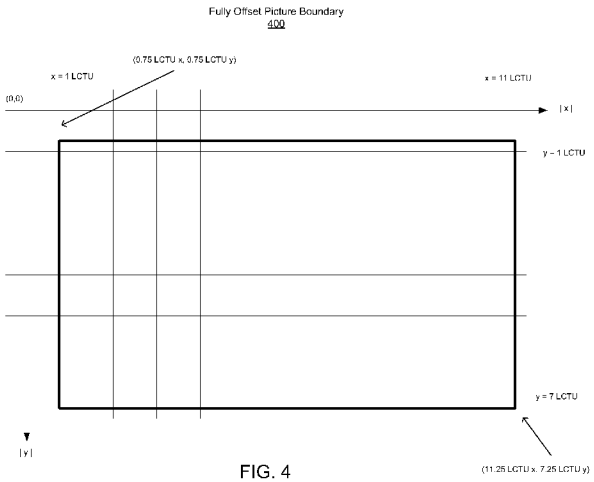


FIG. 4

(57) **Abstract**: There is a coding including preparing coding units based on source pictures. The coding units are associated with largest coding tree units (LCTUs) which are polygons of source pictures. A tree format is utilized in processing the LCTUs into coding units. The preparing includes calculating an efficiency measure associated with a source picture position in a coordinate system based on fitting the coordinate system and the source picture with respect to each other. The preparing includes determining the source picture position based on a coding efficiency goal. The source picture is divided into LCTUs based on the coordinate system and the determined source picture position. There is also a decoding including processing video compression data which is generated based on the coding units based on partitions of LCTUs.

WO 2012/044707 A1

CODING AND DECODING UTILIZING PICTURE BOUNDARY VARIABILITY IN FLEXIBLE PARTITIONING

PRIORITY

[001] The present application claims the benefit of priority to U.S. Provisional Patent Application Serial No. 61/388,741, filed on October 1, 2010, entitled "Flexible Picture Partitioning", by Krit Panusopone, et al., and to U.S. Provisional Patent Application Serial No. 61/388,895, also filed on October 1, 2010, and also entitled "Flexible Picture Partitioning", by Krit Panusopone, et al., the disclosures of which are hereby incorporated by reference in their entirety.

CROSS REFERENCE TO RELATED APPLICATIONS

[002] The present application is related to U.S. Utility Patent Application Serial No. 13/247190, filed on September 28, 2011, entitled "Coding and Decoding Utilizing Picture Boundary Padding in Flexible Partitioning", by Krit Panusopone, et al., which claims priority to U.S. Provisional Patent Application Serial No. 61/391, 350, filed on October 8, 2010, entitled "Arbitrarily Padding", by Krit Panusopone, et al., the disclosure of which is hereby incorporated by reference in its entirety.

BACKGROUND

[003] Video compression utilizes block processing for many operations. In block processing, a block of neighboring pixels is grouped into a coding unit and compression operations treat this group of pixels as one unit to take advantage of correlations among neighboring pixels within the coding unit. Theoretically, a larger coding unit is commonly preferred to reduce the overhead associated with processing multiple smaller coding units instead of one larger

coding unit for the same part of a picture. The larger coding unit is also preferred because the bandwidth used to transmit information associated with processing a single larger coding unit is often lower than for multiple smaller coding units.

[004] Coding units having block sizes of 8x8 and 16x16 pixels have been utilized in earlier video compression standards; e.g., MPEG-1, MPEG-2 and MPEG-4 AVC. In these earlier standards, a coding system utilized a fixed block size for block processing. Partitioning pictures based on fixed block size is relatively simple. If the processing order is predetermined, such as with a raster scan pattern scanning order, the earlier coding systems merely relied upon a block index to specify a location of a coding unit within a picture area.

[005] High Efficiency Video Coding (HEVC) is a new video compression standard which has been proposed as a successor to MPEG-4 AVC. One goal in developing HEVC is to standardize an improved coding efficiency compared with the "high profile" of MPEG-4 AVC. The high profile of MPEG-4 AVC is associated with high definition television (HDTV). Another goal in developing HEVC is to reduce the bitrate requirements of transmitting HDTV compressed video data while also maintaining comparable image quality with the MPEG-4 AVC high profile.

[006] Among the proposals made for HEVC are those including concepts relating to flexible block size partitioning, or simply flexible partitioning. Flexible partitioning adds flexibility over fixed block size partitioning by utilizing a range of sizes associated with the coding units of a partitioned picture. In flexible partitioning, a picture is initially divided into equal sized square blocks called largest coding tree units (LCTUs). The size of the LCTUs utilized to partition pictures may be substantially higher than the fixed block sizes used in earlier standards. After a picture has been divided into a group of LCTUs, the individual LCTUs in the group are often partitioned into coding units which commonly include a range of sizes. However, in some circumstances an LCTU may not be

partitioned. In these circumstances, the LCTU is associated with a single coding unit having a size equivalent to the LCTU.

[007] The partitioning of an LCTU is commonly performed utilizing a quadtree format. The quadtree format is a recursive partitioning process following a tree structure having layers. At the top layer, the complete LCTU is a parent of the quadtree. If there is no partitioning, the complete LCTU is also a coding unit. In flexible partitioning according to the quadtree format, the parent is commonly divided into four leaves. Each leaf represents an equivalent size quadrant of the parent and is a square block, like the parent. A leaf in the quadtree may also form a coding unit or be further partitioned. A leaf which is further partitioned is a parent of its leaves in the next layer of the quadtree.

[008] A quadtree of a parent LCTU often continues to be divided recursively through one or more leaves at different layers. The unpartitioned leaves at each layer commonly form coding units of different sizes, all being based on the parent LCTU. The unpartitioned leaves at each layer represent coding units having a predetermined condition, such as a measure of homogeneity associated with the pixels in a coding unit. The quadtree format is commonly utilized in video processing applications due to its efficiency in representing pictures. The recursive nature of the quadtree, in general, requires little overhead to represent the various sized coding units of a parent LCTU.

[009] In the HEVC models which have been considered, pictures are commonly divided into LCTUs utilizing a coordinate system, such as a Cartesian coordinate system. A complete picture occupies only that part of a single quadrant of the coordinate system which is closest to the origin. The coordinate system marks the location of all the LCTUs in a group of LCTUs associated with the picture. The LCTUs in this group are marked by the coordinate system with respect to the picture. Also, the lengths of the axes nearer the origin of the coordinate system, on the boundaries of the single quadrant, contact two of the picture's boundaries. Block processing according to these HEVC models has

certain inefficiencies. These are due to the fixed location of the picture in the quadrant of the coordinate system. The inefficiencies are also due to the fixed locations of the marked LCTUs with respect to the fixed picture location.

[0010] One reason for these inefficiencies is because the location of the LCTUs with respect to the picture are all fixed. Since they are fixed, the quadtree format often necessitates recursively dividing an LCTU into very small coding units in order to attain a measure of homogeneity associated with the pixels in all the partitioned coding units of the LCTU. A greater number of smaller coding units, as compared with a smaller number of larger coding units, often requires a higher overhead to generate and process for all the coding units associated with a picture. Also more bandwidth is commonly required to transmit compression data associated with the greater number of small coding units when they are packaged in a compressed video bitstream.

[0011] Also, a picture may have pixels in areas which fall into incomplete LCTUs located farther from the coordinate system axes. These LCTUs are irregular due to the picture boundary of the fixed picture not extending to fill these boundary issue LCTUs. This is a boundary issue that commonly occurs when a height or length of a picture are not some complete multiple of the dimensions of an LCTU size used for block processing the picture.

[0012] In the HEVC models which have been considered, attempts to address boundary issues commonly include iteratively partitioning the quadtree format leaves of the boundary issue LCTUs (i.e., those LCTUs containing at least some area the picture to form coding units) while ignoring the leaves in the boundary issue LCTUs which contain no area of the picture. The partitioning repeats with the leaves containing a part of the picture until these partitioned leaves become square in shape. This methodology in attempting to address the boundary issue often causes degradation in coding efficiency. It commonly requires the coding units in the boundary issue LCTUs to be partitioned into smaller blocks than optimal. The degradation in coding efficiency in these

circumstances is more common when larger LCTU sizes are utilized and/or when the pictures being partitioned involve lower video resolutions.

SUMMARY

[0013] According to principles of the invention, there are systems, methods, and computer readable mediums (CRMs) which provide for coding and decoding utilizing picture boundary variability. By utilizing picture boundary variability, LCTUs and/or pictures may be freely located with respect to each other so that larger coding units may be partitioned. This increases the coding efficiency associated with the processing overhead and/or bandwidth required to generate and/or to transmit video compression data associated with the coding units prepared utilizing picture boundary variability.

[0014] According to a first principle of the invention, there is a system for coding. The system may include a processor configured to prepare coding units based on source pictures. The preparing may include calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other. The coordinate system may include two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin. The preparing may also include determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal. The preparing may also include dividing the source picture into a plurality of LCTUs based on the determined source picture position.

[0015] According to a second principle of the invention, there is a method for coding. The method may include preparing coding units based on source pictures utilizing a processor. The preparing may include calculating an

efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other. The coordinate system may include two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin. The preparing may also include determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal. The preparing may also include dividing the source picture into a plurality of LCTUs based on the determined source picture position.

[0016] According to a third principle of the invention, there is a non-transitory computer readable medium (CRM) storing computer readable instructions which, when executed by a computer system, performs a method for coding. The method may include preparing coding units based on source pictures utilizing a processor. The preparing may include calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other. The coordinate system may include two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin. The preparing may also include determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal. The preparing may also include dividing the source picture into a plurality of LCTUs based on the determined source picture position.

[0017] According to a fourth principle of the invention, there is a system for decoding. The system may include an interface configured to receive video compression data. The system may also include a processor configured to process the received video compression data. The received video compression data may be based on coding units, based on source pictures. The coding units

may be prepared by steps including calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other. The coordinate system may include two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin. The preparing may also include determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal. The preparing may also include dividing the source picture into a plurality of LCTUs based on the determined source picture position.

[0018] According to a fifth principle of the invention, there is a method for decoding. The method may include receiving video compression data. The method may also include processing the received video compression data utilizing a processor. The received video compression data may be based on coding units, based on source pictures. The coding units may be prepared by steps including calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other. The coordinate system may include two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin. The preparing may also include determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal. The preparing may also include dividing the source picture into a plurality of LCTUs based on the determined source picture position.

