



(51) International Patent Classification:

H04N 19/119 (2014.01) H04N 19/169 (2014.01)

H04N 19/159 (2014.01)

(21) International Application Number:

PCT/CN2022/122197

(22) International Filing Date:

28 September 2022 (28.09.2022)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

PCT/CN2021/121968

29 September 2021 (29.09.2021) CN

(71) Applicants: BEIJING BYTEDANCE NETWORK

TECHNOLOGY CO., LTD. [CN/CN]; Room B-0035, 2/F, No. 3 Building, No.30, Shixing Road, Shijingshan District, Beijing 100041 (CN). BYTEDANCE INC. [US/US]; 12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US).

(72) Inventors: DENG, Zhipin; Jinritoutiao Post Office, Chi-

na Satellite Communications Tower, No. 63, Zhichun Road, Haidian District, Beijing 100080 (CN). ZHANG, Kai; 12655 West Jefferson Boulevard Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US). ZHANG, Li; 12655

West Jefferson Boulevard Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US).

(74) Agent: SHIHUI PARTNERS; 42/F, Tower C, Beijing

Yintai Centre, No. 2 Jianguomenwai Avenue, Chaoyang District, Beijing 100022 (CN).

(81) Designated States (unless otherwise indicated, for every

kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, 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, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every

kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

(54) Title: METHOD, APPARATUS, AND MEDIUM FOR VIDEO PROCESSING

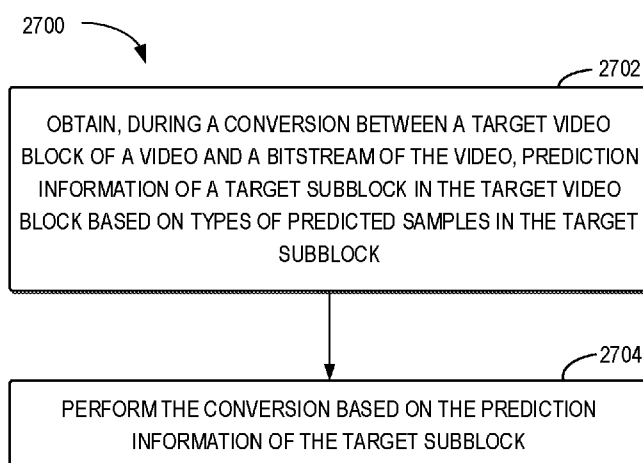


Fig. 27

(57) Abstract: Embodiments of the present disclosure provide a solution for video processing. A method for video processing is proposed. The method comprises: obtaining, during a conversion between a target video block of a video and a bitstream of the video, prediction information of a target subblock in the target video block based on types of predicted samples in the target subblock, the target video block being coded by a geometric partitioning tool; and performing the conversion based on the prediction information of the target subblock.

TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
KM, ML, MR, NE, SN, TD, TG).

Published:

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

METHOD, APPARATUS, AND MEDIUM FOR VIDEO PROCESSING

FIELD

[0001] Embodiments of the present disclosure relates generally to video coding techniques, and more particularly, to prediction for geometric partitioning coded block.

5 BACKGROUND

[0002] In nowadays, digital video capabilities are being applied in various aspects of peoples' lives. Multiple types of video compression technologies, such as MPEG-2, MPEG-4, ITU-TH.263, ITU-TH.264/MPEG-4 Part 10 Advanced Video Coding (AVC), ITU-TH.265 high efficiency video coding (HEVC) standard, versatile video coding (VVC) standard, have been
10 proposed for video encoding/decoding. However, coding efficiency of conventional video coding techniques is generally very low, which is undesirable.

SUMMARY

[0003] Embodiments of the present disclosure provide a solution for video processing.

[0004] In a first aspect, a method for video processing is proposed. The method comprises:
15 obtaining, during a conversion between a target video block of a video and a bitstream of the video, prediction information of a target subblock in the target video block based on types of predicted samples in the target subblock, the target video block being coded by a geometric partitioning tool; and performing the conversion based on the prediction information of the target subblock. Compared with the conventional solution, the proposed method can advantageously
20 improve the coding effectiveness and coding efficiency.

[0005] In a second aspect, an apparatus for processing video data is proposed. The apparatus for processing video data comprises a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to perform a method in accordance with the first aspect of the present disclosure.

