



(43) International Publication Date
04 October 2018 (04.10.2018)

(51) International Patent Classification:

H04N 19/176 (2014.01) H04N 19/119 (2014.01)
H04N 19/70 (2014.01) H04N 19/167 (2014.01)

avenue des Champs Blancs, CS17616, 35576 Cesson-Sévigné Cedex (FR). **LELEANNEC, Fabrice**; c/o Technicolor, 975, avenue des Champs Blancs, CS17616, 35576 Cesson-Sévigné Cedex (FR).

(21) International Application Number:

PCT/EP2018/056254

(74) Agent: **LORETTE, Anne** et al.; c/o Technicolor, 1-5, rue Jeanne d'Arc, 92130 Issy-les-Moulineaux (FR).

(22) International Filing Date:

13 March 2018 (13.03.2018)

(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.

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

17305347.1 27 March 2017 (27.03.2017) EP

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

(72) Inventors: **POIRIER, Tangi**; c/o Technicolor, 975, avenue des Champs Blancs, CS17616, 35576 Cesson-Sévigné Cedex (FR). **GALPIN, Franck**; c/o Technicolor, 975,

(54) Title: METHODS AND APPARATUS FOR PICTURE ENCODING AND DECODING

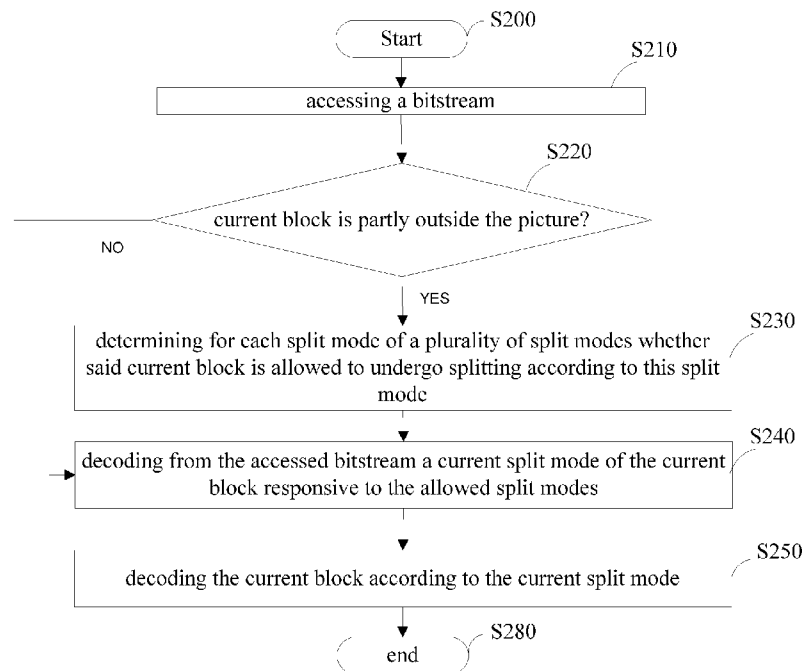


FIGURE 11

(57) Abstract: A decoding method is disclosed that comprises : - determining whether a current block of a picture extends beyond a boundary of the picture; - determining for each split mode of a plurality of split modes whether said current block is allowed to undergo splitting according to said split mode by checking whether at least one of the split line is co-located with one of the picture border or whether the size of the block part inside the picture along the picture boundary is a multiple of a minimum block size; - decoding from a bitstream a current split mode of the current block responsive to the allowed split modes; and - decoding the current block according to the current split mode.



(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).

Declarations under Rule 4.17:

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

Published:

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

METHODS AND APPARATUS FOR PICTURE ENCODING AND DECODING

1. TECHNICAL FIELD

5 The present principles generally relate to a method and an apparatus for picture encoding and decoding, and more particularly, to a method and an apparatus for picture block encoding and decoding at picture boundaries.

2. BACKGROUND ART

10 To achieve high compression efficiency, video coding schemes usually employ prediction and transform to leverage spatial and temporal redundancy in the video content. Generally, intra or inter prediction is used to exploit the intra or inter frame correlation, then the differences between the original image and the predicted image, often denoted as residuals, are transformed, quantized and entropy coded. To
15 reconstruct the video, the compressed data is decoded by inverse processes corresponding to the prediction, transform, quantization and entropy coding.

In HEVC coding ("*ITU-T H.265 Telecommunication standardization sector of ITU (10/2014), series H: audiovisual and multimedia systems, infrastructure of audiovisual services - coding of moving video, High efficiency video coding, Recommendation ITU-T H.265*"), a picture is partitioned into coding tree units (CTU)
20 of square shape with a configurable size typically 64x64, 128x128, or 256x256. A CTU is the root of a quad-tree partitioning into Coding Units (CU). For each CU, a prediction mode is signaled which indicates whether the CU is coded using intra or inter prediction. A Coding Unit is partitioned into one or more Prediction Units (PU) and
25 forms the root of a quad-tree (known as transform tree) partitioning into Transform Units (TUs). A PU may have a square or a rectangular shape while a TU has a square shape. Each PU is assigned some prediction information, for instance motion information, spatial intra prediction, etc.

Quad-Tree plus Binary-Tree (QTBT) coding tool ("*Algorithm Description of
30 Joint Exploration Test Model 3*", Document JVET-C1001_v3, Joint Video Exploration Team of ISO/IEC JTC1/SC29/WG11, 3rd meeting, 26th May-1st June 2015, Geneva, CH) is a new video coding tool that provides a more flexible CTU representation than the CU/PU/TU arrangement of the HEVC standard. This coding tool was introduced in the Joint Exploration Model (JEM) which is the reference software for the Joint Video

Exploration Team of ISO/IEC JTC1/SC29/WG11. The QTBT coding tool defines a coding tree where coding units can be split both in a quad-tree and in a binary-tree fashion. Such coding tree representation of a Coding Tree Unit is illustrated on **Figure 1**, where solid lines indicate quad-tree partitioning and dotted lines indicate binary partitioning of a CU and further on **Figure 2**. On Figure 2, solid lines represent the quad-tree splitting and dotted lines represent the binary splitting that is spatially embedded in the quad-tree leaves. On this figure, a value 1 corresponds to a vertical binary split and a value 0 corresponds to an horizontal binary split.

The splitting of a CTU into coding units is decided on the encoder side, e.g. through a rate distortion optimization procedure which consists in determining the QTBT representation of the CTU with minimal rate distortion cost. In the QTBT representation, a CU has either a square or a rectangular shape. The size of a coding unit is always a power of 2, and typically goes from 4 to 128. The QTBT decomposition of a CTU comprises two stages: the CTU is first split into 4 CUs in a quad-tree fashion, then each quad-tree leaf can be further divided into two CUs in a binary fashion or into 4 CUs in a quad-tree fashion, as illustrated on **Figure 2**.

With the QTBT representation, a CU is not further partitioned into PUs or TUs. In other words, once the partitioning of a CTU is decided, each CU is considered as a single prediction unit and a single transform unit. However, such a QTBT representation only allows for symmetric splitting of a CU as illustrated by **Figure 3**. Figure 3 depicts 4 split modes allowed by QTBT. The mode NO_SPLIT indicates that the CU is not further split. The mode QT_SPLIT indicates that the CU is split into 4 quadrants according to a quad-tree, the quadrants being separated by two split lines. The mode HOR indicates that the CU is split horizontally into two CUs of equal size separated by one split line. VER indicates that the CU is split vertically into two CUs of equal size separated by one split line. The split lines are represented by dashed lines on Figure 3.

In “*Asymmetric Coding Units in QTBT*”, Document: JVET-D0064, Joint Video Exploration Team (JVET) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 4th Meeting: Chengdu, CN, 15–21 October 2016, new asymmetric split modes are introduced in QTBT. A CU may thus be split horizontally into two coding units with respective rectangular sizes $(w, h/4)$ and $(w, 3h/4)$ or vertically into two coding units with respective rectangular sizes $(w/4, h)$ and $(3w/4, h)$ depicted on **Figure 4**. Furthermore, a CU with a size multiple of 3 in width or height can be further split in a

binary fashion horizontally or vertically, provided the size is even. The two coding units are separated by one split line represented by dashed line on Figure 4.

- In “*Multi-Type-Tree*”, Document: JVET-D0117-r1, Joint Video Exploration Team (JVET) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 4th Meeting: Chengdu, CN, 15–21 October 2016, new triple split mode are introduced. A CU may thus be split horizontally into three coding units with respective rectangular sizes $(w, h/4)$, $(w, h/2)$ and $(w, h/4)$ or vertically into three coding units with respective rectangular sizes $(w/4, h)$, $(w/2, h)$ and $(w/4, h)$ as depicted on **Figure 5**. The three coding units are separated by two split lines represented by dashed lines on Figure 5.
- 10 In HEVC and JEM, most of these modes are disallowed for the blocks located at the border of the picture which are thus not coded efficiently.

3. BRIEF SUMMARY

A decoding method for decoding a current block, the current block including at least a portion of a picture, is disclosed that comprises :

- determining whether the current block extends beyond a boundary of the picture;
- determining for each split mode of a plurality of split modes whether the split mode is allowed for the current block by checking whether at least one of the split lines is co-located with one of the picture boundaries or whether a size of a block part, i.e. a remaining part resulting from the splitting, inside the picture along the picture boundary is a multiple of a minimum block size;
- decoding from a bitstream a current split mode of the current block responsive to the allowed split modes; and
- decoding the current block according to the current split mode.

25

A decoding apparatus configured to decode a current block, the current block including at least a portion of a picture, is disclosed that comprises :

- means for determining whether the current block extends beyond a boundary of the picture;
- means for determining for each split mode of a plurality of split modes whether the split mode is allowed for the current block by checking whether at least one of the split lines is co-located with one of the picture boundaries or whether a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size;

30

- means for decoding from a bitstream a current split mode of the current block responsive to the allowed split modes; and
- means for decoding the current block according to the current split mode.

- 5 A decoding apparatus is disclosed that comprises a communication interface configured to access at least a bitstream and at least one processor configured to:
- determine whether a current block extends beyond a boundary of the picture, wherein the current block includes at least a portion of a picture;
 - determine for each split mode of a plurality of split modes whether the split mode is
- 10 allowed for the current block by checking whether at least one of the split lines is co-located with one of the picture boundaries or whether a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size;
- decode from the accessed bitstream a current split mode of the current block responsive to the allowed split modes; and
- 15 - decode the current block according to the current split mode.

The following embodiments apply to the decoding method and decoding apparatus disclosed above.

- 20 According to a specific embodiment, the plurality of split modes comprises at least two of the following split modes, herein h is the height of the current block and w is the width of the current block:
- splitting the block horizontally into one upper sub-block of height $h/4$ and one bottom sub-block of height $3h/4$;
- 25 - splitting the block horizontally into one bottom sub-block of height $h/4$ and one upper sub-block of height $3h/4$;
- splitting the block vertically into one left sub-block of width $w/4$ and one right sub-block of width $3w/4$;
- 30 - splitting the block vertically into one right sub-block of width $w/4$ and one left sub-block of width $3w/4$;
- splitting the block horizontally into one sub-block of height $h/4$, one central sub-block of height $h/2$ and one sub-block of height $h/4$;
 - splitting the block vertically into one sub-block of width $w/4$, one central sub-block of width $w/2$ and one sub-block of width $w/4$;

- splitting the block horizontally into two sub-blocks of height $h/2$; and
- splitting the block vertically into two sub-blocks of width $w/2$.

According to a specific characteristic, the minimum block size in at least one of
5 height and width is equal to 4.

An encoding method for encoding a current block, the current block including
at least a portion of a picture, is disclosed that comprises :

- determining whether the current block extends beyond a boundary of the picture;
- 10 - determining, for each split mode of a plurality of split modes, whether the split mode
is allowed for the current block by checking whether at least one of the split lines is co-
located with one of the picture boundaries or whether a size of a block part inside the
picture along the picture boundary is a multiple of a minimum block size;
- encoding in a bitstream a current split mode of the current block determined
15 responsive to the allowed split modes; and
- encoding the current block in the bitstream according to the current split mode.