[0019] According to a sixth principle of the invention, there is a non-transitory computer readable medium (CRM) storing computer readable instructions which, when executed by a computer system, performs a method for decoding. The method may include receiving video compression data. The

method may also include processing the received video compression data utilizing a processor. The received video compression data may be based on coding units, based on source pictures. The coding units may be prepared by steps including calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other. The coordinate system may include two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin. The preparing may also include determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal. The preparing may also include dividing the source picture into a plurality of LCTUs based on the determined source picture position.

[0020] These and other objects are accomplished in accordance with the principles of the invention in providing systems, methods and CRMs which code and decode utilizing picture boundary variability. Further features, their nature and various advantages will be more apparent from the accompanying drawings and the following detailed description of the preferred embodiments.

BRIEF DESCRIPTION OF THE DRAWINGS

[0021] Features of the examples and disclosure are apparent to those skilled in the art from the following description with reference to the figures, in which:

[0022] FIG 1 is a block diagram illustrating a coding system and a decoding system utilizing picture boundary variability, according to an example;

[0023] FIG 2 is a graph showing a fixed picture boundary in a coordinate system, according to a comparative example;

[0024] FIG 3 is a partially offset picture boundary in a coordinate system,

according to an example;

[0025] FIG 4 is a fully offset picture boundary in a coordinate system, according to an example;

[0026] FIG 5 is a flow diagram illustrating a method for preparing coding units utilizing picture boundary variability, according to an example;

[0027] FIG 6 is a flow diagram illustrating a method for coding utilizing picture boundary variability, according to an example;

[0028] FIG 7 is a flow diagram illustrating a method for decoding utilizing picture boundary variability, according to an example; and

[0029] FIG 8 is a block diagram illustrating a computer system to provide a platform for a system for coding and/or a system for decoding utilizing picture boundary variability, according to examples.

DETAILED DESCRIPTION

[0030] For simplicity and illustrative purposes, the present invention is described by referring mainly to embodiments, principles and examples thereof. In the following description, numerous specific details are set forth in order to provide a thorough understanding of the examples. It is readily apparent however, that the embodiments may be practiced without limitation to these specific details. In other instances, some methods and structures have not been described in detail so as not to unnecessarily obscure the description. Furthermore, different embodiments are described below. The embodiments may be used or performed together in different combinations.

[0031] As used herein, the term “includes” means “includes at least” but is not limited to the term “including only”. The term “based on” means “based at least in part on”. The term “picture” means a picture which is either equivalent to a frame or equivalent to a field associated with a frame, such as a field which is

one of two sets of interlaced lines of an interlaced video frame. The term "coding" may refer to the encoding of an uncompressed video bitstream. The term "coding" may also refer to the transcoding a compressed video bitstream from one compressed format to another.

[0032] As demonstrated in the following examples and embodiments, there are systems, methods, and machine readable instructions stored on computer-readable media (e.g., CRMs) for coding and decoding utilizing picture boundary variability. Referring to FIG 1, there is disclosed a content distribution system 100 including a coding system 110 and decoding system 140 which utilize picture boundary variability associated coding units based on video frames or pictures. The prepared coding units are utilized in generating video compression data, according to an example. The coding system 110 and a decoding system 140 are described in greater detail below after the following detailed description of picture boundary variability.

[0033] FIG 2 is an example of a picture which may be partitioned based on LCTUs according to a default mode. In the default mode, LCTUs are fixedly located with respect to the pictures based only upon their location in a coordinate system fixedly superimposed upon the picture. In the default mode, the origin of the superimposed coordinate system, as a coordinate system locus, is always located at the corner of a fixed picture boundary as a source picture locus which may be fitted to the coordinate system in a plane. In FIG 2, the source picture locus associated with the upper left corner of a picture coincides with the coordinate system locus occurring at the origin (0,0) of the superimposed coordinate system. It occurs in the default mode so as to coincide with a corner of a source picture corner LCTU (i.e., an LCTU which occurs at a corner of the source picture) in the plurality of LCTUs associated with a source picture.

[0034] Referring to FIG 2, there is disclosed a fixed picture boundary 200 (i.e., the bold line rectangle) of the picture. The periphery of a picture, such as a picture in a video sequence, may be described by the fixed picture boundary 200.

The coordinate system may have an origin (0,0), a horizontal "x-axis" and a vertical "y-axis". The fixed picture boundary 200 may be superimposed by a single quadrant of the coordinate system. The quadrant may be described by the two axes of the coordinate system.

[0035] Each of the axes may be described by a number line marked by equivalent line lengths separating marking points along each number line. The intersections of lines perpendicular to each marking point may describe a corner of a polygon, such as a square. Each square may represent an LCTU associated with an area of a picture within the fixed picture boundary 200. Each side of the squares may represent an LCTU side length. The marking points of the number line describing each axis may be an absolute value of a multiple of a value of an LCTU side length. For example, all the pictures in a video sequence may be partitioned by first being divided up into LCTUs having an LCTU size such as 64x64 pixels, 128x128 pixels, etc.

[0036] In the example of FIG 2, the fixed picture boundary 200 and the coordinate system are fixed in their location with respect to each other. The coordinate system describing the locations of the LCTUs is superimposed over the fixed picture boundary 200 such that the x-axis and y-axis of a single quadrant of the coordinate system always coincide with a side of the picture boundary of the pictures. In this circumstance, the origin (0,0) of the coordinate system also coincides with a corner of the picture boundary 200. The LCTUs near to the origin are filled with a corresponding square area of the picture associated with the fixed picture boundary 200. The LCTUs in the column and row farthest from the origin are only partially filled by an area of the picture. These partially filled LCTUs are boundary issue LCTUs.

[0037] The coordinate system used in determining the location of the LCTUs may always be superimposed this way with respect to source pictures. If no other consideration is made with respect the placement of the coordinate system with respect to a picture boundary of a source picture, then LCTUs

determined by this placement may generate a greater number coding units. A greater number of coding units may be due to the quad tree format leaves of boundary issue LCTUs being partitioned iteratively to form square shaped leaves that are filled. A greater number of coding units may be due to excessive partitioning before a homogeneity rule is reached with respect to the pixels within the coding units.

[0038] Greater numbers of smaller coding units may be required to address the boundary issue LCTUs and/or to reach a homogeneity rule, for example, if the placement of the LCTUs fails to take into consideration the location of objects in the pictures of the video sequence. Or in another example, this may also occur if the placement of LCTUs fails to consider the location of motion within pictures of a video sequence. In either circumstance, the partitioning of the LCTUs may result in a greater number of smaller coding units, thus requiring more overhead to generate and process all the coding units associated with a picture. Also more bandwidth may be required to transmit compression data associated with the greater number of smaller coding units when packaged in a compressed video bitstream.

[0039] Fig 2 depicts a default mode according to an example showing a fixed picture boundary 200. In the default mode depicted in FIG 2, all the LCTUs nearer to the axes of the coordinate system superimposed on the fixed picture boundary 200 are associated with an area of a picture which completely fill the square area of these LCTUs. However, the LCTUs which occur at the 11^{th} LCTU along the x-axis are boundary issue LCTUs. The LCTUs which occur at the 7^{th} LCTU along the y-axis are also boundary issue LCTUs. A boundary issue LCTU is an incomplete or partially filled LCTU and may have any portion that is less than 100% of its LCTU area associated with a picture. According the example exemplified in FIG 2, the boundary issue LCTUs have 50% or less of their LCTU area associated with any part of a picture in the video sequence. Note that the boundary issue LCTU located at the outer corner of the fixed picture boundary

200 occurring at (10.5 LCTU x, 6.5 LCTU y) has only 25% of its LCTU area associated with any part of a picture. The default mode may be selected based upon a determination of a homogeneity goal associated with pixels in prepared coding units. However, if the placement of the fixed picture boundary 200 is made without consideration of the boundary issue in partitioning these LCTUs, this may also result in generating a larger number of smaller coding units.

[0040] Referring to FIG 3, there is shown a partially offset picture boundary 300 as an example of source picture which may be flexibly partitioned according to a corner mode. In the corner mode, LCTUs may be located with respect to the pictures based upon a determination of a homogeneity goal associated with pixels in prepared coding units. The determination includes some consideration of a location of a coordinate system with respect to a picture upon which it is superimposed. The determination of the homogeneity goal may include consideration of numerous factors, including the location of objects or motion in the pictures of the video sequence. In the corner mode, a first locus of the source picture position occurs at a nearest to the origin corner of a first source picture corner LCTU located nearest to the origin. The first locus may be separated from the origin of the coordinate system by an offset distance. In FIG 3, the first locus occurs at (0.5 LCTU x, 0.5 LCTU y).

[0041] According to an example, the corner mode may coincide with the default mode. In this circumstance, the first locus coincides with the origin. In the corner mode, a second locus coincides with a corner of a second source picture corner LCTU also coinciding with a farthest from the origin corner of a second source picture corner LCTU. In FIG 3, one second locus occurs at (11 LCTU x, 7 LCTU y). Other second loci may occur at the other two source picture corner LCTUs. In FIG 3, these appear at (11 LCTU x, 1 LCTU y) and (0.5 LCTU x, 7 LCTU y). All these second loci coincide with a line intersection forming a corner of an LCTU at the source picture corner. The coordinate system and the pictures associated with the picture boundary are otherwise as described above with respect to FIG 2.

[0042] In the corner mode, the outside corner of the picture boundary which is located furthest from the origin of the coordinate system may be fitted to an outside corner of an LCTU in the coordinate system based on a determination that the shifting of the picture boundary placement away from the origin will increase coding efficiency. In the partially offset picture boundary 300, a coordinate system locus and a picture boundary locus coincide at the coordinate pair (11 LCTU x, 7 LCTU y). The corner mode increases coding efficiency while utilizing very little overhead. According to an example, the corner mode may require only 2 bits of overhead to indicate directions of the horizontal shift and/or the vertical shift associated with the partially offset picture boundary 300 in the coordinate system.

[0043] Referring to FIG 4, there is shown a fully offset picture boundary 400 (i.e., the bold line rectangle), as an example of a picture which may be flexibly partitioned according to an explicit mode. The periphery of pictures in a video sequence may be described by the fully offset picture boundary 400. The coordinate system and the picture boundary are otherwise as described above with respect to FIG 2 and FIG 3.In the example demonstrated in FIG 4, the pictures associated with the coordinate system are free located with respect to each other. Based on the homogeneity determination and/or the determination based on a boundary issue consideration or some combination thereof, the fully offset picture boundary 400 may be shifted away from the origin (0,0) and/or the LCTU sides as shown in FIG 4.