25 [0006] In a third aspect, a non-transitory computer-readable recording medium is proposed. The non-transitory computer-readable recording medium stores a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises: obtaining prediction information of a target subblock in a target video block of the video based

on types of predicted samples in the target subblock, the target video block being coded by a geometric partitioning tool; and generating the bitstream based on the prediction information of the target subblock.

5 [0007] In a fourth aspect, another method for storing a bitstream of a video is proposed. The method comprises: obtaining prediction information of a target subblock in a target video block of the video based on types of predicted samples in the target subblock, the target video block being coded by a geometric partitioning tool; generating the bitstream based on the prediction information of the target subblock; and storing the bitstream in a non-transitory computer-readable recording medium.

10 [0008] This Summary is provided to introduce a selection of concepts in a simplified form that are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used to limit the scope of the claimed subject matter.

BRIEF DESCRIPTION OF THE DRAWINGS

15 [0009] Through the following detailed description with reference to the accompanying drawings, the above and other objectives, features, and advantages of example embodiments of the present disclosure will become more apparent. In the example embodiments of the present disclosure, the same reference numerals usually refer to the same components.

20 [0010] Fig. 1 illustrates a block diagram that illustrates an example video coding system, in accordance with some embodiments of the present disclosure;

[0011] Fig. 2 illustrates a block diagram that illustrates a first example video encoder, in accordance with some embodiments of the present disclosure;

[0012] Fig. 3 illustrates a block diagram that illustrates an example video decoder, in accordance with some embodiments of the present disclosure;

25 [0013] Fig. 4 illustrates a schematic diagram of intra prediction modes;

[0014] Fig. 5A illustrates a schematic diagram of top references;

[0015] Fig. 5B illustrates a schematic diagram of left references;

[0016] Fig. 6 illustrates a schematic diagram of discontinuity in case of directions beyond 45° ;

[0017] Fig. 7A illustrates a schematic diagram of the definition of samples used by PDPC applied to diagonal top-right intra mode;

[0018] Fig. 7B illustrates a schematic diagram of the definition of samples used by PDPC applied to diagonal bottom-left intra mode;

5 [0019] Fig. 7C illustrates a schematic diagram of the definition of samples used by PDPC applied to adjacent diagonal top-right intra mode;

[0020] Fig. 7D illustrates a schematic diagram of the definition of samples used by PDPC applied to adjacent diagonal bottom-left intra mode;

10 [0021] Fig. 8 illustrates example diagram of four reference lines neighboring to a prediction block;

[0022] Figs. 9A and 9B illustrate examples of sub-partitions;

[0023] Fig. 10 illustrates a schematic diagram of matrix weighted intra prediction process;

[0024] Fig. 11 illustrates a schematic diagram of positions of spatial merge candidates;

15 [0025] Fig. 12 illustrates a schematic diagram of candidate pairs considered for redundancy check of spatial merge candidates;

[0026] Fig. 13 illustrates a schematic diagram of motion vector scaling for temporal merge candidate;

[0027] Fig. 14 illustrates a schematic diagram of candidate positions for temporal merge candidates;

20 [0028] Fig. 15A and Fig. 15B illustrate schematic diagrams of MMVD Search Point;

[0029] Fig. 16 illustrates a schematic diagram of an extended CU region used in BDOF;

[0030] Fig. 17 illustrates a schematic diagram of an illustration for symmetrical MVD mode;

[0031] Fig. 18 illustrates a decoding side motion vector refinement;

25 [0032] Fig. 19 illustrates a schematic diagram of top and left neighboring blocks used in CIIP weight derivation;

[0033] Fig. 20 illustrates a schematic diagram of examples of the GPM splits grouped by identical angles;

[0034] Fig. 21 illustrates a schematic diagram of uni-prediction MV selection for geometric partitioning mode;

[0035] Fig. 22 illustrates a schematic diagram of exemplified generation of a bending weight w_0 using geometric partitioning mode;

5 [0036] Fig. 23 illustrates a schematic diagram of a proposed intra block decoding process;

[0037] Fig. 24 illustrates a schematic diagram of HoG computation from a template;

[0038] Fig. 25 illustrates a schematic process of prediction fusion by weighted averaging of two HoG modes and planar;

10 [0039] Fig. 26 illustrates an example of subblock based motion/mode information storage of a GPM coded block;

[0040] Fig. 27 illustrates a flowchart of a method for video processing in accordance with some embodiments of the present disclosure; and

[0041] Fig. 28 illustrates a block diagram of a computing device in which various embodiments of the present disclosure can be implemented.

15 [0042] Throughout the drawings, the same or similar reference numerals usually refer to the same or similar elements.