An encoding apparatus configured to encode a current block, the current block
including at least a portion of a picture, is disclosed that comprises :

- 20 - means for determining whether the current block extends beyond a boundary of the
picture;
- means for determining, for each split mode of a plurality of split modes, whether the
split mode is allowed for the current block by checking whether at least one of the split
lines is co-located with one of the picture boundaries or whether a size of a block part
25 inside the picture along the picture boundary is a multiple of a minimum block size;
- means for encoding in a bitstream a current split mode of the current block determined
responsive to the allowed split modes; and
- means for encoding the current block in the bitstream according to the current split
mode.

30

An encoding apparatus comprising a communication interface configured to
access at least a current block, the current block including at least a portion of a picture,
is disclosed. The encoding apparatus further comprises at least one processor
configured to:

- determine whether the accessed current block extends beyond a boundary of the picture;
- determine, for each split mode of a plurality of split modes, whether the split mode is allowed for the current block by checking whether at least one of the split lines is co-
- 5 located with one of the picture boundaries or whether a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size;
- encode in a bitstream a current split mode of the current block determined responsive to the allowed split modes; and
- encode the accessed current block in the bitstream according to the current split mode.

10

A bitstream is disclosed that comprises:

- coded data representative of a current split mode of a current block determined responsive to allowed split modes, the current block including at least a portion of a picture;
- 15 - coded data representative of the current block encoded according to a current split mode;

wherein the allowed split modes are determined by:

- determining whether the current block extends beyond a boundary of the picture;
- determining, for each split mode of a plurality of split modes, whether the split mode
- 20 is allowed for the current block by checking whether at least one of the split lines is co-located with one of the picture boundaries or whether a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size.

A non-transitory processor readable medium having stored thereon a bitstream

25 comprising:

- coded data representative of a current split mode of a current block determined responsive to allowed split modes, the current block including at least a portion of a picture;
- coded data representative of the current block encoded according to a current split
- 30 mode;

wherein the allowed split modes are determined by:

- determining whether the current block extends beyond a boundary of the picture;
- determining, for each split mode of a plurality of split modes, whether the split mode is allowed for the current block by checking whether at least one of the split lines is co-

located with one of the picture boundaries or whether a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size.

Transmitting method and apparatus for transmitting the above bitstream are also
5 disclosed.

The following embodiments apply to the coding method, coding apparatus,
bitstream, processor readable medium, transmitting method and transmitting apparatus
disclosed above.
10

The plurality of split modes comprises at least two of the following split modes
wherein h is the height of the current block and w is the width of the current block:

- splitting the block horizontally into one upper sub-block of height $h/4$ and one bottom
sub-block of height $3h/4$;
- 15 - splitting the block horizontally into one bottom sub-block of height $h/4$ and one upper
sub-block of height $3h/4$;
- splitting the block vertically into one left sub-block of width $w/4$ and one right sub-
block of width $3w/4$;
- splitting the block vertically into one right sub-block of width $w/4$ and one left sub-
20 block of width $3w/4$;
- splitting the block horizontally into one sub-block of height $h/4$, one central sub-block
of height $h/2$ and one sub-block of height $h/4$;
- splitting the block vertically into one sub-block of width $w/4$, one central sub-block
of width $w/2$ and one sub-block of width $w/4$;
- 25 - splitting the block horizontally into two sub-blocks of height $h/2$; and
- splitting the block vertically into two sub-blocks of width $w/2$.

According to a specific characteristic, the minimum block size in at least one of
height and width is equal to 4.
30

4. BRIEF SUMMARY OF THE DRAWINGS

- Figure 1 represents a coding tree unit divided into smaller coding units
according to various split modes;

- Figure 2 represents in the form of a tree the splitting of the coding tree unit of figure 1;
- Figures 3 illustrates the symmetric binary split modes in addition to the quad-tree and no split modes;
- 5 - Figure 4 illustrates the asymmetric split modes;
- Figure 5 illustrates the symmetric triple split modes;
- Figure 6 represents blocks split by a quad-tree that extends beyond the picture boundaries;
- Figure 7 represents an exemplary architecture of a transmitter configured to encode a picture in a bitstream according to a specific and non-limiting embodiment;
- 10 - Figure 8 illustrates an exemplary video encoder according to a specific and non-limiting embodiment;
- Figure 9 represents an exemplary architecture of a receiver configured to decode a picture in a bitstream according to a specific and non-limiting embodiment;
- 15 - Figure 10 illustrates an exemplary video decoder according to a specific and non-limiting embodiment;
- Figure 11 represents a flowchart of a method for decoding a block of a picture from a bitstream according to a specific and non-limiting embodiment;
- 20 - Figure 12 represents a flowchart of a method for decoding a syntax according to a specific and non-limiting embodiment;
- Figure 13 represents splitting of blocks that extend beyond the picture boundaries ;
- Figures 14 and 15 illustrate the pruning of the syntax;
- 25 - Figure 16 represents splitting of blocks that extend beyond the picture boundaries ; and
- Figure 17 represents a flowchart of a method for encoding a block of a picture in a bitstream according to a specific and non-limiting embodiment.

30 5. DETAILED DESCRIPTION

It is to be understood that the figures and descriptions have been simplified to illustrate elements that are relevant for a clear understanding of the present principles, while eliminating, for purposes of clarity, many other elements found in typical encoding and/or decoding devices. It will be understood that, although the terms first

and second may be used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another.

A picture is an array of luma samples in monochrome format or an array of luma samples and two corresponding arrays of chroma samples in 4:2:0, 4:2:2, and 4:4:4 colour format. Generally, a “block” addresses a specific area in a sample array (e.g., luma Y), and a “unit” includes the collocated block of all color components (luma Y and possibly chroma Cb and chroma Cr). A slice is an integer number of basic coding units such as HEVC coding tree units or H.264 macroblock units. A slice may consist of a complete picture as well as part thereof. Each slice may include one or more slice segments.

In the following, the term “block” or “picture block” can be used to refer to any one of a CTU, a CU, a PU, a TU, a CB, a PB and a TB. In addition, the term “block” or “picture block” can be used to refer to a macroblock, a partition and a sub-block as specified in H.264/AVC or in other video coding standards, and more generally to refer to an array of samples of various sizes. JEM does not distinguish between CU, PU and TU since a CU is a block of samples on which a same transform and prediction information is applied. A CU may be split into several coding units according to a split mode. The lines separating coding units in a larger coding unit are called split lines.

In the following, the word “reconstructed” and “decoded” can be used interchangeably. Usually but not necessarily “reconstructed” is used on the encoder side while “decoded” is used on the decoder side. It should be noted that the term “decoded” or “reconstructed” may mean that a bitstream is partially “decoded” or “reconstructed,” for example, the signals obtained after deblocking filtering but before SAO filtering, and the reconstructed samples may be different from the final decoded output that is used for display. We may also use the terms “image,” “picture,” and “frame” interchangeably. We may use the terms “border” and “boundary” interchangeably.

Various embodiments are described with respect to the HEVC standard. However, the present principles are not limited to HEVC, and can be applied to other standards, recommendations, and extensions thereof, including for example HEVC or HEVC extensions like Format Range (RExt), Scalability (SHVC), Multi-View (MV-HEVC) Extensions and H.266. The various embodiments are described with respect to the encoding/decoding of a slice. They may be applied to encode/decode a whole picture or a whole sequence of pictures.

Various methods are described above, and each of the methods comprises one or more steps or actions for achieving the described method. Unless a specific order of steps or actions is required for proper operation of the method, the order and/or use of specific steps and/or actions may be modified or combined.

5

Allowing the new split modes depicted on Figures 3, 4 and 5 may dramatically increase the signaling cost. There is thus a need to improve the syntax without decreasing the quality of the reconstructed pictures. The blocks that are completely inside the picture may use all types of split modes. However, for the blocks located on the boundaries/borders of the picture especially at the right and at the bottom boundaries of the picture (in case of classical raster scan order or Z-scan) some of the split modes may be disallowed especially if the block extends outside the picture. In HEVC and JEM, in the case where a block is extends partly outside the picture, the block is necessarily split according to a quad-tree as depicted on **Figure 6**. In this case, there is no need to signal the split mode since a single split mode is allowed for these blocks that are partly outside the picture. Indeed, once a block is determined to extend partly outside the picture, then the split mode for this block is inferred to be a quad-tree split (e.g. QT_SPLIT). However, by inferring the split mode to be a quad-tree split, the quality of the reconstructed block at the boundaries of the picture may decrease since the number of split modes to encode the block is drastically limited. In addition, such a method requires the width and height of the picture to be a multiple of the minimum coding block size (typically 4 pixels) to ensure that a border of a picture is reached by successive quad-tree splits. In case of multiple tiles or slices in a frame, same issue will occur at the boundaries of the tiles/slices. This could lead to similar quality issues at the border of the tiles/slices, and especially if used with some layout for 360° videos.

10
15
20
25

Figure 7 represents an exemplary architecture of a transmitter 1000 configured to encode a picture in a bitstream according to a specific and non-limiting embodiment.

The transmitter 1000 comprises one or more processor(s) 1005, which could comprise, for example, a CPU, a GPU and/or a DSP (English acronym of Digital Signal Processor), along with internal memory 1030 (e.g. RAM, ROM, and/or EPROM). The transmitter 1000 comprises one or more communication interface(s) 1010 (e.g. a keyboard, a mouse, a touchpad, a webcam), each adapted to display output information and/or allow a user to enter commands and/or data; and a power source 1020 which

30

may be external to the transmitter 1000. The transmitter 1000 may also comprise one or more network interface(s) (not shown). Encoder module 1040 represents the module that may be included in a device to perform the coding functions. Additionally, encoder module 1040 may be implemented as a separate element of the transmitter 1000 or may
5 be incorporated within processor(s) 1005 as a combination of hardware and software as known to those skilled in the art.

The picture may be obtained from a source. According to different embodiments, the source can be, but is not limited to:

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

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

25 According to an exemplary and non-limiting embodiment, the transmitter 1000 further comprises a computer program stored in the memory 1030. The computer program comprises instructions which, when executed by the transmitter 1000, in particular by the processor 1005, enable the transmitter 1000 to execute the encoding method described with reference to Figure 17. According to a variant, the computer program is
30 stored externally to the transmitter 1000 on a non-transitory digital data support, e.g. on an external storage medium such as a HDD, CD-ROM, DVD, a read-only and/or DVD drive and/or a DVD Read/Write drive, all known in the art. The transmitter 1000 thus comprises a mechanism to read the computer program. Further, the transmitter 1000

could access one or more Universal Serial Bus (USB)-type storage devices (e.g., “memory sticks.”) through corresponding USB ports (not shown).

According to exemplary and non-limiting embodiments, the transmitter 1000 can be, but is not limited to:

- 5 - a mobile device;
- a communication device;
- a game device;
- a tablet (or tablet computer);
- a laptop;
- 10 - a still picture camera;
- a video camera;
- an encoding chip or encoding device/apparatus;
- a still picture server; and
- a video server (e.g. a broadcast server, a video-on-demand server or a web
- 15 server).

Figure 8 illustrates an exemplary video encoder 100, e.g. a HEVC video encoder or encoder of the JEM type, adapted to execute the encoding method of Figure 17. The encoder 100 is an example of a transmitter 1000 or part of such a transmitter 1000.

For coding, a picture is usually partitioned into basic coding units, e.g. into coding tree units (CTU) in HEVC or into macroblock units in H.264. A set of possibly consecutive basic coding units is grouped into a slice. A basic coding unit contains the basic coding blocks of all color components. In HEVC, the smallest CTB size 16x16 corresponds to a macroblock size as used in previous video coding standards. It will be understood that, although the terms CTU and CTB are used herein to describe encoding/decoding methods and encoding/decoding apparatus, these methods and apparatus should not be limited by these specific terms that may be worded differently (e.g. macroblock) in other standards such as H.264.

In the exemplary encoder 100, a picture is encoded by the encoder elements as described below. The picture to be encoded is processed in units of CUs that may be square as in HEVC or rectangular as in JEM. Each CU is encoded using either an intra or inter mode. When a CU is encoded in an intra mode, it performs intra prediction (160). In an inter mode, motion estimation (175) and compensation (170) are

performed. The encoder decides (105) which one of the intra mode or inter mode to use for encoding the CU, and indicates the intra/inter decision by a prediction mode flag. Residuals are calculated by subtracting (110) a predicted sample block (also known as a predictor) from the original picture block.