[0044] The explicit mode includes highly precise placement of the fully offset picture boundary 400 and may be set at any desired accuracy, such as a 1 pixel interval, a 4 pixels interval, an 8 pixels interval, a 16 pixel interval, etc. Picture analysis may be utilized to determine an offset vector for flexible partitioning in the explicit mode. The offset vector may include positioning the picture within the quadrant by an angular orientation of a side of the picture boundary with respect to an axis of the two axes, such as rotating the picture in

the plane of the coordinate system within the quadrant so as to increase the coding efficiency based on some aspect such as texture or motion associated with a feature of the picture, such as an object or background in the picture.

[0045] Flexible partitioning utilizing picture boundary variability may improve coding efficiency as the prepared coding units are more likely to be fitted to picture content in source pictures. For example, a source picture may contain a simple background area at the bottom and a more detailed area to the top. In this circumstance, flexible partitioning utilizing picture boundary variability may prepare larger coding units associated with the background in the bottom of the source picture and thus provide higher coding efficiency. A coding system or device may analyze picture content to determine picture boundary offsets to improve the flexible partitioning of the source picture.

[0046] Referring again to FIG 1, the coding system 110 includes an input interface 130, a controller 111, a counter 112, a frame memory 113, an encoding unit 114, a transmitter buffer 115 and an output interface 135. The decoding system 140 includes a receiver buffer 150, a decoding unit 151, a frame memory 152 and a controller 153. The coding system 110 and the decoding system 140 are coupled to each other via a transmission path including a compressed bitstream 105. The controller 111 of the coding system 110 controls the amount of data to be transmitted on the basis of the capacity of the receiver buffer 150 and may include other parameters such as the amount of data per a unit of time. The controller 111 controls the encoding unit 114, to prevent the occurrence of a failure of a received signal decoding operation of the decoding system 140. The controller 111 may be a processor or include, for example, a microcomputer having a processor, a random access memory and a read only memory.

[0047] A source bitstream 120 supplied from, for example, a content provider may include a video sequence of frames including source pictures in the video sequence. The source bitstream 120 may be uncompressed or compressed. If the source bitstream 120 is uncompressed, the coding system

110 may be associated with an encoding function. If the source bitstream 120 is compressed, the coding system 110 may be associated with a transcoding function. Coding units may be derived from the source pictures utilizing the controller 111. The frame memory 113 may have a first area which may used for storing the incoming source pictures from the source bitstream 120 and a second area may be used for reading out the source pictures and outputting them to the encoding unit 114. The controller 111 may output an area switching control signal 123 to the frame memory 113. The area switching control signal 123 may indicate whether the first area or the second area is to be utilized.

[0048] The controller 111 outputs an encoding control signal 124 to the encoding unit 114. The encoding control signal 124 causes the encoding unit 114 to start an encoding operation such as preparing the coding units based on a source picture. In response to the encoding control signal 124 from the controller 111, the encoding unit 114 starts to read out the prepared coding units to a high-efficiency encoding process, such as a prediction coding process or a transform coding process which process the prepared coding units generating video compression data based on the source pictures associated with the coding units.

[0049] The encoding unit 114 may package the generated video compression data in a packetized elementary stream (PES) including video packets. The encoding unit 114 may map the video packets into an encoded video signal 122 using control information and a program time stamp (PTS) and the encoded video signal 122 may be signaled to the transmitter buffer 115.

[0050] The encoded video signal 122 including the generated video compression data may be stored in the transmitter buffer 114. The information amount counter 112 is incremented to indicate the total amount of data in the transmitted buffer 115. As data is retrieved and removed from the buffer, the counter 112 may be decremented to reflect the amount of data in the transmitter buffer 114. The occupied area information signal 126 may be transmitted to the counter 112 to indicate whether data from the encoding unit 114 has been added

or removed from the transmitted buffer 115 so the counter 112 may be incremented or decremented. The controller 111 may control the production of video packets produced by the encoding unit 114 on the basis of the occupied area information 126 which may be communicated in order to prevent an overflow or underflow from taking place in the transmitter buffer 115.

[0051] The information amount counter 112 may be reset in response to a preset signal 128 generated and output by the controller 111. After the information counter 112 is reset, it may count data output by the encoding unit 114 and obtain the amount of video compression data and/or video packets which has been generated. Then, the information amount counter 112 may supply the controller 111 with an information amount signal 129 representative of the obtained amount of information. The controller 111 may control the encoding unit 114 so that there is no overflow at the transmitter buffer 115.

[0052] The decoding system 140 includes an input interface 170, a receiver buffer 150, a controller 153, a frame memory 152, a decoding unit 151 and an output interface 175. The receiver buffer 150 of the decoding system 140 may temporarily store the compressed bitstream 105 including the received video compression data and video packets based on the source pictures from the source bitstream 120. The decoding system 140 may read the control information and presentation time stamp information associated with video packets in the received data and output a frame number signal 163 which is applied to the controller 153. The controller 153 may supervise the counted number of frames at a predetermined interval, for instance, each time the decoding unit 151 completes a decoding operation.

[0053] When the frame number signal 163 indicates the receiver buffer 150 is at a predetermined capacity, the controller 153 may output a decoding start signal 164 to the decoding unit 151. When the frame number signal 163 indicates the receiver buffer 150 is at less than a predetermined capacity, the controller 153 may wait for the occurrence of a situation in which the counted number of frames becomes equal to the predetermined amount. When the frame

number signal 163 indicates the receiver buffer 150 is at the predetermined capacity, the controller 153 may output the decoding start signal 164. The encoded video packets and video compression data may be decoded in a monotonic order (i.e., increasing or decreasing) based on presentation time stamps associated with the encoded video packets.

[0054] In response to the decoding start signal 164, the decoding unit 151 may decode data amounting to one picture associated with a frame and compressed video data associated with the picture associated with video packets from the receiver buffer 150. The decoding unit 151 may write a decoded video signal 162 into the frame memory 152. The frame memory 152 may have a first area into which the decoded video signal is written, and a second area used for reading out a decoded bitstream 160 to the output interface 175.

[0055] The different modes described above associated with picture boundary variability (e.g., corner mode, explicit mode) may be implemented through the HEVC models under consideration through a syntax change to headers of video packets in the compressed bitstream 105. The syntax changes may be implemented at different layers of a video sequence, such as at the sequence, picture and/or slice layer.

[0056] TABLE I shows a syntax change, highlighted in boldface, which may be implemented in an HEVC header at the sequence layer.

[0057] TABLE I - Syntax change at sequence layer.

seq_parameter_set_rbsp( ) {	C	Descriptor
profile_idc	0	u(8)
reserved_zero_8bits /* equal to 0 */	0	u(8)
level_idc	0	u(8)
seq_parameter_set_id	0	ue(v)
bit_depth_luma_minus8	0	ue(v)
bit_depth_chroma_minus8	0	ue(v)
increased_bit_depth_luma	0	ue(v)
ine_bit_depth_chroma	0	ue(v)
log2_max_frame_num_minus4	0	ue(v)

log2_max_pic_order_cnt_lsb_minus4	0	ue(v)
max_num_ref_frames	0	ue(v)
gaps_in_frame_num_value_allowed_flag	0	u(1)
log2_min_coding_unit_size_minus3	0	ue(v)
max_coding_unit_hierarchy_depth	0	ue(v)
log2_min_transform_unit_size_minus2	0	ue(v)
max_transform_unit_hierarchy_depth	0	ue(v)
pic_width_in_luma_samples	0	u(16)
pic_height_in_luma_samples	0	u(16)
flexible_picture_partitioning_mode	**1**	**ue(v)**
If (!flexible_picture_partitioning_mode){		
If (flexible_picture_partitioning_mode == corner mode) {		
corner_index_horizontal_	**1**	**u(1)**
corner_index_vertical_	**1**	**u(1)**
}		
else if (flexible_picture_partitioning_mode == explicit mode) {		
{		
offset_coordinate_horizontal	**1**	**ue(v)**
offset_coordinate_vertical	**1**	**ue(v)**
}		
}		
numExtraFilters	0	ue(v)
for(i=0; i< numExtraFilters; i++){		
log2_filterCoeffPrecision	0	ue(v)
halfNumTap	0	ue(v)
for(j=0;j<(3*halfNumTap); j++){		
filterCoef[i][j]	0	i(v)
}		
}		
rbsp_trailing_bits( )	0	
}		

[0058] TABLE II shows a syntax change, highlighted in boldface, which may be implemented in an HEVC header at the picture layer.

[0059] TABLEII - Syntax change at picture layer.

pic_parameter_set_rbsp( ) {	C	Descriptor
pic_parameter_set_id	1	ue(v)
seq_parameter_set_id	1	ue(v)
entropy_coding_mode_flag	1	u(1)
num_ref_idx_l0_default_active_minus1	1	ue(v)
num_ref_idx_l1_default_active_minus1	1	ue(v)
pic_init_qp_minus26 /* relative to 26 */	1	se(v)
constrained_intra_pred_flag	1	u(1)
flexible_picture_partitioning_mode	**1**	**ue(v)**
If (!flexible_picture_partitioning_mode){		
If (flexible_picture_partitioning_mode == corner mode) {		
corner_index_horizontal_	**1**	**u(1)**
corner_index_vertical_	**1**	**u(1)**
}		
else if (flexible_picture_partitioning_mode == explicit mode) {		
{		
offset_coordinate_horizontal	**1**	**ue(v)**
offset_coordinate_vertical	**1**	**ue(v)**
}		
}		
for(i=0;i<15; i++){		
numAllowedFilters[i]	1	ue(v)
for(j=0;j<numAllowedFilters;j++){		
filtIdx[i][j]	1	ue(v)
}		
}		
rbsp_trailing_bits( )	1	
}		

[0060] TABLE III shows a syntax change, highlighted in boldface, which may be implemented in an HEVC header at the slice layer.