DETAILED DESCRIPTION

20 [0043] Principle of the present disclosure will now be described with reference to some embodiments. It is to be understood that these embodiments are described only for the purpose of illustration and help those skilled in the art to understand and implement the present disclosure, without suggesting any limitation as to the scope of the disclosure. The disclosure described herein can be implemented in various manners other than the ones described below.

25 [0044] In the following description and claims, unless defined otherwise, all technical and scientific terms used herein have the same meaning as commonly understood by one of ordinary skills in the art to which this disclosure belongs.

[0045] References in the present disclosure to “one embodiment,” “an embodiment,” “an example embodiment,” and the like indicate that the embodiment described may include a particular feature, structure, or characteristic, but it is not necessary that every embodiment

includes the particular feature, structure, or characteristic. Moreover, such phrases are not necessarily referring to the same embodiment. Further, when a particular feature, structure, or characteristic is described in connection with an example embodiment, it is submitted that it is within the knowledge of one skilled in the art to affect such feature, structure, or characteristic in connection with other embodiments whether or not explicitly described.

[0046] It shall be understood that although the terms “first” and “second” etc. 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. For example, a first element could be termed a second element, and similarly, a second element could be termed a first element, without departing from the scope of example embodiments. As used herein, the term “and/or” includes any and all combinations of one or more of the listed terms.

[0047] The terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting of example embodiments. As used herein, the singular forms “a”, “an” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will be further understood that the terms “comprises”, “comprising”, “has”, “having”, “includes” and/or “including”, when used herein, specify the presence of stated features, elements, and/or components etc., but do not preclude the presence or addition of one or more other features, elements, components and/or combinations thereof.

Example Environment

[0048] Fig. 1 is a block diagram that illustrates an example video coding system 100 that may utilize the techniques of this disclosure. As shown, the video coding system 100 may include a source device 110 and a destination device 120. The source device 110 can be also referred to as a video encoding device, and the destination device 120 can be also referred to as a video decoding device. In operation, the source device 110 can be configured to generate encoded video data and the destination device 120 can be configured to decode the encoded video data generated by the source device 110. The source device 110 may include a video source 112, a video encoder 114, and an input/output (I/O) interface 116.

[0049] The video source 112 may include a source such as a video capture device. Examples of the video capture device include, but are not limited to, an interface to receive video data from a video content provider, a computer graphics system for generating video data, and/or a combination thereof.

[0050] The video data may comprise one or more pictures. The video encoder 114 encodes the video data from the video source 112 to generate a bitstream. The bitstream may include a sequence of bits that form a coded representation of the video data. The bitstream may include coded pictures and associated data. The coded picture is a coded representation of a picture. The associated data may include sequence parameter sets, picture parameter sets, and other syntax structures. The I/O interface 116 may include a modulator/demodulator and/or a transmitter. The encoded video data may be transmitted directly to destination device 120 via the I/O interface 116 through the network 130A. The encoded video data may also be stored onto a storage medium/server 130B for access by destination device 120.

[0051] The destination device 120 may include an I/O interface 126, a video decoder 124, and a display device 122. The I/O interface 126 may include a receiver and/or a modem. The I/O interface 126 may acquire encoded video data from the source device 110 or the storage medium/server 130B. The video decoder 124 may decode the encoded video data. The display device 122 may display the decoded video data to a user. The display device 122 may be integrated with the destination device 120, or may be external to the destination device 120 which is configured to interface with an external display device.

[0052] The video encoder 114 and the video decoder 124 may operate according to a video compression standard, such as the High Efficiency Video Coding (HEVC) standard, Versatile Video Coding (VVC) standard and other current and/or further standards.

[0053] Fig. 2 is a block diagram illustrating an example of a video encoder 200, which may be an example of the video encoder 114 in the system 100 illustrated in Fig. 1, in accordance with some embodiments of the present disclosure.

[0054] The video encoder 200 may be configured to implement any or all of the techniques of this disclosure. In the example of Fig. 2, the video encoder 200 includes a plurality of functional components. The techniques described in this disclosure may be shared among the various components of the video encoder 200. In some examples, a processor may be configured to perform any or all of the techniques described in this disclosure.

[0055] In some embodiments, the video encoder 200 may include a partition unit 201, a predication unit 202 which may include a mode select unit 203, a motion estimation unit 204, a motion compensation unit 205 and an intra-prediction unit 206, a residual generation unit 207, a

transform unit 208, a quantization unit 209, an inverse quantization unit 210, an inverse transform unit 211, a reconstruction unit 212, a buffer 213, and an entropy encoding unit 214.