5 CUs in intra mode may be predicted from reconstructed neighboring samples within the same slice. A set of 35 intra prediction modes is available in HEVC, including a DC, a planar and 33 angular prediction modes. The intra prediction reference is reconstructed from the row and column adjacent to the current block.

For an inter CU, the corresponding coding block may be further partitioned into
10 one or more prediction blocks as in HEVC. Inter prediction is performed on the PB level, and the corresponding PU contains the information about how inter prediction is performed.

The residuals are transformed (125) and quantized (130). The quantized transform coefficients, as well as motion vectors and other syntax elements, are entropy
15 coded (145) to output a bitstream. The encoder may also skip the transform and apply quantization directly to the non-transformed residual signal on a 4x4 TU basis. The encoder may also bypass both transform and quantization, i.e., the residual is coded directly without the application of the transform or quantization process. In direct PCM coding, no prediction is applied and the coding unit samples are directly coded into the
20 bitstream.

The encoder comprises a decoding loop and thus decodes an encoded block to provide a reference for further predictions. The quantized transform coefficients are de-quantized (140) and inverse transformed (150) to decode residuals. A picture block is reconstructed by combining (155) the decoded residuals and the predicted sample
25 block. An in-loop filter (165) is applied to the reconstructed picture, for example, to perform deblocking/SAO (Sample Adaptive Offset) filtering to reduce coding artifacts. The filtered picture may be stored in a reference picture buffer (180) and used as reference for other pictures.

30 **Figure 9** represents an exemplary architecture of a receiver 2000 configured to decode a picture from a bitstream to obtain a decoded picture according to a specific and non-limiting embodiment.

The receiver 2000 comprises one or more processor(s) 2005, which could comprise, for example, a CPU, a GPU and/or a DSP (English acronym of Digital Signal

Processor), along with internal memory 2030 (e.g. RAM, ROM and/or EPROM). The receiver 2000 comprises one or more communication interface(s) 2010 (e.g. a keyboard, a mouse, a touchpad, a webcam), each adapted to display output information and/or allow a user to enter commands and/or data (e.g. the decoded picture); and a power source 2020 which may be external to the receiver 2000. The receiver 2000 may also comprise one or more network interface(s) (not shown). The decoder module 2040 represents the module that may be included in a device to perform the decoding functions. Additionally, the decoder module 2040 may be implemented as a separate element of the receiver 2000 or may be incorporated within processor(s) 2005 as a combination of hardware and software as known to those skilled in the art.

The bitstream may be obtained from a source. According to different embodiments, the source can be, but is not limited to:

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

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

According to a specific and non-limiting embodiment, the receiver 2000 further comprises a computer program stored in the memory 2030. The computer program comprises instructions which, when executed by the receiver 2000, in particular by the processor 2005, enable the receiver to execute the decoding method described with reference to Figure 11. According to a variant, the computer program is stored

externally to the receiver 2000 on a non-transitory digital data support, e.g. on an external storage medium such as a HDD, CD-ROM, DVD, a read-only and/or DVD drive and/or a DVD Read/Write drive, all known in the art. The receiver 2000 thus comprises a mechanism to read the computer program. Further, the receiver 2000 could
5 access one or more Universal Serial Bus (USB)-type storage devices (e.g., “memory sticks.”) through corresponding USB ports (not shown).

According to exemplary and non-limiting embodiments, the receiver 2000 can be, but is not limited to:

- a mobile device;
- 10 - a communication device;
- a game device;
- a set top box;
- a TV set;
- a tablet (or tablet computer);
- 15 - a laptop;
- a video player, e.g. a Blu-ray player, a DVD player;
- a display; and
- a decoding chip or decoding device/apparatus.

20 **Figure 10** illustrates a block diagram of an exemplary video decoder 200, e.g. an HEVC video decoder or a decoder of the JEM type, adapted to execute the decoding method of Figure 11. The video decoder 200 is an example of a receiver 2000 or part of such a receiver 2000. In the exemplary decoder 200, a bitstream is decoded by the decoder elements as described below. Video decoder 200 generally performs a
25 decoding pass reciprocal to the encoding pass as described in Figure 8, which performs video decoding as part of encoding video data.

In particular, the input of the decoder includes a video bitstream, which may be generated by the video encoder 100. The bitstream is first entropy decoded (230) to obtain transform coefficients, motion vectors, and other coded information. The
30 transform coefficients are de-quantized (240) and inverse transformed (250) to decode residuals. The decoded residuals are then combined (255) with a predicted sample block (also known as a predictor) to obtain a decoded/reconstructed picture block. The predicted sample block may be obtained (270) from intra prediction (260) or motion-compensated prediction (i.e., inter prediction) (275). An in-loop filter (265) is applied

to the reconstructed picture. The in-loop filter may comprise a deblocking filter and a SAO filter. The filtered picture is stored at a reference picture buffer (280).

Figure 11 represents a flowchart of a method for decoding a current block from a bitstream according to a specific and non-limiting embodiment, wherein the current block includes at least a portion of a picture.

The method starts at step S200. At step S210, a receiver 2000 such as the decoder 200 accesses a bitstream.

At step S220, the receiver determines whether a current block to be decoded is partly outside the picture, i.e. extends beyond one boundary of the picture. This may be determined by computing Diff_X and Diff_Y as follows:

$\text{Diff_X} = \text{blkPosX} + \text{width} - \text{frameWidth}$; and

$\text{Diff_Y} = \text{blkPosY} + \text{height} - \text{frameHeight}$

where blkPosX is the position of the current block in X, e.g. coordinate in X of the top left pixel of the block, and blkPosY is the position of the current block in Y, e.g. coordinate in Y of the top left pixel of the block, width is the block width, height is the block height, frameWidth is the picture width and frameHeight is the picture height. If $\text{Diff_X} \leq 0$ and $\text{Diff_Y} \leq 0$, then the block is completely inside the picture, if $\text{Diff_X} > 0$ && $\text{Diff_Y} \leq 0$, a right part of the block is outside the picture, if $\text{Diff_X} > 0$ && $\text{Diff_Y} > 0$, a bottom part of the block is outside the picture, otherwise a right part and a bottom part of the block are outside the picture.

In the case the current block is partly outside the picture, the method continues at step S230 otherwise the method continues at step S240. In this latter case, the block is completely inside the picture and all split modes may be allowed. In particular cases, some of the split modes may be disallowed for blocks completely inside the picture. For example, if the width of the block is equal to the minimum width authorized for a block (e.g. 4 pixels), then the vertical split modes are disallowed. In the same way, if the height of the block is equal to the minimum height authorized for a block (e.g. 4 pixels), then the horizontal split modes are disallowed. In addition, some of the split modes may be disabled by high level syntax elements, e.g. for some specific profiles. In this latter case, the disabled split modes are disallowed for all blocks of the image or for possibly all blocks of a video. In the specific case where the current block is outside the picture horizontally and vertically as for example the bottom right block in Figure 6, then no split modes is allowed except the QT_SPLIT mode.

At step S230, the receiver determines for each split mode of a plurality of split modes whether the current block is allowed to undergo splitting according to this split mode by checking whether at least one of the split lines is co-located with one of the picture borders. In an exemplary embodiment, the plurality of split modes comprises the following split modes: HOR_UP, HOR_DOWN, HOR_TRIPLE, HOR, VER_TRIPLE, VER, VER_LEFT, VER_RIGHT. Usually but not necessarily, the split modes other than QT_SPLIT mode are looked into only when splitting according to the QT_SPLIT mode is decided. In a variant, the plurality of split modes comprises a subset of the split modes listed previously. With the introduction of triple split modes or asymmetric binary split modes, many different splits may be available to reach the picture border. In this embodiment, an asymmetric binary split mode is allowed at a border of a picture for a current block if the split line is exactly on the picture border. A triple split mode is allowed at a border of a picture for a current block if the first split line or second split line is exactly on the frame border.

In a specific embodiment, the split modes allowed are as specified below.

The mode NO_SPLIT is allowed only in the case where the block is completely inside the picture, otherwise this mode is not allowed. A flag noSplitAllowed is set to 1 (or to true) to indicate that the NO_SPLIT mode is allowed.

The mode HOR is allowed if the block does not extend beyond the boundary of the picture at right (or at left depending on scanning order). In a variant, the mode HOR is allowed if the right part of the block does not extend beyond the boundary of the picture at right and, in the case where the block extends beyond the boundary of the picture at bottom, if the border of the picture and the horizontal split line resulting from a HOR splitting are co-located. In this case, a flag binaryHorAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

The mode VER is allowed if the block does not extend beyond the boundary of the picture at bottom (or at top depending on scanning order). In a variant, the mode VER is allowed if the block does not extend beyond the boundary of the picture at bottom and, in the case where the block extends beyond the boundary of the picture at right, if the border of the picture and the vertical split line resulting from a VER splitting are co-located.

In this case, a flag binaryVerAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

The mode HOR_TRIPLE is allowed if the block does not extend beyond the boundary of the picture at right and, in the case where the block extends beyond the boundary of the picture at bottom, if the border of the picture is co-located with one of the horizontal split lines resulting from a HOR_TRIPLE splitting, i.e. if $\text{blkPosY} + \text{height}/4$ is equal to frameHeight or if $\text{blkPosY} + 3*\text{height}/4$ is equal to frameHeight . A flag horizontalTripleAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

The mode VER_TRIPLE is allowed if the block does not extend beyond the boundary of the picture at bottom and, in the case where the block extends beyond the boundary of the picture at right, if the border of the picture is co-located with one of the vertical split lines resulting from a VER_TRIPLE splitting, i.e. if $\text{blkPosX} + \text{width}/4$ is equal to frameWidth or if $\text{blkPosX} + 3*\text{width}/4$ is equal to frameWidth . A flag verticalTripleAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

HOR_UP is allowed if the block does not extend beyond the boundary of the picture at right and, in the case where the block extends beyond the boundary of the picture at bottom, if the border of the picture is co-located with the horizontal split line resulting from a HOR_UP splitting, i.e. if $\text{blkPosY} + \text{height}/4$ is equal to frameHeight . A flag horUpAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

HOR_DOWN is allowed if the block does not extend beyond the boundary of the picture at right and, in the case where the block extends beyond the boundary of the picture at bottom, if the border of the picture is co-located with the horizontal split line resulting from a HOR_DOWN splitting, i.e. if $\text{blkPosY} + 3*\text{height}/4$ is equal to frameHeight . A flag horDownAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

VER_LEFT is allowed if the block does not extend beyond the boundary of the picture at bottom and, in the case where the block extends beyond the boundary of the picture at right, if the border of the picture is co-located with the vertical split line resulting from a VER_LEFT splitting, i.e. if $\text{blkPosX} + \text{width}/4$ is equal to frameWidth . A flag verLeftAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

VER_RIGHT is allowed if the block does not extend beyond the boundary of the picture at bottom and, in the case where the block extends beyond the boundary of the picture

at right, if the border of the picture is co-located with the vertical split line resulting from a VER_RIGHT splitting, i.e. if $\text{blkPosX} + 3 \cdot \text{width} / 4$ is equal to frameWidth . A flag `verRightAllowed` is set to 1 to indicate that the corresponding split mode is allowed for the current block.

- 5 In a variant, additional splitting constraints may be added to the above decisions of allowability.

If at least one of the HOR and HOR_TRIPLE split modes is allowed then `horBinaryOrTripleAllowed` flag is set to 1.

- 10 If at least one of the VER and VER_TRIPLE split modes is allowed then `verBinaryOrTripleAllowed` flag is set to 1.

If at least one of the vertical split modes is allowed then `verticalSplitAllowed` flag is set to 1. If at least one of the horizontal split modes is allowed then `horizontalSplitAllowed` flag is set to 1.

- 15 If at least one of the asymmetric horizontal split modes is allowed then `horizontalAsymmetricAllowed` flag is set to 1.

If at least one of the asymmetric vertical split modes is allowed then `verticalAsymmetricAllowed` flag is set to 1.