[0061] TABLE III - Syntax change at slice layer.

slice_header( ) {	C	Descriptor
first_lctb_in_slice	2	ue(v)
slice_type	2	ue(v)
pic_parameter_set_id	2	ue(v)
frame_num	2	u(v)
if(IdrPicFlag)		
idr_pic_id	2	ue(v)
pic_order_cnt_lsb	2	u(v)
if(slice_type == P || slice_type == B) {		
num_ref_idx_active_override_flag	2	u(1)
if(num_ref_idx_active_override_flag) {		
num_ref_idx_l0_active_minus1	2	ue(v)
if(slice_type == B)		
num_ref_idx_l1_active_minus1	2	ue(v)
}		
}		
flexible_partitioning_mode	**1**	**ue(v)**
If (!flexible_partitioning_mode){		
If (flexible_partitioning_mode == corner mode) {		
corner_index_horizontal_	**1**	**u(1)**
corner_index_vertical_	**1**	**u(1)**
}		
else if (flexible_partitioning_mode == explicit mode) {		
{		
offset_coordinate_horizontal	**1**	**ue(v)**
offset_coordinate_vertical	**1**	**ue(v)**
}		
}		

if(nal_ref_idc != 0)		
dec_ref_pic_marking( )	2	
if(entropy_coding_mode_flag && slice_type != I)		
cabac_init_idc	2	ue(v)
slice_qp_delta	2	se(v)
alf_param()		
if(slice_type = = P || slice_type = = B) {		
mc_interpolation_idc	2	ue(v)
mv_competition_flag	2	u(1)
if (mv_competition_flag) {		
mv_competition_temporal_flag	2	u(1)
}		
}		
if (slice_type = = B && mv_competition_flag)		
collocated_from_l0_flag	2	u(1)
sifo_param()		
if (entropy_coding_mode_flag == 3)		
parallel_v2v_header()	2	
edge_based_prediction_flag	2	u(1)
if(edge_prediction_ipd_flag = = 1)		
threshold_edge	2	u(8)
}		

[0062] Semantics which may be utilized with the syntax changes shown in TABLES I – III include “flexible_partitioning_mode” which specifies the flexible partitioning mode of the sequence, picture and/or slice. If the default mode “(flexible_partitioning_mode equals 0)” is used as a comparative example, the top-left corner of picture aligns with the coordinate origin (0,0). If the corner mode “(flexible_partitioning_mode equals 1)” is used, a picture is shifted at the top-left corner away from coordinate (0,0) by a fixed distance determined by the “corner_index_horizontal” and the “corner_index_vertical”.

[0063] If the “corner_index_horizontal” is equal to 0, the top-left corner of picture is shifted from coordinate (0,0) by 0 pixels horizontally. If the “corner_index_horizontal” is equal to 1, the top-left corner of picture is shifted

from coordinate (0,0) by "pic_width_in_luma_samples" – "MaxCodingUnitSize *[pic_width_in_luma_samples/ MaxCodingUnitSize] pixels horizontally".

[0064] If "corner_index_vertical" is equal to 0, the top-left corner of picture is shifted from coordinate (0,0) by 0 pixel vertically. If "corner_index_vertical" is equal to 1, the top-left corner of picture is shifted from coordinate (0,0) by = "pic_height_in_luma_samples" – "MaxCodingUnitSize* (pic_height_in_ luma_samples /MaxCodingUnitSize)" pixels vertically.

[0065] When explicit mode "(flexible_partitioning_mode equals 2)" is used, the picture is shifted with the top-left corner away from the coordinate origin (0,0) by an amount indicated by "offset_coordinate_horizontal" horizontally and "offset_coordinate_vertical vertically". The "offset_coordinate_horizontal" specifies a number of pixels for the top-left corner of picture to be shifted from coordinate (0,0) in the horizontal direction. The "offset_coordinate_vertical" specifies a number of pixels for the top-left corner of picture to be shifted from coordinate (0,0) in vertical direction.

[0066] According to different examples, the coding system 110 may be incorporated or otherwise associated with a transcoder or an encoding apparatus at a headend and the decoding system 140 may be incorporated or otherwise associated with a downstream device, such as a mobile device, a set top box or a transcoder. These may be utilized separately or together in methods of coding and/or decoding utilizing picture boundary variability in preparing coding units. Various manners in which the coding system 110 and the decoding system 140 may be implemented are described in greater detail below with respect to FIGs 5, 6 and 7, which depict flow diagrams of methods 500, 600 and 700.

[0067] Method 500 is a method for preparing coding units utilizing picture boundary variability. Method 600 is a method for coding utilizing coding units prepared utilizing picture boundary variability. Method 700 is a method for decoding utilizing compression data generated utilizing picture boundary

variability. It is apparent to those of ordinary skill in the art that the methods 500, 600 and 700 represent generalized illustrations and that other steps may be added and existing steps may be removed, modified or rearranged without departing from the scope of the methods 500, 600 and 700. The descriptions of the methods 500, 600 and 700 are made with particular reference to the coding system 110 and the decoding system 140 depicted in FIG 1. It should, however, be understood that the methods 500, 600 and 700 may be implemented in systems and/or devices which differ from the coding system 110 and the decoding system 140 without departing from the scope of the methods 500, 600 and 700.

[0068] With reference to the method 500 in FIG 5, at step 501, the controller 111 associated with the coding system 110 calculates an efficiency measure associated with a potential source picture position in a coordinate system based on fitting the coordinate system and the source picture with respect to each other.

[0069] At step 502, the controller 111 determines if the calculated efficiency measure meets a minimum efficiency measure associated with a coding efficiency goal.

[0070] At step 503, if the calculated efficiency measure meets the minimum efficiency measure, the controller 111 determines a potential source picture position in the coordinate system based on the calculated efficiency measure, the source picture and the coordinate system.

[0071] At step 504, the controller 111 determines an actual source picture position based on one or more of the determined potential source picture positions and the coding efficiency goal.

[0072] At step 505, the controller 111 divides the at least one source picture into a plurality of LCTUs based on the coordinate system and the determined actual source picture position.

[0073] At step 506, the controller 111 partitions the LCTUs into at least one coding unit based on the tree format and a homogeneity rule associated with the pixels in the coding units.

[0074] With reference to the method 600 in FIG. 6, at step 601, the interface 130 and the frame memory 113 of the coding system 110 receive the source bitstream 120 including source pictures.

[0075] At step 602, the controller 111 prepares coding units based on the received source pictures. The preparing may be performed as described above with respect to method 500.

[0076] At step 603, the controller 111 and the encoding unit 114 process the prepared coding units generating video compression data based on the processed coding units.

[0077] At step 604, the controller 111 and the encoding unit 114 package the generated video compression data.

[0078] At step 605, the controller 111 and the transmitter buffer 115 transmit the packaged video compression data in compressed bitstream 105 via the interface 135.

[0079] With reference to the method 700 in FIG. 7, at step 701, the decoding system 140 receives the compressed bitstream 105 including the video compression data via the interface 170 and the receiver buffer 150.

[0080] At step 702, the decoding system 140 receives residual pictures associated with the video compression data via the interface 170 and the receiver buffer 150.

[0081] At step 703, the decoding unit 151 and the controller 153 process the received video compression data.

[0082] At step 704, the decoding unit 151 and the controller 153 generate reconstructed pictures based on the processed video compression data and the

received residual pictures.

[0083] At step 705, the decoding unit 151 and the controller 153 package the generated reconstructed pictures and signal them to the frame memory 152.

[0084] At step 706, the controller 153 signals the generated reconstructed pictures in the decoded signal 180 via the interface 175.

[0085] Some or all of the methods and operations described above may be provided as machine readable instructions, such as a utility, a computer program, etc., stored on a computer readable storage medium, which may be non-transitory such as hardware storage devices or other types of storage devices. For example, they may exist as program(s) comprised of program instructions in source code, object code, executable code or other formats.

[0086] An example of a computer readable storage media includes a conventional computer system RAM, ROM, EPROM, EEPROM, and magnetic or optical disks or tapes. Concrete examples of the foregoing include distribution of the programs on a CD ROM. It is therefore to be understood that any electronic device capable of executing the above-described functions may perform those functions enumerated above.

[0087] Referring to FIG. 8, there is shown a platform 800, which may be employed as a computing device in a system for coding or decoding utilizing picture boundary variability, such as coding system 100 and/or decoding system 200. The platform 800 may also be used for an upstream encoding apparatus, a transcoder, or a downstream device such as a set top box, a handset, a mobile phone or other mobile device, a transcoder and other devices and apparatuses which may utilize picture boundary variability and associated coding units prepared based on picture boundary variability. It is understood that the illustration of the platform 800 is a generalized illustration and that the platform 800 may include additional components and that some of the components described may be removed and/or modified without departing from a scope of the

platform 800.

[0088] The platform 800 includes processor(s) 801, such as a central processing unit; a display 802, such as a monitor; an interface 803, such as a simple input interface and/or a network interface to a Local Area Network (LAN), a wireless 802.11x LAN, a 3G or 4G mobile WAN or a WiMax WAN; and a computer-readable medium 804. Each of these components may be operatively coupled to a bus 808. For example, the bus 808 may be an EISA, a PCI, a USB, a FireWire, a NuBus, or a PDS.

[0089] A computer readable medium (CRM), such as CRM 804 may be any suitable medium which participates in providing instructions to the processor(s) 801 for execution. For example, the CRM 804 may be non-volatile media, such as an optical or a magnetic disk; volatile media, such as memory; and transmission media, such as coaxial cables, copper wire, and fiber optics. Transmission media can also take the form of acoustic, light, or radio frequency waves. The CRM 804 may also store other instructions or instruction sets, including word processors, browsers, email, instant messaging, media players, and telephony code.

[0090] The CRM 804 may also store an operating system 805, such as MAC OS, MS WINDOWS, UNIX, or LINUX; applications 806, network applications, word processors, spreadsheet applications, browsers, email, instant messaging, media players such as games or mobile applications (e.g., “apps”); and a data structure managing application 807. The operating system 805 may be multi-user, multiprocessing, multitasking, multithreading, real-time and the like. The operating system 805 may also perform basic tasks such as recognizing input from the interface 803, including from input devices, such as a keyboard or a keypad; sending output to the display 802 and keeping track of files and directories on CRM 804; controlling peripheral devices, such as disk drives, printers, image capture devices; and managing traffic on the bus 808. The applications 806 may include various components for establishing and

maintaining network connections, such as code or instructions for implementing communication protocols including TCP/IP, HTTP, Ethernet, USB, and FireWire.

[0091] A data structure managing application, such as data structure managing application 807 provides various code components for building/updating a computer readable system (CRS) architecture, for a non-volatile memory, as described above. In certain examples, some or all of the processes performed by the data structure managing application 807 may be integrated into the operating system 805. In certain examples, the processes may be at least partially implemented in digital electronic circuitry, in computer hardware, firmware, code, instruction sets, or any combination thereof.

[0092] According to principles of the invention, there are systems, methods, and computer readable mediums (CRMs) which provide for coding and decoding utilizing picture boundary variability in preparing coding units. By utilizing picture boundary variability, LCTUs may be located freely with respect to a picture so that their associated coding units may be partitioned so as to increase the coding efficiencies associated with processing overhead and/or bandwidth required by the systems, methods, and CRMs for coding and decoding utilizing picture boundary variability.

[0093] Although described specifically throughout the entirety of the instant disclosure, representative examples have utility over a wide range of applications, and the above discussion is not intended and should not be construed to be limiting. The terms, descriptions and figures used herein are set forth by way of illustration only and are not meant as limitations. Those skilled in the art recognize that many variations are possible within the spirit and scope of the examples. While the examples have been described with reference to examples, those skilled in the art are able to make various modifications to the described examples without departing from the scope of the examples as described in the following claims, and their equivalents.