- 20 At step S240, the receiver decodes from the accessed bitstream a current split mode for the current block responsive to the allowed split modes determined at step S230. Indeed, knowing the allowed modes for a current block makes it possible to simplify the signaling and thus saving bits. An example is provided by the syntax in Table 1 in connection with **Figure 12**. In table 1 and figure 12, by convention the value true is equal to 1 and the value false is equal to 0. Consequently, a syntax element being true is the same as the syntax element being equal to 1 and a syntax element being false is the same as the syntax element being equal to 0. In table 1, the syntax elements
25 decoded from the bitstream are indicated in bold.

bt_split_mode(x0,y0,width,height,cqtDepth){
if(btSplitAllowed(x0,y0,width,height) && noSplitAllowed){
btSplitFlag
}
if(btSplitFlag==true){
if(horizontalSplitAllowed && verticalSplitAllowed){
btSplitOrientation
}
if(btSplitOrientation==OrientationHOR){
if(horizontalAsymmetricAllowed && horBinaryOrTripleAllowed){
asymmetricSplitFlag
}
if(asymmetricSplitFlag==true && horUpAllowed && horDownAllowed){
horizontalAsymmetricType
}
else if(asymmetricSplitFlag==false && horizontalTripleAllowed && binaryHorAllowed){
horizontalTripleFlag
}
}else if (btSplitOrientation==OrientationVER) {
if(verticalAsymmetricAllowed && verBinaryOrTripleAllowed){
asymmetricSplitFlag
}
if(asymmetricSplitFlag==true && verLeftAllowed && verRightAllowed){
verticalAsymmetricType
}
else if(asymmetricSplitFlag==false && verticalTripleAllowed && binaryVerAllowed){
verticalTripleFlag
}
}
}
}

On Figure 12 only the steps relating to the binary split mode decoding of a current block are represented.

At step S1020, the receiver checks whether binary or triple splittings and NO_SPLIT mode are allowed for the current block to be decoded. Checking whether binary/triple splitting is allowed (*btSplitAllowed()*) typically consists in checking if the current block width (*w*) or the current block height (*h*) is higher than a threshold
5 (*minBTSize*). A typical minimum rectangular block size in width or height is *minBTSize* = 4.

If the binary/triple splitting is not allowed, the block is not split (*btSplitMode* is set to NO_SPLIT at step S1045) and no other syntax element related to the splitting is decoded. If both binary splitting and NO_SPLIT mode are allowed, a flag *btSplitFlag*
10 is decoded from the bitstream at step S1030 that indicates if the current block is split or not.

Otherwise (if binary/triple splitting is allowed and the NO_SPLIT mode is not allowed), the *btSplitFlag* is not decoded and is inferred to be equal to 1 (or true) and the method continues at step S1050. For a current block that extends partially beyond the
15 boundaries of the picture, the NO_SPLIT mode is not allowed, therefore the *btSplitFlag* is not decoded and is inferred to be equal to 1.

At step S1040, the receiver checks whether the current block is split or not. If the current block is not split (e.g. *btSplitFlag* is False at step S1040), then no other syntax element related to the splitting is decoded. *btSplitMode* is set to NO_SPLIT at
20 step S1045.

In the case where the current block is split (*btSplitFlag* is True at step S1040), the receiver determines at step S1050 whether both horizontal and vertical split modes are allowed for the current block. If both horizontal and vertical split modes are allowed, a syntax element (*btSplitOrientation*) is decoded at step S1060 to indicate the binary
25 splitting orientation (horizontal or vertical). If either horizontal or vertical splitting is not allowed, then this syntax element (*btSplitOrientation*) is not decoded. In this latter case, the binary splitting orientation is inferred. If horizontal splitting is not allowed (*horizontalSplitAllowed* is equal to 0 or False), then the *btSplitOrientation* is inferred to be equal to OrientationVER. Otherwise, if vertical splitting is not allowed
30 (*verticalSplitAllowed* is equal to 0 or False), the *btSplitOrientation* is inferred to be equal to OrientationHOR.

Then, the receiver checks at step S1070 if both flags *horizontalAsymmetricAllowed* and *horizontalBinOrTripleAllowed* are true. If both

flags are true then a syntax element (`asymmetricSplitFlag`) is decoded at step S1090 and the method continues at step S1100. If `horizontalAsymmetricAllowed` is true and `horizontalBinOrTripleAllowed` is false, `asymmetricSplitFlag` is inferred to be equal to true and the method continues at step S1100. Otherwise
5 (horizontalAsymmetricAllowed is false and horizontalBinOrTripleAllowed is true), `asymmetricSplitFlag` is inferred to be equal to false and the method continues at step S1100.

At step S1100, the receiver checks whether the flag `asymmetricSplitFlag` is true or false. If `asymmetricSplitFlag` is true, then the method continues at step S1110, else
10 the method continues at step S1120.

At step S1110, the receiver checks if both flags `horUpAllowed` and `horDownAllowed` are true. If both flags are true, then a syntax element `horizontalAsymmetricType` is decoded at step S1112, otherwise (if one of the flags is false), then the split mode of the block is inferred to `HOR_UP` if `horDownAllowed` is
15 false or to `HOR_DOWN` if `horUpAllowed` is false.

At step S1120, the receiver checks if both flags `binaryHorAllowed` and `horizontalTripleAllowed` are true. If both flags are true, then a syntax element `horizontalTripleFlag` is decoded at step S1122, otherwise (if one of the flags is false), then the split mode of the block is inferred to `HOR_TRIPLE` if `binaryHorAllowed` is
20 false or to `HOR` if `horizontalTripleAllowed` is false.

The receiver proceeds the same way for the vertical orientation.

The receiver checks at step S1080 if both flags `verticalAsymmetricAllowed` and `verticalBinOrTripleAllowed` are true. If both flags are true then a syntax element (`asymmetricSplitFlag`) is decoded at step S1090 and the method continues at step
25 S1200. If `verticalAsymmetricAllowed` is true and `verticalBinOrTripleAllowed` is false, `asymmetricSplitFlag` is inferred to be equal to true and the method continues at step S1200. Otherwise (`verticalAsymmetricAllowed` is false and `verticalBinOrTripleAllowed` is true), `asymmetricSplitFlag` is inferred to be equal to false and the method continues at step S1200.

At step S1200, the receiver checks whether the flag `asymmetricSplitFlag` is true or false. If `asymmetricSplitFlag` is true, then the method continues at step S1210, else
30 the method continues at step S1220.

At step S1210, the receiver checks if both flags `verLeftAllowed` and `verRightAllowed` are true. If both flags are true, then a syntax element

verticalAsymmetricType is decoded at step S1212, otherwise (if one of the flags is false), then the split mode btSplitMode of the block is inferred to VER_LEFT if verRightAllowed is false or to VER_RIGHT if verLeftAllowed is false.

At step S1220, the receiver checks if both flags binaryVerAllowed and verticalTripleAllowed are true. If both flags are true, then a syntax element verticalTripleFlag is decoded at step S1222, otherwise (if one of the flags is false), then the split mode btSplitMode of the block is inferred to VER_TRIPLE if binaryVerAllowed is false or to VER if verticalTripleAllowed is false.

As an example, for the top right block of **Figure 13**, binary tree syntax can be pruned as depicted on **Figure 14** to remove unnecessary splits. In this example only VER_TRIPLE and VER_RIGHT are allowed, so only one flag **asymmetricSplitFlag** is needed to decode the split. In the same way as depicted on **Figure 15**, for the bottom left block of Figure 13, binary tree syntax can be pruned to remove unnecessary splits. In this example only HOR_TRIPLE and HOR_UP are available. These two split modes give the same partition for the bottom left block, so one of the two splits can be removed. More generally, in case where two different split modes give the same partition of the block, only one is allowed. Figures 14 and 15 are graphic representations of the syntax as defined in table 1. The method according to the present principles thus enable competition between Quad Tree splitting and Binary/Triple splitting at the border of the picture with improved rate distortion performance.

In a first variant of step S230, the receiver determines whether said current block is allowed to undergo splitting according to at least one split mode of a plurality of split modes by checking whether the size (i.e. the width in case of vertical split or the height in case of horizontal split) of the remaining part of the block after splitting is a multiple of the minimum block size (typically 4 pixels). This embodiment may be combined with the previous embodiment. The remaining part is the part of the block, i.e. the sub-block resulting from the splitting, located inside the picture along a picture's boundary. As an example, in a first step, it is checked whether the border of the picture is co-located with one of the split lines, if not then it is checked whether the size (width in case of vertical split or height in case of horizontal split) of the remaining part after splitting is a multiple of the minimum block size. In this embodiment the triple split modes can be used even when the border of the picture is not co-located with one of the split lines. In **Figure 16**, a block of 64x64 is split vertically with a left part of 16x64, a

central part of 32x64 and a remaining right part of size 8x64 that is identified by a hatched block.

Equivalently, an asymmetric binary split is allowed if a size (width in case of vertical split or height in case of horizontal split) of the remaining part along the boundary of the picture is still a multiple of the minimum block size (for example 4).

This second embodiment may be advantageously combined with the first embodiment as disclosed below.

The mode NO_SPLIT is allowed only in the case where the block is completely inside the picture, otherwise this mode is not allowed. A flag noSplitAllowed is set to 1 to indicate that the NO_SPLIT mode is allowed.

The mode HOR is allowed if the block does not extend beyond the boundary of the picture at right. In a variant, the mode HOR is allowed if the right part of the block does not extend beyond the boundary of the picture at right and, in the case where the block extends beyond the boundary of the picture at bottom, if the border of the picture and the horizontal split line resulting from a HOR splitting are co-located or if the height of the remaining part of the block after splitting is a multiple of the minimum block size. In this case, a flag binaryHorAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

The mode VER is allowed if the block does not extend beyond the boundary of the picture at bottom. In a variant, the mode VER is allowed if the block does not extend beyond the boundary of the picture at bottom and, in the case where the block extends beyond the boundary of the picture at right, if the border of the picture and the vertical split line resulting from a VER splitting are co-located or if the width of the remaining part of the block after splitting is a multiple of the minimum block size. In this case, a flag binaryVerAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

The mode HOR_TRIPLE is allowed if the block does not extend beyond the boundary of the picture at right and, in the case where the block extends beyond the boundary of the picture at bottom, if the border of the picture is co-located with one of the horizontal split lines resulting from a HOR_TRIPLE splitting, i.e. if $\text{blkPosY} + \text{height}/4$ is equal to frameHeight or if $\text{blkPosY} + 3*\text{height}/4$ is equal to frameHeight , or if the height of the remaining part of the block after splitting is a multiple of the minimum block size. A flag horizontalTripleAllowed is set to 1 to indicate that the corresponding split mode is allowed for the current block.

The mode VER_TRIPLE is allowed if the block does not extend beyond the boundary of the picture at bottom and, in the case where the block extends beyond the boundary of the picture at right, if the border of the picture is co-located with one of the vertical split lines resulting from a VER_TRIPLE splitting, i.e. if $\text{blkPosX} + \text{width}/4$ is equal to frameWidth or if $\text{blkPosX} + 3*\text{width}/4$ is equal to frameWidth , or if the width of the remaining part of the block after splitting is a multiple of the minimum block size. A flag `verticalTripleAllowed` is set to 1 to indicate that the corresponding split mode is allowed for the current block.

HOR_UP is allowed if the block does not extend beyond the boundary of the picture at right and, in the case where the block extends beyond the boundary of the picture at bottom, if the border of the picture is co-located with the horizontal split line resulting from a HOR_UP splitting, i.e. if $\text{blkPosY} + \text{height}/4$ is equal to frameHeight , or if the height of the remaining part of the block after splitting is a multiple of the minimum block size. A flag `horUpAllowed` is set to 1 to indicate that the corresponding split mode is allowed for the current block.

HOR_DOWN is allowed if the block does not extend beyond the boundary of the picture at right and, in the case where the block extends beyond the boundary of the picture at bottom, if the border of the picture is co-located with the horizontal split line resulting from a HOR_DOWN splitting, i.e. if $\text{blkPosY} + 3*\text{height}/4$ is equal to frameHeight , or if the height of the remaining part of the block after splitting is a multiple of the minimum block size. A flag `horDownAllowed` is set to 1 to indicate that the corresponding split mode is allowed for the current block.

VER_LEFT is allowed if the block does not extend beyond the boundary of the picture at bottom and, in the case where the block extends beyond the boundary of the picture at right, if the border of the picture is co-located with the vertical split line resulting from a VER_LEFT splitting, i.e. if $\text{blkPosX} + \text{width}/4$ is equal to frameWidth , or if the width of the remaining part of the block after splitting is a multiple of the minimum block size. A flag `verLeftAllowed` is set to 1 to indicate that the corresponding split mode is allowed for the current block.