What is claimed is:

1. A system for coding, the system comprising

a processor configured to

prepare coding units based on source pictures,

the preparing including

calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other,

wherein the coordinate system includes two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin,

determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal, and

dividing the source picture into a plurality of largest coding tree units based on the determined source picture position.

2. The system of claim 1, wherein

the source picture is in the shape of a polygon having four corners,

the coordinate system and the determined source picture position are both located in one plane,

a plurality of equivalently spaced axially perpendicular lines are located along the two axes of the coordinate system, and

a plurality of pairs of perpendicular lines in the plurality of lines, have line intersections coinciding with corners of the largest coding tree units in the plurality of largest coding tree units.

3. The system of claim 2, wherein

the source picture is in the shape of a rectangle,

the complete determined source picture position is located within a single quadrant of the coordinate system, and

a locus of the determined source picture position, coinciding at a corner of a source picture corner largest coding tree unit in the plurality of largest coding tree units, coincides with the origin of the coordinate system.

4. The system of claim 2, wherein

the source picture is in the shape of a rectangle,

the complete determined source picture position is located within a single quadrant of the coordinate system,

a first locus of the determined source picture position,

coinciding with a nearest to the origin corner of a first source picture corner largest coding tree unit located nearest to the origin of the plurality of largest coding tree units,

is separated from the origin of the coordinate system by an offset distance, and

a second locus of the determined source picture position,

coinciding with a farthest from the origin corner of a second source picture corner largest coding tree unit of the plurality of largest coding tree units,

coincides with a line intersection of a pair of the plurality of pairs of perpendicular lines in the plurality of lines having line intersections coinciding with corners of the largest coding tree units in the plurality of largest coding tree units.

5. The system of claim 2, wherein

the source picture is in the shape of a rectangle,

the complete determined source picture position is located within a single quadrant of the coordinate system,

a first locus of the determined source picture position,

coinciding with a nearest to the origin corner of a nearest to the origin source picture corner largest coding tree unit in the plurality of largest coding tree units,

is separated from the origin of the coordinate system by an offset distance, and

a second locus of the determined source picture position,

coinciding with a farthest from the origin corner of a farthest from the origin source picture corner largest coding tree unit in the plurality of largest coding tree units,

is separated by an offset distance from a line intersection of a pair of the plurality of pairs of perpendicular lines in the plurality of lines having line intersections coinciding with corners of the largest coding tree units in the plurality of largest coding tree units.

6. The system of claim 5, wherein the offset distance associated with the second locus is equivalent to a number of pixels.

7. The system of claim 1, wherein the coding efficiency goal is at least one of

a predetermined homogeneity measure of the prepared coding units associated with at least one aspect of at least one feature of the source picture, and

a predetermined maximum number of prepared coding units based on the source picture.

8. The system of claim 1, wherein the determined source picture position includes an angular orientation of a side of the polygon of the source picture with respect to an axis of the two axes.

9. A method for coding, the method comprising

preparing coding units based on source pictures utilizing a processor;

the preparing including

calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other,

wherein the coordinate system includes two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin,

determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal, and

dividing the source picture into a plurality of largest coding tree units based on the determined source picture position.

10. A non-transitory computer readable medium storing computer readable instructions that when executed by a computer system perform a method for coding, the method comprising:

preparing coding units based on source pictures utilizing a processor;

the preparing including

calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other,

wherein the coordinate system includes two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin,

determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal, and

dividing the source picture into a plurality of largest coding tree units based on the determined source picture position.

11. A system for decoding, the system comprising:

an interface configured to receive video compression data; and

a processor configured to process the received video compression data,

wherein the received video compression data is based on coding units, based on source pictures, and the coding units are prepared by steps including

calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other,

wherein the coordinate system includes two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin,

determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal, and

dividing the source picture into a plurality of largest coding tree units based on the determined source picture position.

12. The system of claim 11, wherein

the source picture is in the shape of a polygon having four corners,

the coordinate system and the determined source picture position are both located in one plane,

a plurality of equivalently spaced axially perpendicular lines are located along the two axes of the coordinate system, and

a plurality of pairs of perpendicular lines in the plurality of lines, have line intersections coinciding with corners of the largest coding tree units in the plurality of largest coding tree units.

13. The system of claim 12, wherein

the source picture is in the shape of a rectangle,

the complete determined source picture position is located within a single quadrant of the coordinate system, and

a locus of the determined source picture position, coinciding at a corner of a source picture corner largest coding tree unit in the plurality of largest coding tree units, coincides with the origin of the coordinate system.

14. The system of claim 12, wherein

the source picture is in the shape of a rectangle,

the complete determined source picture position is located within a single quadrant of the coordinate system,

a first locus of the determined source picture position,

coinciding with a nearest to the origin corner of a first source picture corner largest coding tree unit located nearest to the origin of the plurality of largest coding tree units, and

separated from the origin of the coordinate system by an offset distance, and

a second locus of the determined source picture position,

coinciding with a farthest from the origin corner of a second source picture corner largest coding tree unit of the plurality of largest coding tree units, and

coinciding with a line intersection of a pair of the plurality of pairs of perpendicular lines in the plurality of lines having line intersections coinciding with corners of the largest coding tree units in the plurality of largest coding tree units.

15. The system of claim 12, wherein

the source picture is in the shape of a rectangle,

the complete determined source picture position is located within a single quadrant of the coordinate system,

a first locus of the determined source picture position,

coinciding with a nearest to the origin corner of a nearest to the origin source picture corner largest coding tree unit in the plurality of largest coding tree units,

is separated from the origin of the coordinate system by an offset distance, and

a second locus of the determined source picture position,

coinciding with a farthest from the origin corner of a farthest from the origin source picture corner largest coding tree unit in the plurality of largest coding tree units,

is separated by an offset distance from a line intersection of a pair of the plurality of pairs of perpendicular lines in the plurality of lines having line intersections coinciding with corners of the largest coding tree units in the plurality of largest coding tree units.

16. The system of claim 15, wherein the offset distance associated with the second locus is equivalent to a number of pixels.

17. The system of claim 11, wherein the coding efficiency goal is at least one of

a predetermined homogeneity measure of the prepared coding units associated with at least one aspect of at least one feature of the source picture, and

a predetermined maximum number of prepared coding units based on the source picture.

18. The system of claim 11, wherein the determined source picture position includes an angular orientation of a side of the polygon of the source picture with respect to an axis of the two axes.

19. A method for decoding, the method comprising:

receiving video compression data; and

processing the received video compression data utilizing a processor,

wherein the received video compression data is based on coding units, based on source pictures, and the coding units are prepared by steps including

calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other,

wherein the coordinate system includes two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin,

determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal,

dividing the source picture into a plurality of largest coding tree units based on the determined source picture position, and

partitioning largest coding tree units of the plurality of largest cording tree units to form the prepared coding units.

20. A non-transitory computer readable medium storing computer readable instructions that when executed by a computer system perform a method for decoding, the method comprising:

receiving video compression data; and

processing the received video compression data utilizing a processor,

wherein the received video compression data is based on coding units, based on source pictures, and the coding units are prepared by steps including

calculating an efficiency measure associated with at least one potential source picture position in a coordinate system based on fitting the coordinate system and at least one source picture with respect to each other,

wherein the coordinate system includes two perpendicular axes in a plane intersecting at an origin of the coordinate system and dividing the plane into four quadrants meeting at the origin,

determining a source picture position for the source picture in the coordinate system based on the calculated efficiency measure, the potential source picture position and a coding efficiency goal,

dividing the source picture into a plurality of largest coding tree units based on the determined source picture position, and

partitioning largest coding tree units of the plurality of largest cording tree units to form the prepared coding units.

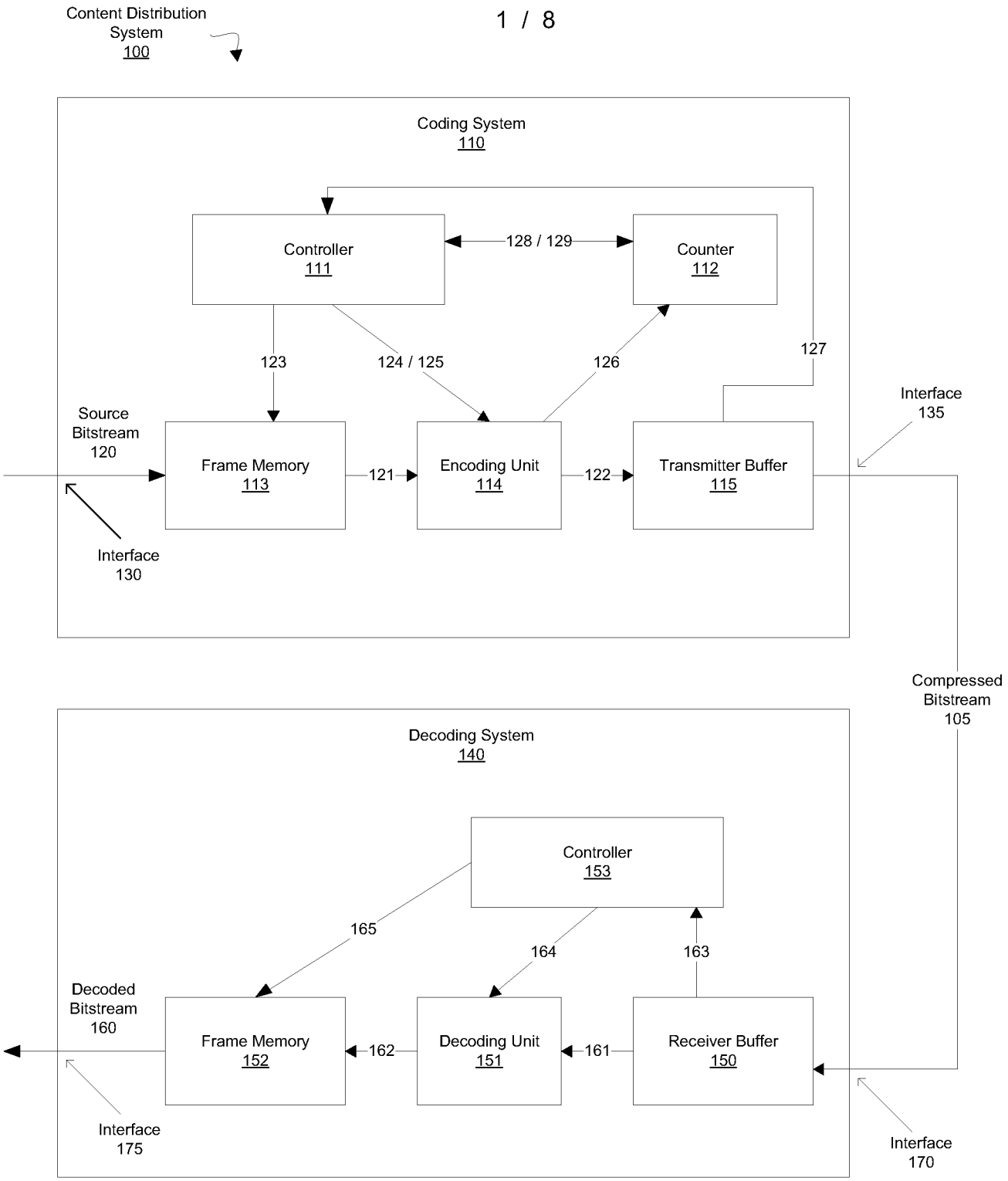

Content Distribution System 100
Coding System 110
Controller 111
128 / 129
Counter 112
123
124 / 125
126
127
Source Bitstream 120
Frame Memory 113
121
Encoding Unit 114
122
Transmitter Buffer 115
Interface 135
Interface 130
Compressed Bitstream 105
Decoding System 140
Controller 153
165
164
163
Decoded Bitstream 160
Frame Memory 152
162
Decoding Unit 151
161
Receiver Buffer 150
Interface 175
Interface 170

FIG. 1

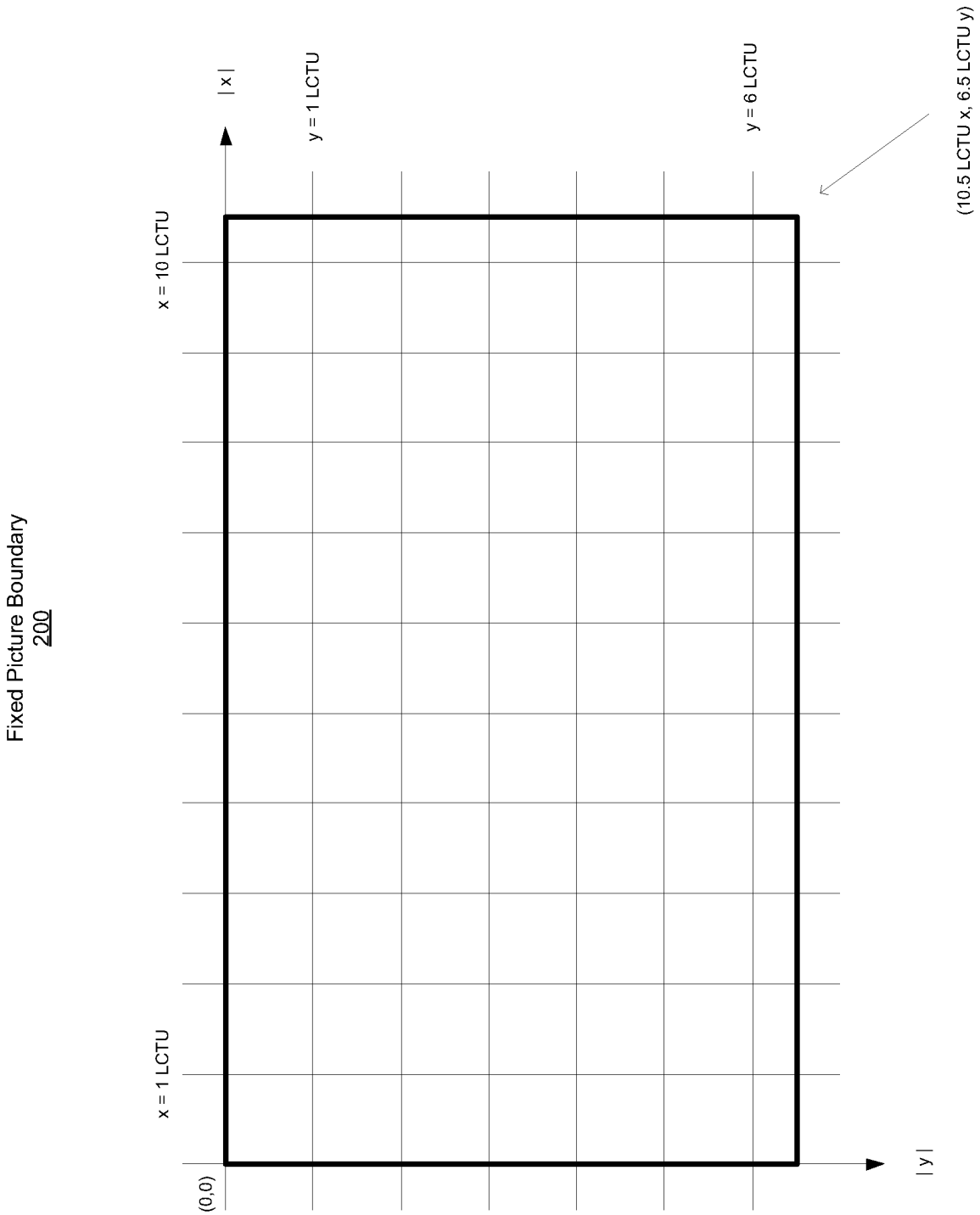

Fixed Picture Boundary
200
| x |
y = 1 LCTU
y = 6 LCTU
(10.5 LCTU x, 6.5 LCTU y)
x = 10 LCTU
x = 1 LCTU
(0,0)
| y |

FIG. 2

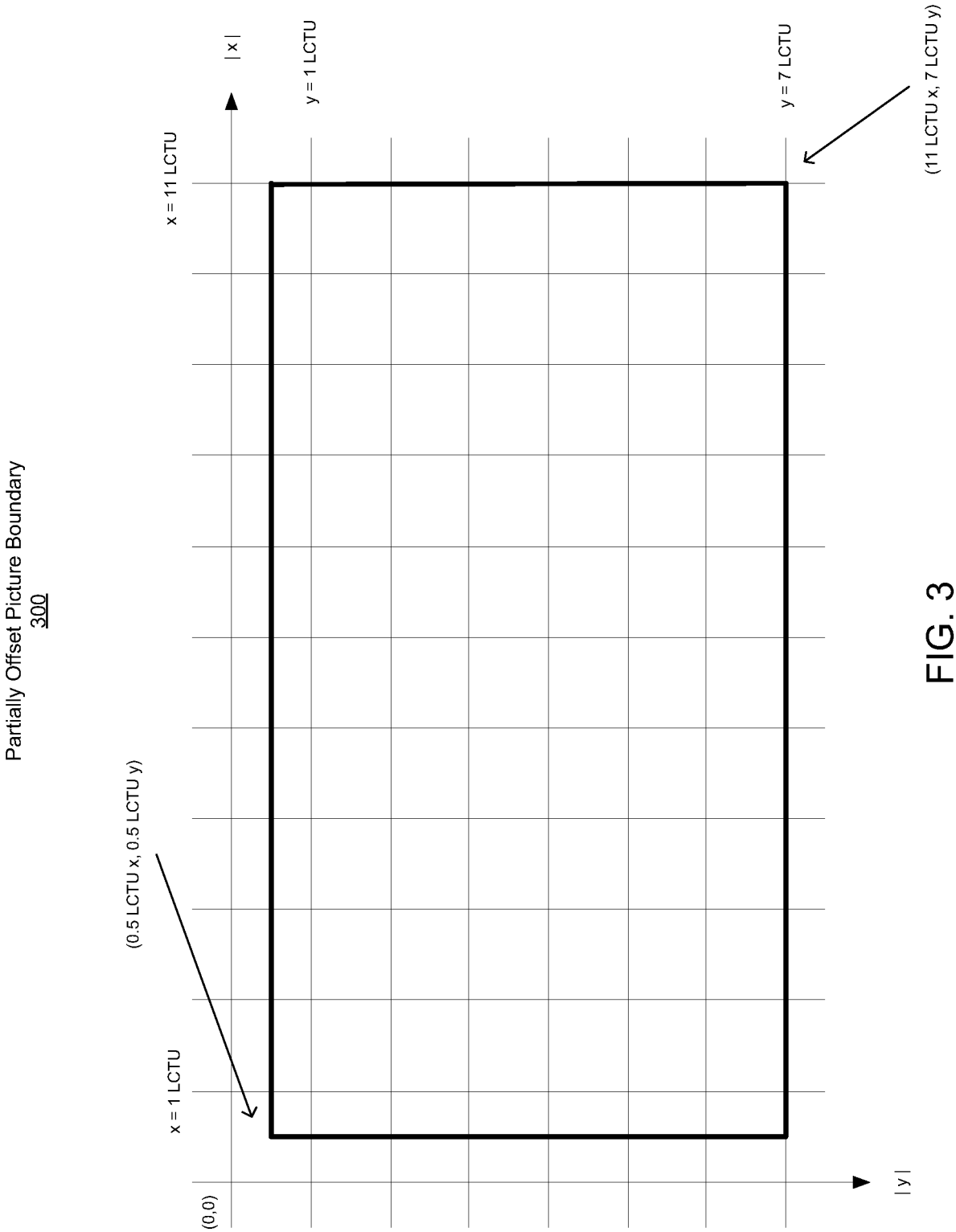

Partially Offset Picture Boundary
300
| x |
y = 1 LCTU
y = 7 LCTU
(11 LCTU x, 7 LCTU y)
x = 11 LCTU
(0.5 LCTU x, 0.5 LCTU y)
x = 1 LCTU
(0,0)
| y |