VER_RIGHT is allowed if the block does not extend beyond the boundary of the picture at bottom and, in the case where the block extends beyond the boundary of the picture at right, if the border of the picture is co-located with the vertical split line resulting from a VER_RIGHT splitting, i.e. if $\text{blkPosX} + 3*\text{width}/4$ is equal to frameWidth , or if the width of the remaining part of the block after splitting is a multiple

of the minimum block size. A flag `verRightAllowed` is set to 1 to indicate that the corresponding split mode is allowed for the current block.

In a second variant of S230, if a current block extends partly beyond the
5 boundary of the picture, the size of the remaining part of the current block is computed. It is then checked if transforms (e.g. DCT or DST) are available to transform the remaining part into a block of coefficients. If such transforms are available, the current block may be encoded without further splitting. In this case, the split mode `NO_SPLIT` is indicated as allowed by setting the flag `noSplitAllowed` equal to 1. In this latter case,
10 a flag is decoded to indicate whether the block is effectively not split or if the block is further split according to a split mode, e.g. as a quad tree. In a specific embodiment, the other split modes either binary or triple are disallowed for such a block. In a variant, all these split modes are allowed in addition to the quad-tree split mode.

As an example, for a current block of size 16x16 with a remaining part of size
15 4x16, the `NO_SPLIT` is allowed for the current block since a block of size 4x16 may be transformed with 1D transforms of size 4 and 16. In this second variant, the `NO_SPLIT` mode may thus be allowed even for a block that extends beyond the boundary of the picture.

In another example, for a current block of size 16x16 with a remaining part of
20 size 12x16, the `NO_SPLIT` is allowed for the current block only in the case where asymmetric splitting is enabled. Indeed, in this case the 12x16 block may be transformed by transforms of size 3×2^N . Such transforms of size 3×2^N are available in the case where asymmetric splitting is enabled. In the case where asymmetric splitting is disabled, only power of 2 transforms are available. In this latter case, the block 12x16
25 is necessarily split, i.e. the `NO_SPLIT` mode is thus disallowed. Asymmetric splitting may be disabled by a high level syntax element, e.g. for some profiles.

At step S250, the current block is decoded according to the current split mode. To this aim, each sub-block of the block obtained by splitting the current block according to the split mode is decoded.
30 Decoding a block, e.g. one of the sub-blocks of the current block, usually but not necessarily comprises entropy decoding a portion of the bitstream representative of the block to obtain a block of transform coefficients, de-quantizing and inverse transforming the block of transform coefficients to obtain a block of residuals and adding a predictor to the block of residuals to obtain a decoded block.

Figure 17 represents a flowchart of a method for encoding a block of a picture in a bitstream according to a specific and non-limiting embodiment.

The method starts at step S100. At step S110, a transmitter, e.g. such as the encoder 100, accesses a current block that includes at least a portion of a picture. At step S120, the transmitter determines whether the current block is partly outside the picture, i.e. extends beyond one boundary of the picture. This step is identical to the step S220 of the decoding method.

In the case where the current block is partly outside the picture, the method continues at step S130 otherwise the method continues at step S140. In this latter case, the block is completely inside the picture and all split modes may be allowed. At step S130, the transmitter determines for each split mode of a plurality of split modes whether the current block is allowed to undergo splitting according to this split mode by checking whether at least one of the split line is co-located with one of the picture border. This step is identical to the step S230 of the decoding method. All the embodiments and variants disclosed with respect to the decoding method also apply to the encoding method. At step S140, the transmitter determines a current split mode for the accessed block responsive to the allowed split modes and encodes, for the current block, the current split mode in a bitstream. The split mode may be selected among the allowed mode based on a rate-distortion criteria. Each allowed split mode may be tested and the one offering the best rate-distortion compromise is selected as the split mode to encode the current block.

At step S150, the transmitter encodes the accessed block into the bitstream according to the current split mode. To this aim, the current block may be split into several sub-blocks according the split mode. Encoding a block, e.g. a sub-block, usually but not necessarily comprises subtracting a predictor from the block to obtain a block of residuals, transforming the block of residuals into a block of transform coefficients, quantizing the block of coefficients with a quantization step size QP to obtain a quantized block of transform coefficients and entropy coding the quantized block of transform coefficients in the bitstream. Reconstructing a block on the encoder side usually but not necessarily comprises de-quantizing and inverse transforming the quantized block of transform coefficients to obtain a block of residuals and adding the predictor to the block of residuals to obtain a decoded block.

The method ends at step S180.

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

Implementations of the various processes and features described herein may be embodied in a variety of different equipment or applications, particularly, for example, equipment or applications. Examples of such equipment include an encoder, a decoder, a post-processor processing output from a decoder, a pre-processor providing input to an encoder, a video coder, a video decoder, a video codec, a web server, a set-top box, a laptop, a personal computer, a cell phone, a PDA, and other communication devices. As should be clear, the equipment may be mobile and even installed in a mobile vehicle.

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

process. Further, a processor-readable medium may store, in addition to or in lieu of instructions, data values produced by an implementation.

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

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

25

30

Claims

1. A decoding method for decoding a current block of a picture, comprising :
 - determining whether said current block extends beyond a boundary of the picture;
 - 5 - determining for each split mode of a plurality of split modes whether said split mode is allowed for said current block by:
 - allowing said split mode in the case where at least one of the split lines is co-located with one of the picture boundaries; and otherwise,
 - allowing said split mode in the case where a size of a block part inside the picture
 - 10 along the picture boundary is a multiple of a minimum block size;
 - decoding from a bitstream a current split mode of the current block responsive to the allowed split modes; and
 - decoding the current block according to the current split mode.
- 15 2. The decoding method according to claim 1, wherein h is the height of said current block and w is the width of said current block, and the plurality of split modes comprises at least two of the following split modes:
 - splitting said block horizontally into one upper sub-block of height $h/4$ and one bottom sub-block of height $3h/4$;
 - 20 - splitting said block horizontally into one bottom sub-block of height $h/4$ and one upper sub-block of height $3h/4$;
 - splitting said block vertically into one left sub-block of width $w/4$ and one right sub-block of width $3w/4$;
 - splitting said block vertically into one right sub-block of width $w/4$ and one left sub-
 - 25 block of width $3w/4$;
 - splitting said block horizontally into one sub-block of height $h/4$, one central sub-block of height $h/2$ and one sub-block of height $h/4$;
 - splitting said block vertically into one sub-block of width $w/4$, one central sub-block of width $w/2$ and one sub-block of width $w/4$;
 - 30 - splitting said block horizontally into two sub-blocks of height $h/2$; and
 - splitting said block vertically into two sub-blocks of width $w/2$.
3. The decoding method according to claim 1 or 2, wherein the minimum block size in at least one of height and width is equal to 4.

4. An encoding method for encoding a current block of a picture, comprising :
- determining whether said current block extends beyond a boundary of the picture;
 - determining, for each split mode of a plurality of split modes, whether said split mode
- 5 is allowed for said current block by:
- allowing said split mode in the case where at least one of the split lines is co-located with one of the picture boundaries; and otherwise,
 - allowing said split mode in the case where a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size;
- 10 - encoding in a bitstream a current split mode of the current block determined responsive to the allowed split modes; and
- encoding the current block in the bitstream according to the current split mode.
5. The encoding method according to claim 4, wherein h is the height of said current
- 15 block and w is the width of said current block, and wherein the plurality of split modes comprises at least two of the following split modes:
- splitting said block horizontally into one upper sub-block of height $h/4$ and one bottom sub-block of height $3h/4$;
 - splitting said block horizontally into one bottom sub-block of height $h/4$ and one
- 20 upper sub-block of height $3h/4$;
- splitting said block vertically into one left sub-block of width $w/4$ and one right sub-block of width $3w/4$;
 - splitting said block vertically into one right sub-block of width $w/4$ and one left sub-block of width $3w/4$;
- 25 - splitting said block horizontally into one sub-block of height $h/4$, one central sub-block of height $h/2$ and one sub-block of height $h/4$;
- splitting said block vertically into one sub-block of width $w/4$, one central sub-block of width $w/2$ and one sub-block of width $w/4$;
 - splitting said block horizontally into two sub-blocks of height $h/2$; and
- 30 - splitting said block vertically into two sub-blocks of width $w/2$.
6. The encoding method according to claim 4 or 5, wherein the minimum block size in at least one of height and width is equal to 4.

7. A decoding apparatus configured to decode a current block of a picture, comprising:
- means for determining whether said current block extends beyond a boundary of the picture;
 - means for determining for each split mode of a plurality of split modes whether said
- 5 split mode is allowed for said current block by:
- allowing said split mode in the case where at least one of the split lines is co-located with one of the picture boundaries; and otherwise,
 - allowing said split mode in the case where a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size;
- 10 - means for decoding from a bitstream a current split mode of the current block responsive to the allowed split modes; and
- means for decoding the current block according to the current split mode.
8. The decoding apparatus according to claim 7, wherein h is the height of said current
- 15 block and w is the width of said current block, and wherein the plurality of split modes comprises at least two of the following split modes:
- splitting said block horizontally into one upper sub-block of height $h/4$ and one bottom sub-block of height $3h/4$;
 - splitting said block horizontally into one bottom sub-block of height $h/4$ and one
- 20 upper sub-block of height $3h/4$;
- splitting said block vertically into one left sub-block of width $w/4$ and one right sub-block of width $3w/4$;
 - splitting said block vertically into one right sub-block of width $w/4$ and one left sub-block of width $3w/4$;
- 25 - splitting said block horizontally into one sub-block of height $h/4$, one central sub-block of height $h/2$ and one sub-block of height $h/4$;
- splitting said block vertically into one sub-block of width $w/4$, one central sub-block of width $w/2$ and one sub-block of width $w/4$;
 - splitting said block horizontally into two sub-blocks of height $h/2$; and
- 30 - splitting said block vertically into two sub-blocks of width $w/2$.
9. The decoding apparatus according to claim 7 or 8, wherein the minimum block size in at least one of height and width is equal to 4.

10. An encoding apparatus configured to encode a current block of a picture, comprising:

- means for determining whether said current block extends beyond a boundary of the picture;

5 - means for determining, for each split mode of a plurality of split modes, whether said split mode is allowed for said current block by:

- allowing said split mode in the case where at least one of the split lines is co-located with one of the picture boundaries; and otherwise,

10 - allowing said split mode in the case where a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size;

- means for encoding in a bitstream a current split mode of the current block determined responsive to the allowed split modes; and

- means for encoding the current block in the bitstream according to the current split mode.

15

11. The encoding apparatus according to claim 10, wherein h is the height of said current block and w is the width of said current block, and wherein the plurality of split modes comprises at least two of the following split modes:

- splitting said block horizontally into one upper sub-block of height $h/4$ and one
20 bottom sub-block of height $3h/4$;

- splitting said block horizontally into one bottom sub-block of height $h/4$ and one upper sub-block of height $3h/4$;

- splitting said block vertically into one left sub-block of width $w/4$ and one right sub-block of width $3w/4$;

25 - splitting said block vertically into one right sub-block of width $w/4$ and one left sub-block of width $3w/4$;

- splitting said block horizontally into one sub-block of height $h/4$, one central sub-block of height $h/2$ and one sub-block of height $h/4$;

30 - splitting said block vertically into one sub-block of width $w/4$, one central sub-block of width $w/2$ and one sub-block of width $w/4$;

- splitting said block horizontally into two sub-blocks of height $h/2$; and

- splitting said block vertically into two sub-blocks of width $w/2$.

12. The encoding apparatus according to claim 10 or 11, wherein the minimum block size in at least one of height and width is equal to 4.

13. A machine readable medium having stored thereon machine executable instructions
5 that, when executed, implement a method for decoding a current block of a picture, the method comprising:

- determining whether said current block extends beyond a boundary of the picture;
- determining for each split mode of a plurality of split modes whether said split mode is allowed for said current block by:
10 - allowing said split mode in the case where at least one of the split lines is co-located with one of the picture boundaries; and otherwise,
- allowing said split mode in the case where a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size;
- decoding from a bitstream a current split mode of the current block responsive to the
15 allowed split modes; and
- decoding the current block according to the current split mode.