FIG. 3

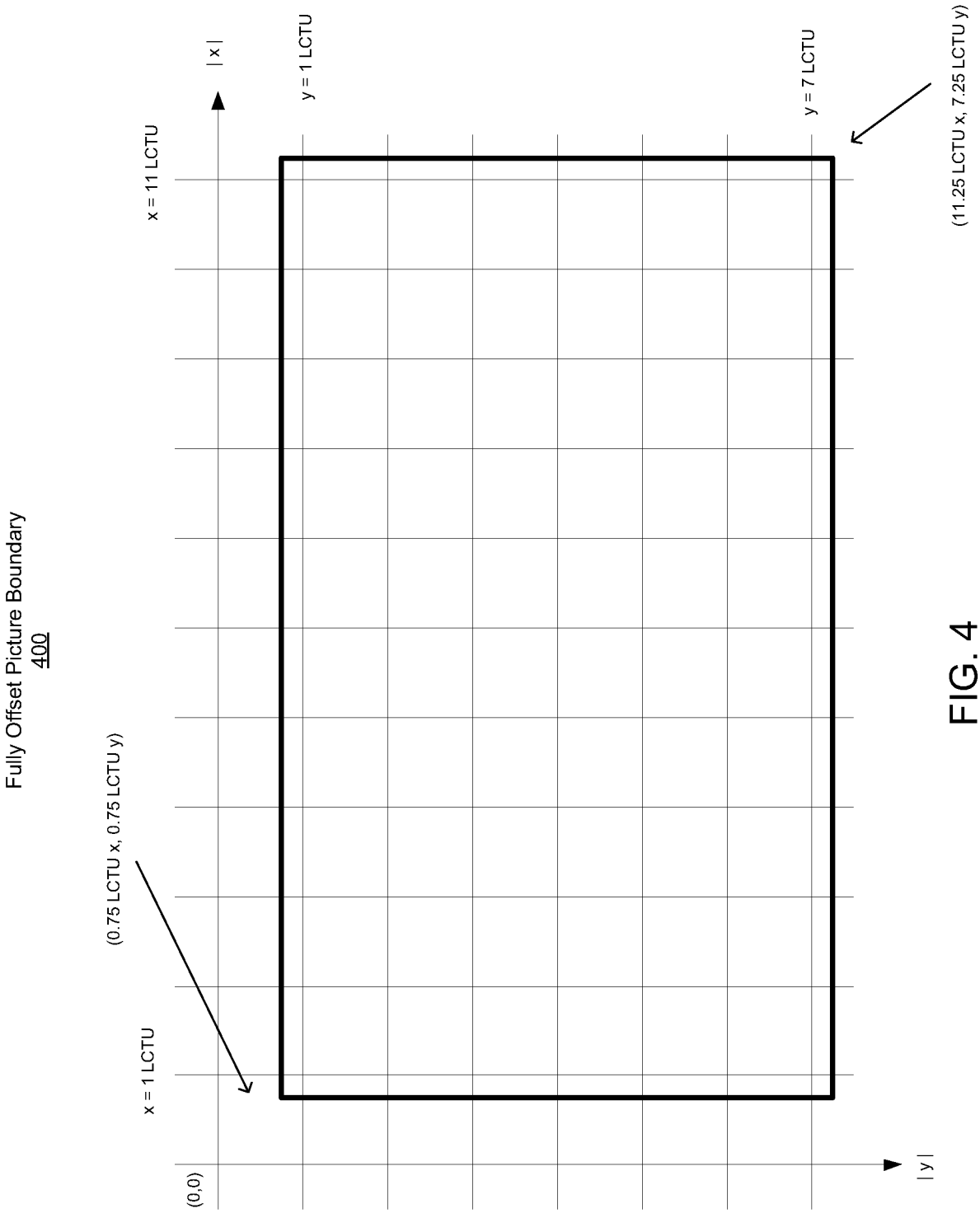

Fully Offset Picture Boundary
400
| x |
y = 1 LCTU
y = 7 LCTU
(11.25 LCTU x, 7.25 LCTU y)
x = 11 LCTU
(0.75 LCTU x, 0.75 LCTU y)
x = 1 LCTU
(0,0)
| y |

FIG. 4

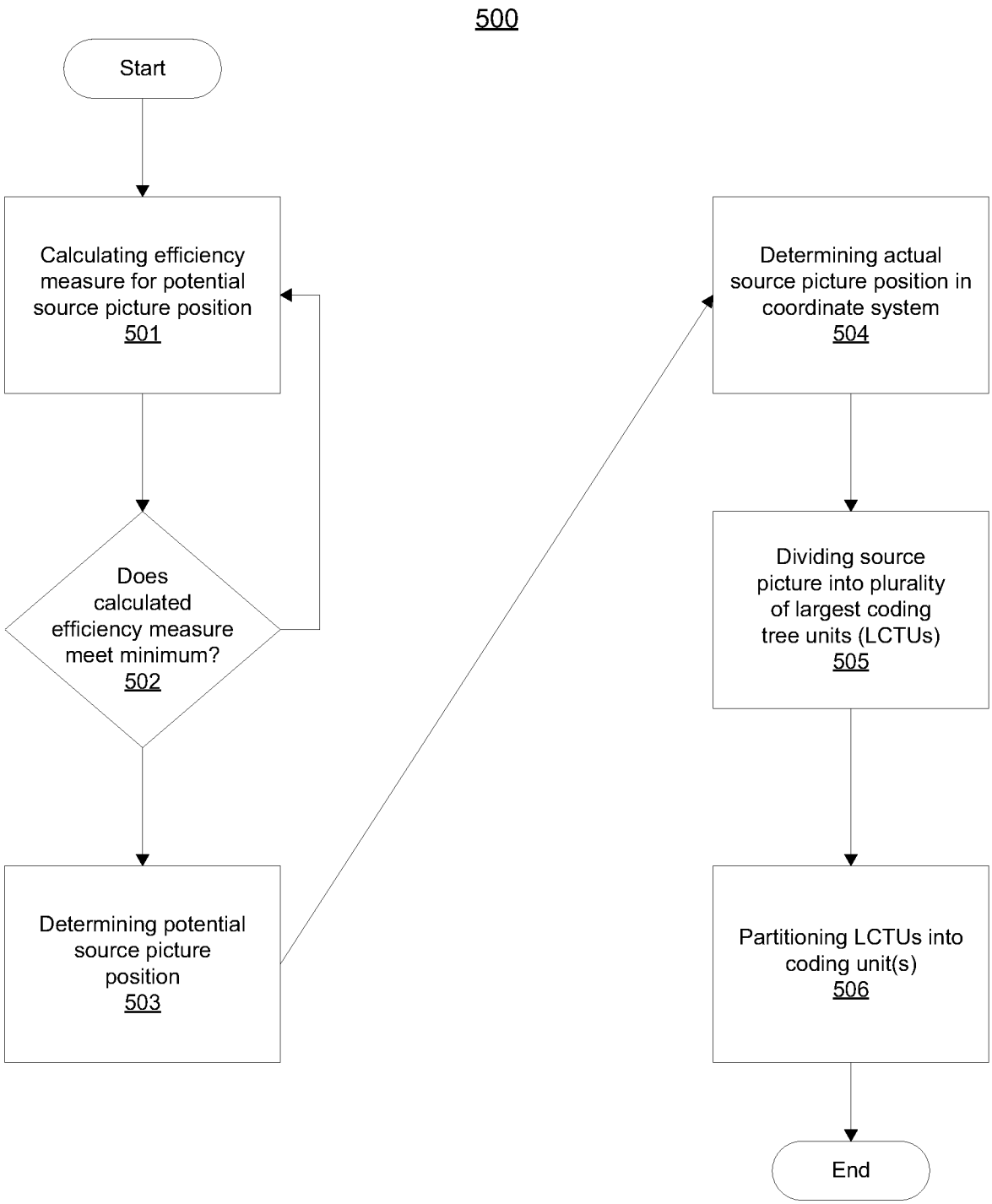

500
Start
Calculating efficiency measure for potential source picture position 501
Does calculated efficiency measure meet minimum? 502
Determining potential source picture position 503
Determining actual source picture position in coordinate system 504
Dividing source picture into plurality of largest coding tree units (LCTUs) 505
Partitioning LCTUs into coding unit(s) 506
End

FIG. 5

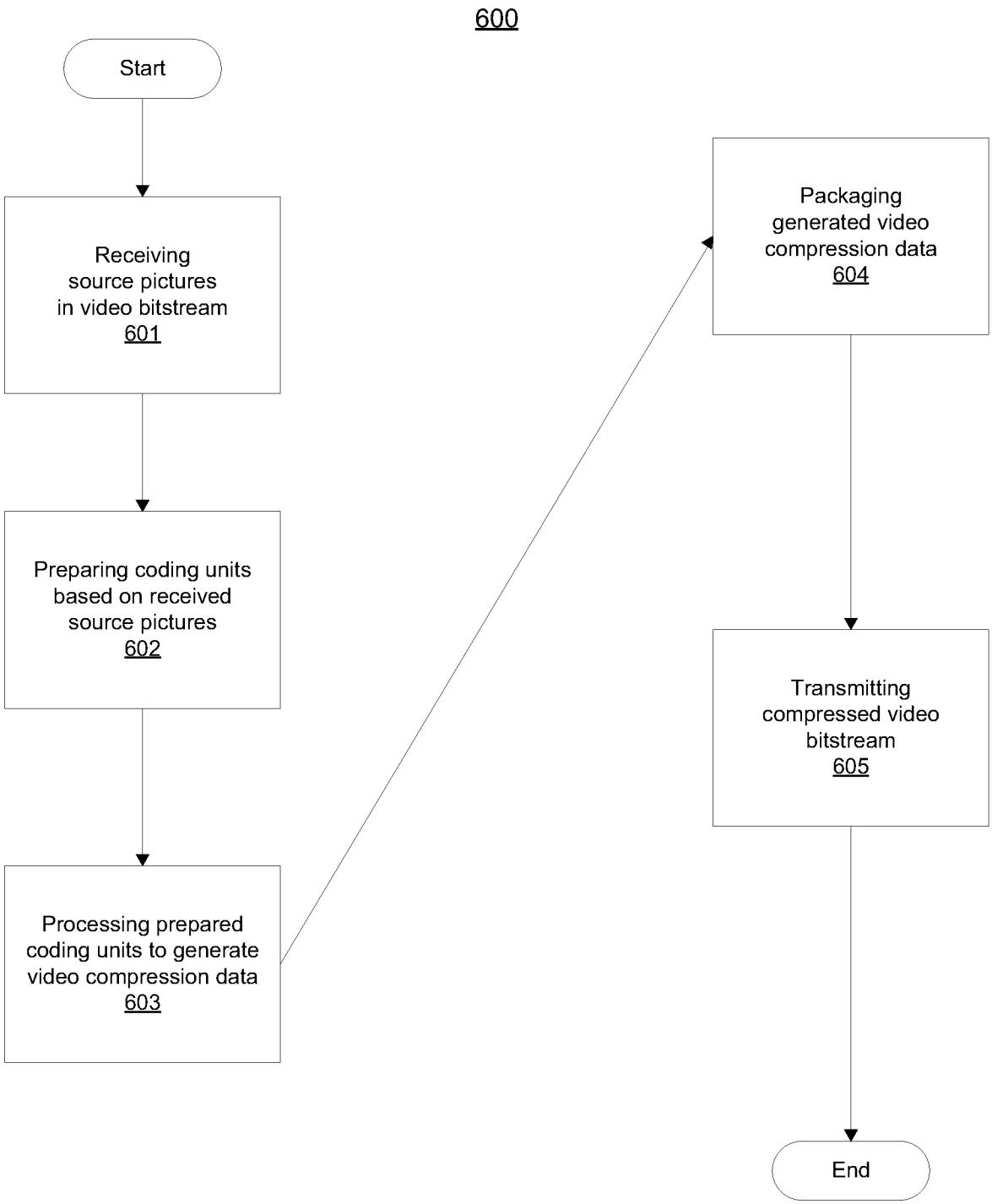

600
Start
Receiving source pictures in video bitstream 601
Preparing coding units based on received source pictures 602
Processing prepared coding units to generate video compression data 603
Packaging generated video compression data 604
Transmitting compressed video bitstream 605
End

FIG. 6

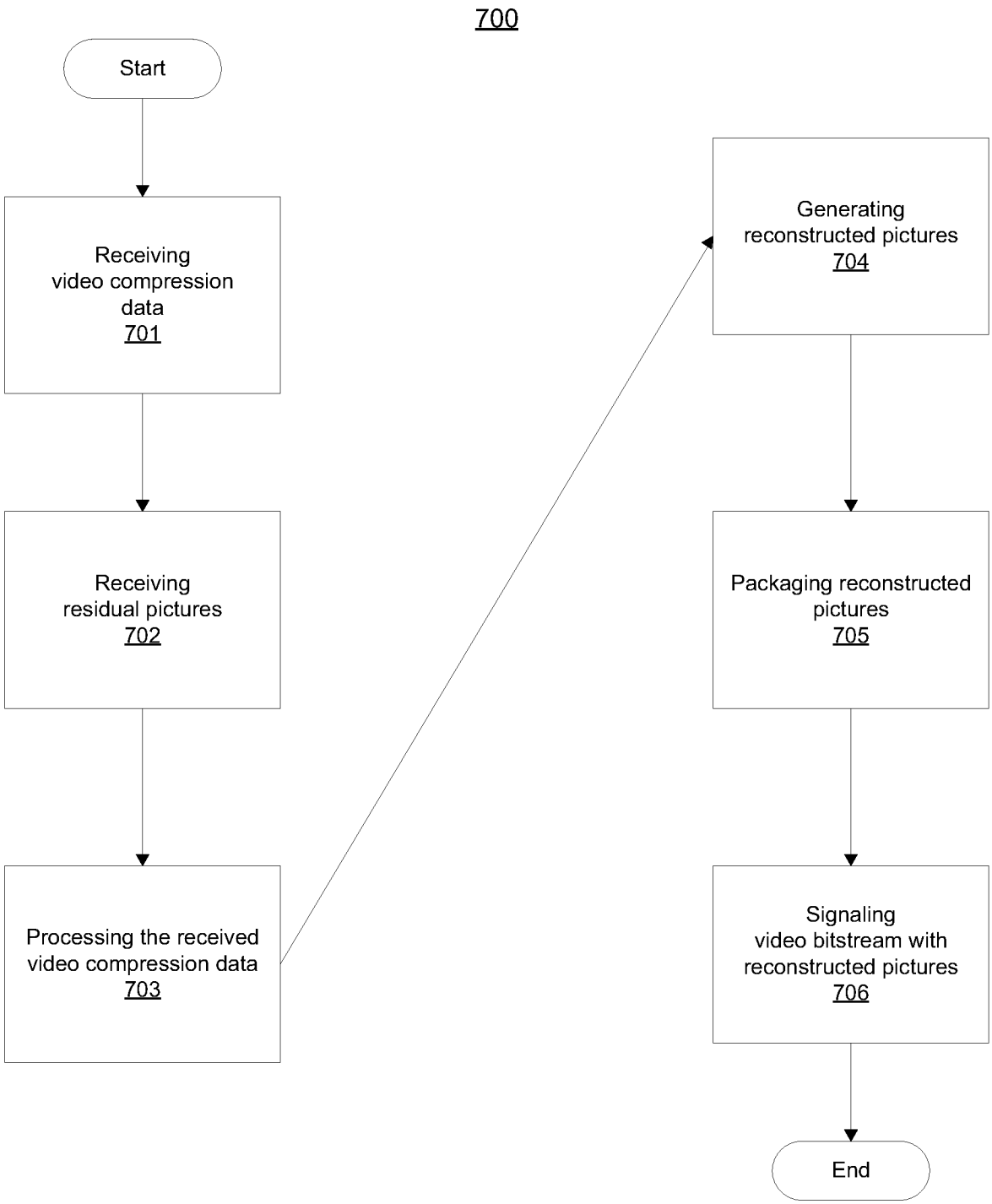

700
Start
Receiving video compression data 701
Receiving residual pictures 702
Processing the received video compression data 703
Generating reconstructed pictures 704
Packaging reconstructed pictures 705
Signaling video bitstream with reconstructed pictures 706
End

FIG. 7

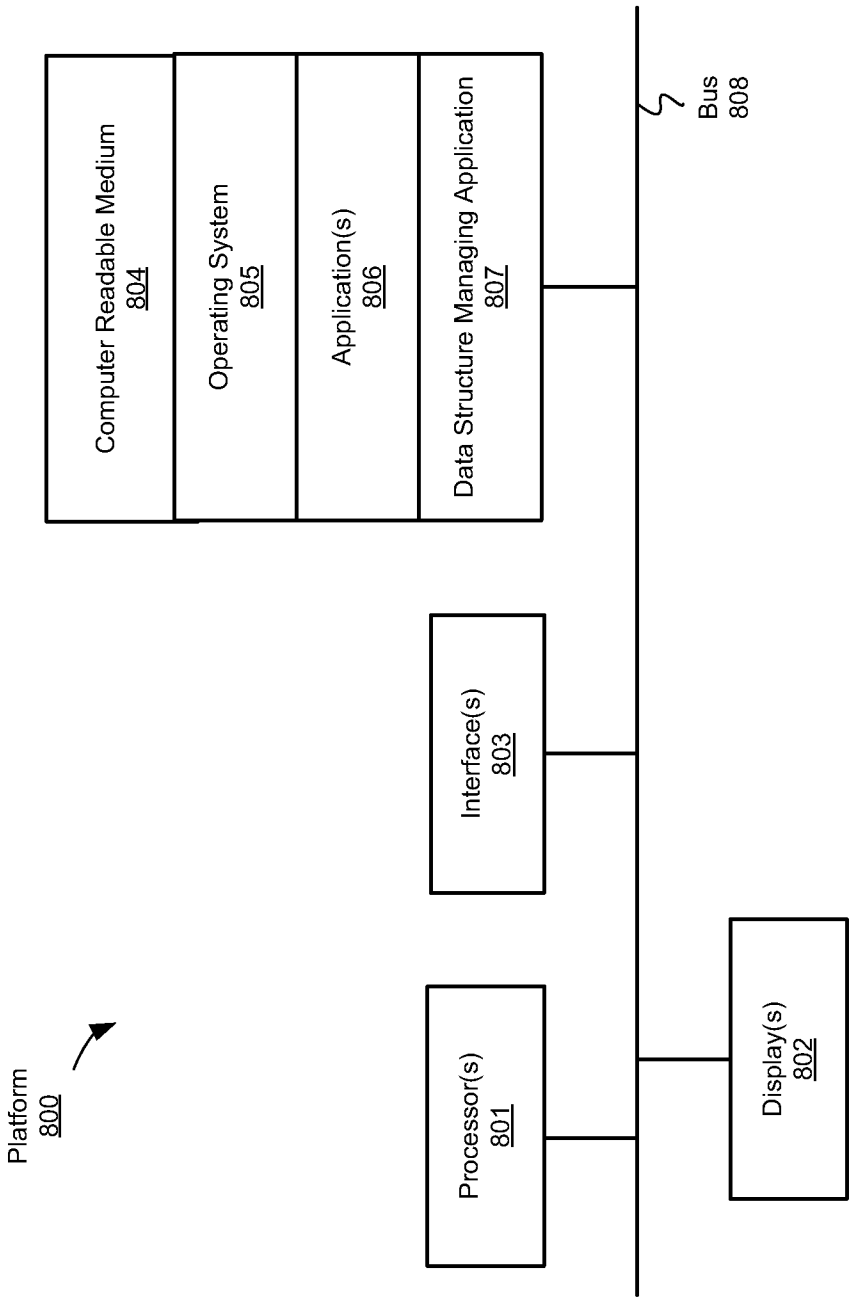

Platform
800
Processor(s)
801
Interface(s)
803
Computer Readable Medium
804
Operating System
805
Application(s)
806
Data Structure Managing Application
807
Bus
808
Display(s)
802

FIG. 8

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2011/053761

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N7/26
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 practical, search terms used)
EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2009/086761 A1 (HUAWEI TECH CO LTD [CN]; YANG HAITAO [CN]; LIN SIXIN [CN]; GAO SHAN [C) 16 July 2009 (2009-07-16)	1-7, 9-17,19, 20
Y	the whole document & US 2010/266048 A1 (YANG HAITAO [CN] ET AL) 21 October 2010 (2010-10-21) figures 3,4 paragraph [0059] - paragraph [0064]	8,18

-/--

[X] Further documents are listed in the continuation of Box C.

[X] 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 document but published on or after the international filing date

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
18 November 2011	29/11/2011

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

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2011/053761

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	GUO L ET AL: "TE3: Simplified geometry block partitioning", 2. JCT-VC MEETING; 21-7-2010 - 28-7-2010; GENEVA; (JOINT COLLABORATIVETEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL:HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-B085, 23 July 2010 (2010-07-23), XP030007665, ISSN: 0000-0046	8,18
A	figure 1 point 1 "Introduction"	1-7, 9-17,19, 20

X,P	PANUSOPONE K ET AL: "Flexible picture partitioning", 3. JCT-VC MEETING; 94. MPEG MEETING; 7-10-2010 - 15-10-2010;GUANGZHOU; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IECJTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-C260, 2 October 2010 (2010-10-02), XP030007967, ISSN: 0000-0019 the whole document	1-20

A	PAPATHANASSIADIS T: "Image block partitioning: a compression technique suitable for parallel processing", IMAGE PROCESSING AND ITS APPLICATIONS, 1992., INTERNATIONAL CONFERENCE ON MAASTRICHT, NETHERLANDS, LONDON, UK,IEE, UK, 1 January 1992 (1992-01-01), pages 105-108, XP006500131, ISBN: 978-0-85296-543-6 the whole document	1-20

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2011/053761

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2009086761 A1	16-07-2009	CN 101478672 A	08-07-2009
		US 2010266048 A1	21-10-2010
		WO 2009086761 A1	16-07-2009