14. A machine readable medium having stored thereon machine executable
instructions that, when executed, implement a method for encoding a current block of
20 a picture, the method comprising:

- determining whether said current block extends beyond a boundary of the picture;
- determining, for each split mode of a plurality of split modes, whether said split mode is allowed for said current block by:
25 - allowing said split mode in the case where at least one of the split lines is co-located with one of the picture boundaries; and otherwise,
- allowing said split mode in the case where a size of a block part inside the picture along the picture boundary is a multiple of a minimum block size;
- encoding in a bitstream a current split mode of the current block determined responsive to the allowed split modes; and
30 - encoding the current block in the bitstream according to the current split mode.

15. A bitstream formatted to include encoded data representative of a block of a picture, the encoded data encoded according to any of claims 4-6.

1/9

CTU

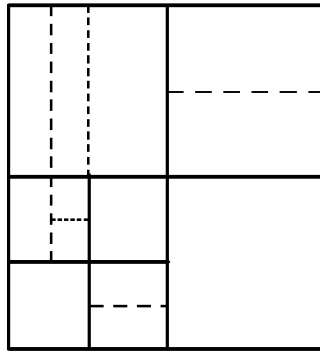


Figure 1 – Prior Art

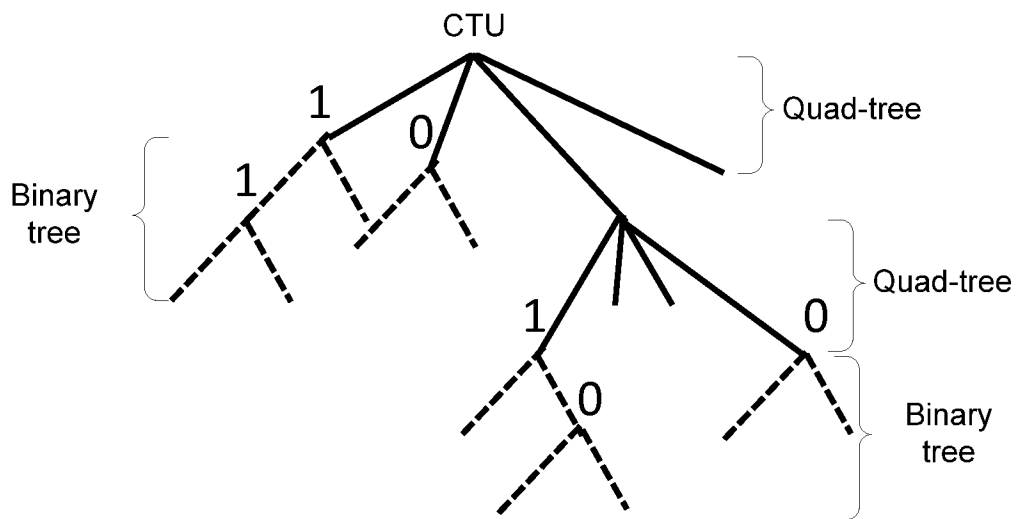


Figure 2 – Prior Art

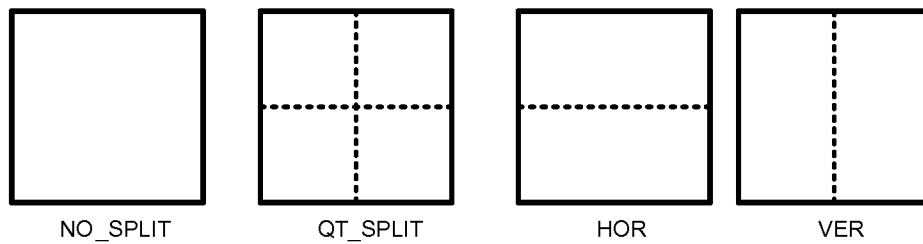


Figure 3 – Prior Art

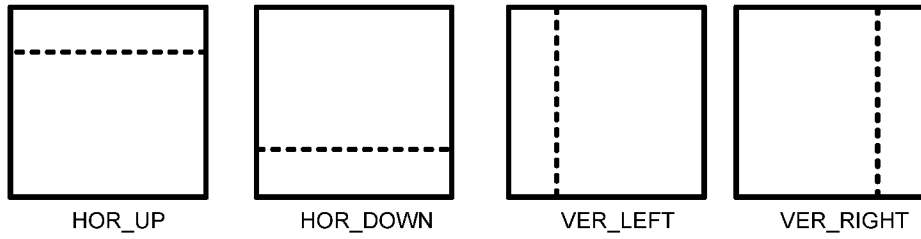


Figure 4 – Prior Art

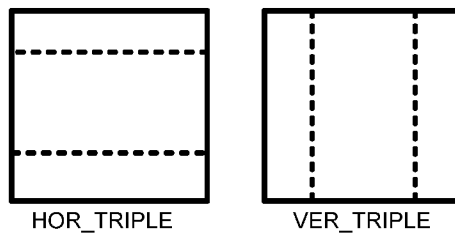


Figure 5 – Prior Art

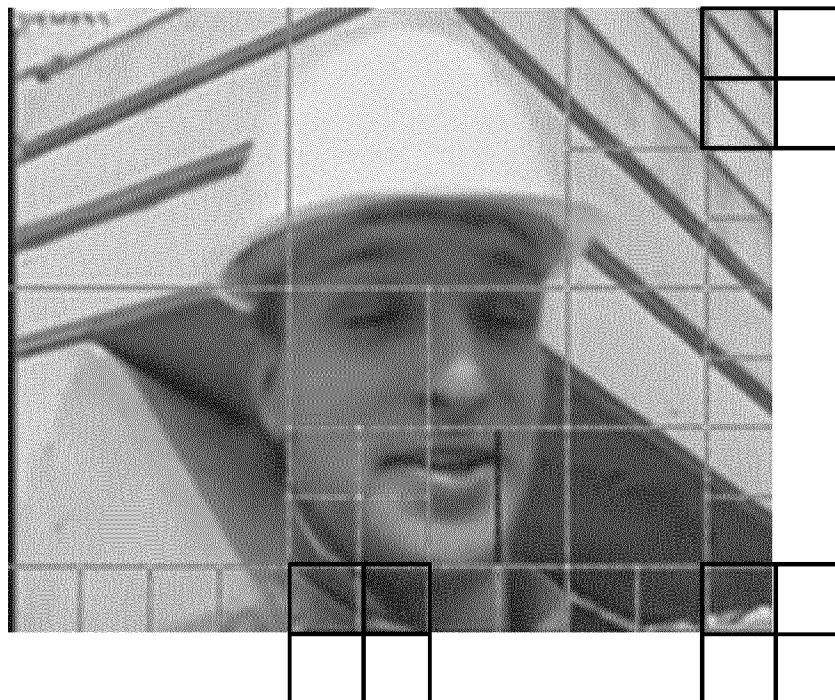


Figure 6

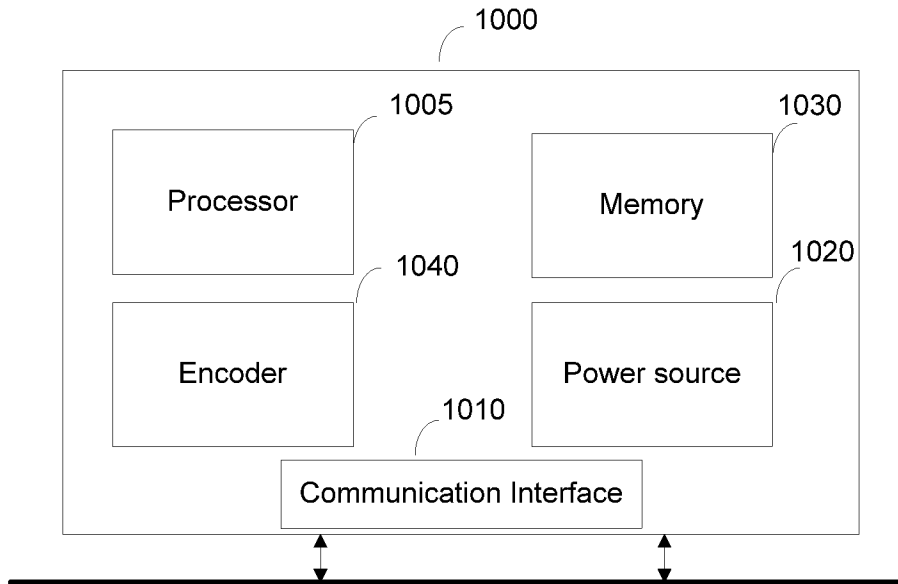


FIGURE 7

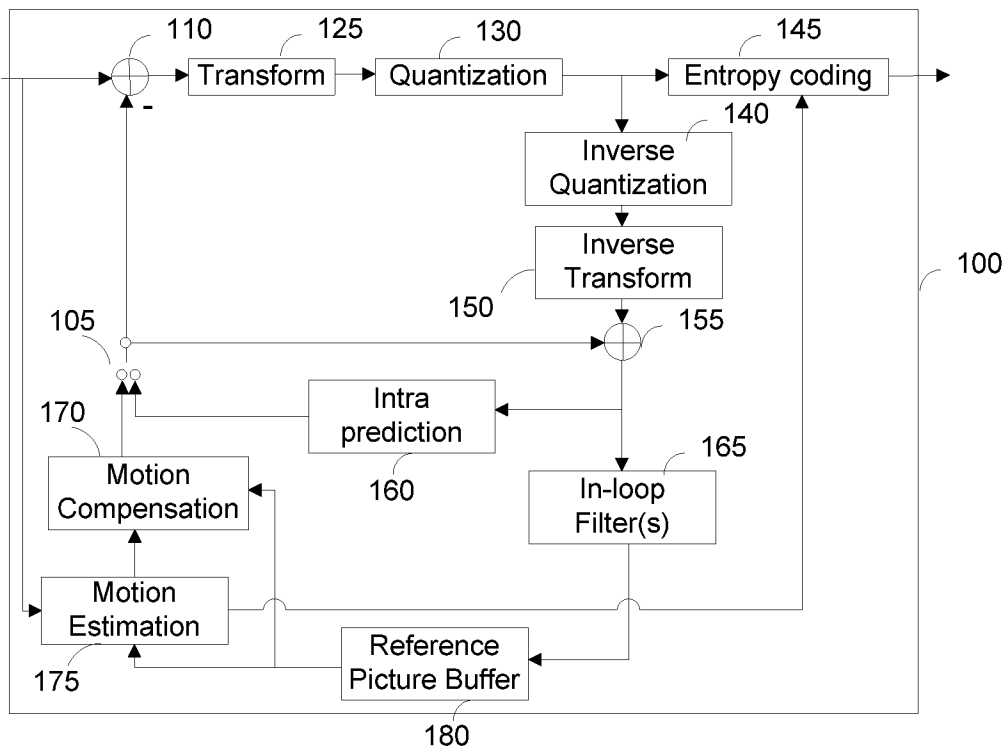


FIGURE 8

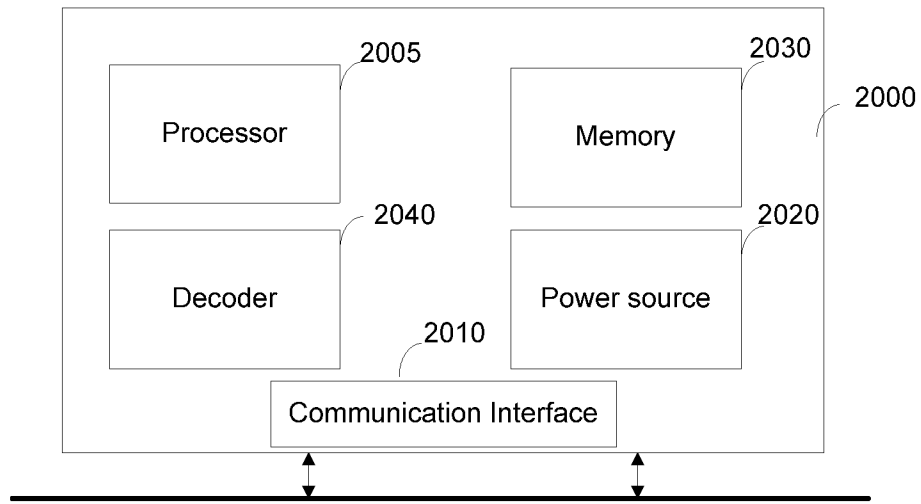


FIGURE 9

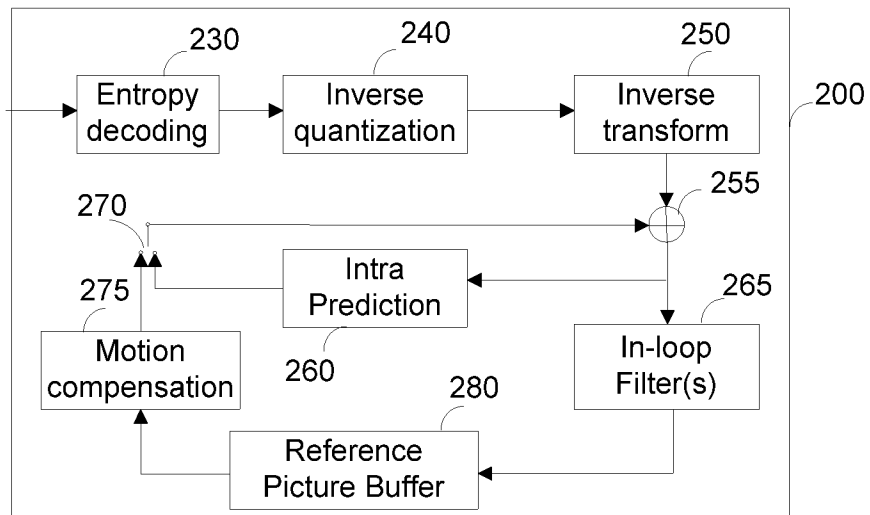


FIGURE 10

5/9

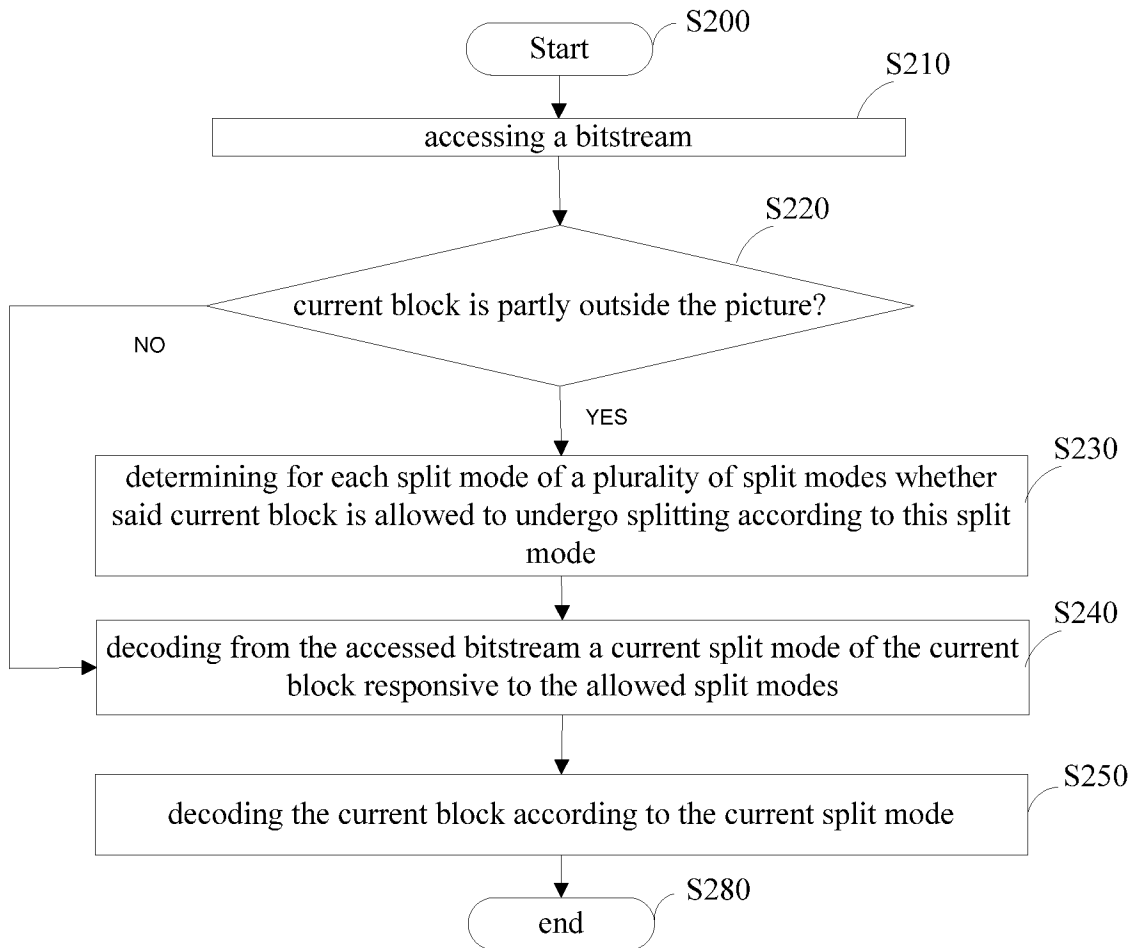


FIGURE 11

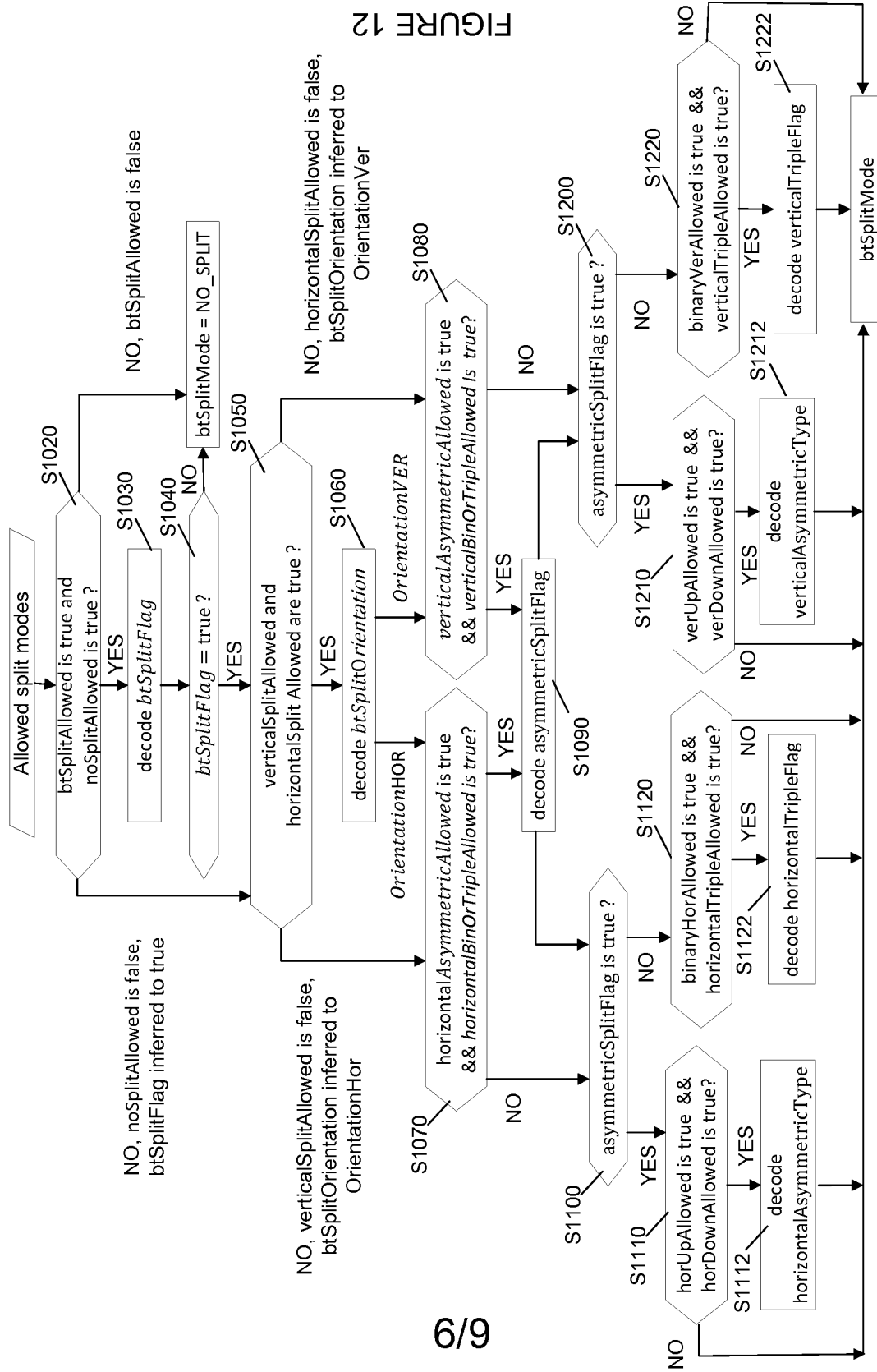


FIGURE 12

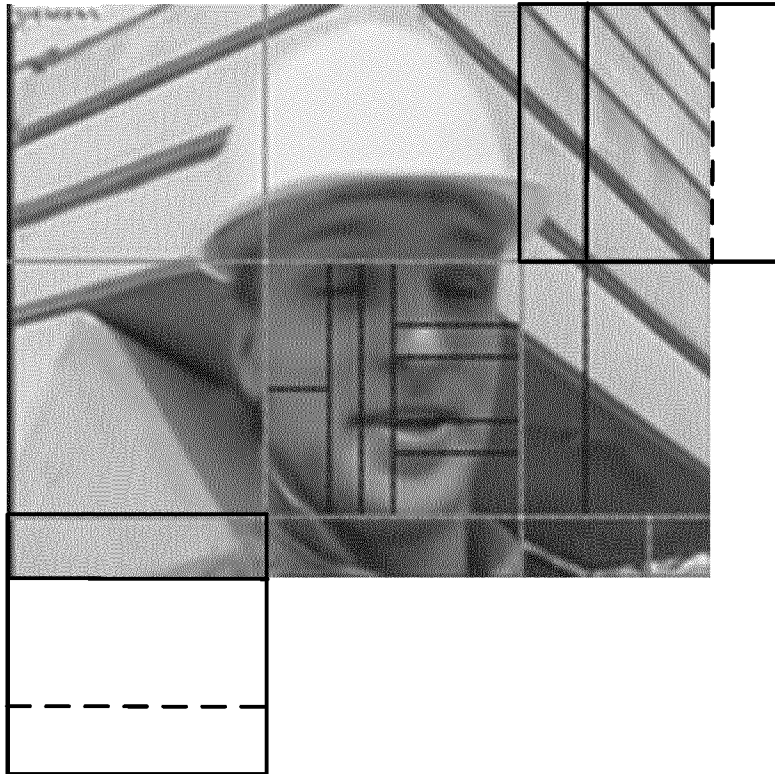


Figure 13

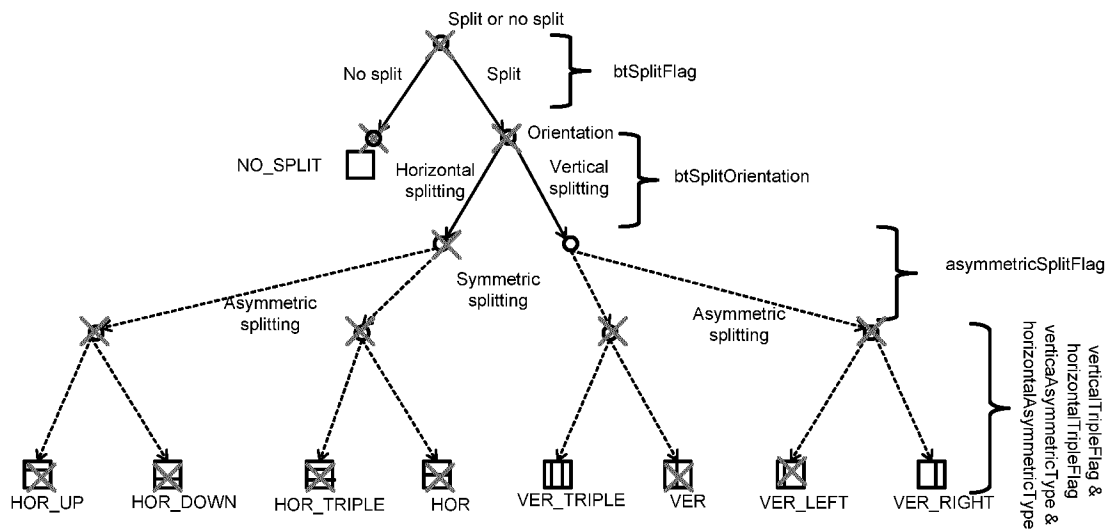


Figure 14

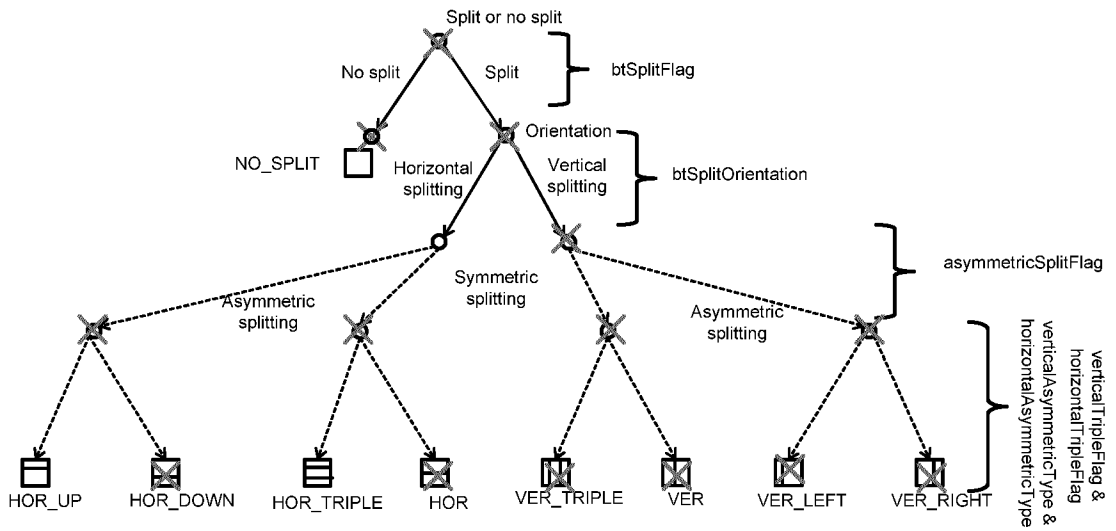


Figure 15

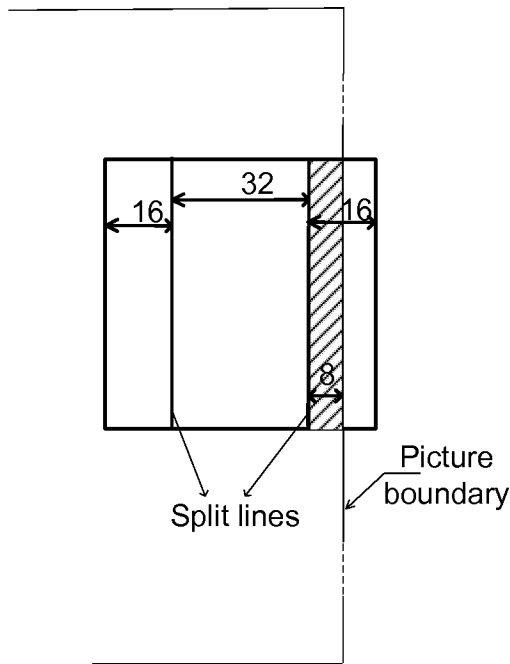


FIGURE 16

9/9

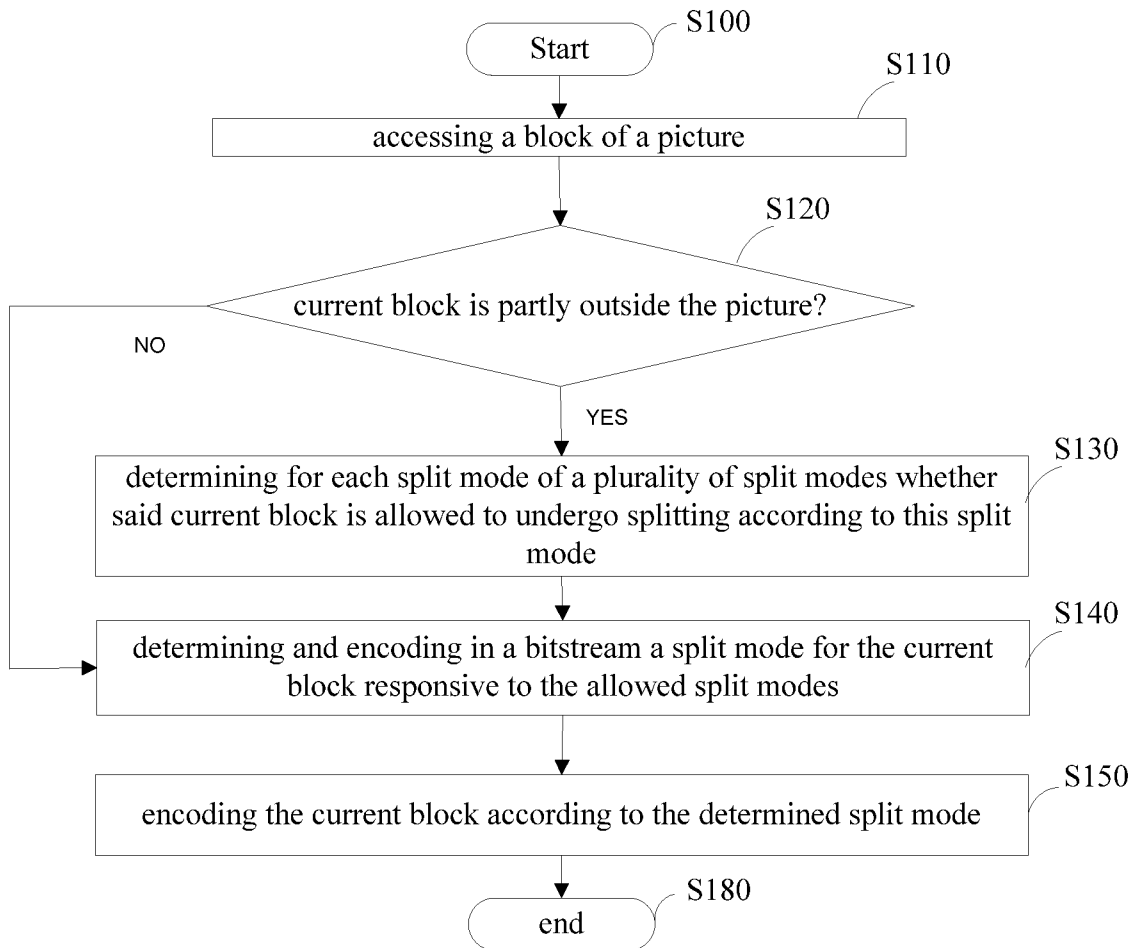


FIGURE 17

INTERNATIONAL SEARCH REPORT

International application No PCT/EP2018/056254

A. CLASSIFICATION OF SUBJECT MATTER INV. H04N19/176 H04N19/70 H04N19/119 H04N19/167 ADD.				
According to International Patent Classification (IPC) or to both national classification and IPC				
B. FIELDS SEARCHED				
Minimum documentation searched (classification system followed by classification symbols) H04N				
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched				
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal				
C. DOCUMENTS CONSIDERED TO BE RELEVANT				
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.		
X	EP 3 059 708 A1 (SAMSUNG ELECTRONICS CO LTD [KR]) 24 August 2016 (2016-08-24) abstract paragraphs [0069], [0088], [0090], [0095], [0123] - [0133], [0136], [0149] - [0158] figures 9, 16, 18a-18g <div style="text-align: center; margin-top: 10px;">----- -/--</div>	1-15		
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.				
* Special categories of cited documents : <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none; vertical-align: top;"> "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 </td> <td style="width: 50%; border: none; vertical-align: top;"> "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 </td> </tr> </table>			"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
"A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family			
Date of the actual completion of the international search	Date of mailing of the international search report			
16 May 2018	28/05/2018			
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Santos Luque, Rocio			

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2018/056254

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>YANG F ET AL: "CTB splitting on frame boundary for arbitrary resolution video", 3. JCT-VC MEETING; 94. MPEG MEETING; 7-10-2010 - 15-10-2010; GUANGZHOU; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16), no. JCTVC-C025, 1 October 2010 (2010-10-01), XP030007732, ISSN: 0000-0045 abstract sections 2 and 3</p>	<p>1,4,7, 10,13,14</p>
A	<p>----- WO 2016/209757 A1 (CISCO TECH INC [US]) 29 December 2016 (2016-12-29) paragraphs [0021] - [0030]</p>	<p>1,4,7, 10,13,14</p>
A	<p>----- LE LÉANNEC F, ET AL.: "Asymmetric Coding Units in QTBT", JVET-D0064, JOINT VIDEO EXPLORATION TEAM (JVET) OF ITU-T SG16 WP3 AND ISO/IEC JTC1/SC29/WG11, 15 October 2016 (2016-10-15), pages 1-8, XP030150297, cited in the application the whole document</p>	<p>1-15</p>
A	<p>----- LI X, ET AL.: "Multi-Type-Tree", JVET-D0117-RL, JOINT VIDEO EXPLORATION TEAM (JVET) OF ITU-T SG16 WP3 AND ISO/IEC JTC1/SC29/WG1 1, 15 October 2016 (2016-10-15), XP030150367, cited in the application the whole document -----</p>	<p>1-15</p>

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/EP2018/056254

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 3059708	A1	24-08-2016	
		AU 2010313997 A1	17-05-2012
		BR 112012009766 A2	17-05-2016
		CA 2778534 A1	05-05-2011
		CA 2818572 A1	05-05-2011
		CA 2818587 A1	05-05-2011
		CA 2888395 A1	05-05-2011
		CA 2888396 A1	05-05-2011
		CN 102640492 A	15-08-2012
		CN 103220521 A	24-07-2013
		CN 103220526 A	24-07-2013
		CN 104581160 A	29-04-2015
		CN 104581164 A	29-04-2015
		CN 104602010 A	06-05-2015
		CN 104796709 A	22-07-2015
		CN 104796710 A	22-07-2015
		EP 2471262 A2	04-07-2012
		EP 2629524 A1	21-08-2013
		EP 2629525 A1	21-08-2013
		EP 2739057 A1	04-06-2014
		EP 2741508 A1	11-06-2014
		EP 2741509 A1	11-06-2014
		EP 2755394 A1	16-07-2014
		EP 3059708 A1	24-08-2016
		EP 3249611 A1	29-11-2017
		JP 5639687 B2	10-12-2014
		JP 5687310 B2	18-03-2015
		JP 5711248 B2	30-04-2015
		JP 5727114 B1	03-06-2015
		JP 5727115 B1	03-06-2015
		JP 5913677 B2	27-04-2016
		JP 2013214990 A	17-10-2013
		JP 2013214991 A	17-10-2013
		JP 2013509788 A	14-03-2013
		JP 2015111894 A	18-06-2015
		JP 2015111895 A	18-06-2015
		JP 2015144486 A	06-08-2015
		KR 20110047697 A	09-05-2011
		MY 163448 A	15-09-2017
		RU 2013114238 A	27-03-2014
		RU 2013114276 A	27-02-2014
		RU 2013119372 A	27-10-2014
		RU 2013119397 A	27-10-2014
		RU 2013119400 A	27-10-2014
		RU 2013152955 A	10-06-2015
		US 2011103701 A1	05-05-2011
		US 2012114261 A1	10-05-2012
		US 2012243799 A1	27-09-2012
		US 2013083861 A1	04-04-2013
		US 2014064367 A1	06-03-2014
		US 2014064629 A1	06-03-2014
		US 2014072238 A1	13-03-2014
		US 2014294084 A1	02-10-2014
		WO 2011053050 A2	05-05-2011
		ZA 201202670 B	25-11-2015
		ZA 201304978 B	25-09-2013
		ZA 201304979 B	25-09-2013
		ZA 201406113 B	30-09-2015
		ZA 201500338 B	29-04-2015

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/EP2018/056254

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		ZA 201500339 B	29-04-2015
-----		-----	-----
WO 2016209757 A1	29-12-2016	US 2016373766 A1	22-12-2016
		WO 2016209757 A1	29-12-2016
		-----	-----