[0056] In other examples, the video encoder 200 may include more, fewer, or different functional components. In an example, the predication unit 202 may include an intra block copy (IBC) unit. The IBC unit may perform predication in an IBC mode in which at least one reference picture is a picture where the current video block is located.

[0057] Furthermore, although some components, such as the motion estimation unit 204 and the motion compensation unit 205, may be integrated, but are represented in the example of Fig. 2 separately for purposes of explanation.

[0058] The partition unit 201 may partition a picture into one or more video blocks. The video encoder 200 and the video decoder 300 may support various video block sizes.

[0059] The mode select unit 203 may select one of the coding modes, intra or inter, e.g., based on error results, and provide the resulting intra-coded or inter-coded block to a residual generation unit 207 to generate residual block data and to a reconstruction unit 212 to reconstruct the encoded block for use as a reference picture. In some examples, the mode select unit 203 may select a combination of intra and inter predication (CIIP) mode in which the predication is based on an inter predication signal and an intra predication signal. The mode select unit 203 may also select a resolution for a motion vector (e.g., a sub-pixel or integer pixel precision) for the block in the case of inter-predication.

[0060] To perform inter prediction on a current video block, the motion estimation unit 204 may generate motion information for the current video block by comparing one or more reference frames from buffer 213 to the current video block. The motion compensation unit 205 may determine a predicted video block for the current video block based on the motion information and decoded samples of pictures from the buffer 213 other than the picture associated with the current video block.

[0061] The motion estimation unit 204 and the motion compensation unit 205 may perform different operations for a current video block, for example, depending on whether the current video block is in an I-slice, a P-slice, or a B-slice. As used herein, an “I-slice” may refer to a portion of a picture composed of macroblocks, all of which are based upon macroblocks within the same picture. Further, as used herein, in some aspects, “P-slices” and “B-slices” may refer to

portions of a picture composed of macroblocks that are not dependent on macroblocks in the same picture.

[0062] In some examples, the motion estimation unit 204 may perform uni-directional prediction for the current video block, and the motion estimation unit 204 may search reference pictures of list 0 or list 1 for a reference video block for the current video block. The motion estimation unit 204 may then generate a reference index that indicates the reference picture in list 0 or list 1 that contains the reference video block and a motion vector that indicates a spatial displacement between the current video block and the reference video block. The motion estimation unit 204 may output the reference index, a prediction direction indicator, and the motion vector as the motion information of the current video block. The motion compensation unit 205 may generate the predicted video block of the current video block based on the reference video block indicated by the motion information of the current video block.

[0063] Alternatively, in other examples, the motion estimation unit 204 may perform bi-directional prediction for the current video block. The motion estimation unit 204 may search the reference pictures in list 0 for a reference video block for the current video block and may also search the reference pictures in list 1 for another reference video block for the current video block. The motion estimation unit 204 may then generate reference indexes that indicate the reference pictures in list 0 and list 1 containing the reference video blocks and motion vectors that indicate spatial displacements between the reference video blocks and the current video block. The motion estimation unit 204 may output the reference indexes and the motion vectors of the current video block as the motion information of the current video block. The motion compensation unit 205 may generate the predicted video block of the current video block based on the reference video blocks indicated by the motion information of the current video block.

[0064] In some examples, the motion estimation unit 204 may output a full set of motion information for decoding processing of a decoder. Alternatively, in some embodiments, the motion estimation unit 204 may signal the motion information of the current video block with reference to the motion information of another video block. For example, the motion estimation unit 204 may determine that the motion information of the current video block is sufficiently similar to the motion information of a neighboring video block.

[0065] In one example, the motion estimation unit 204 may indicate, in a syntax structure associated with the current video block, a value that indicates to the video decoder 300 that the current video block has the same motion information as the another video block.

[0066] In another example, the motion estimation unit 204 may identify, in a syntax structure associated with the current video block, another video block and a motion vector difference (MVD). The motion vector difference indicates a difference between the motion vector of the current video block and the motion vector of the indicated video block. The video decoder 300
5 may use the motion vector of the indicated video block and the motion vector difference to determine the motion vector of the current video block.

[0067] As discussed above, video encoder 200 may predictively signal the motion vector. Two examples of predictive signaling techniques that may be implemented by video encoder 200 include advanced motion vector predication (AMVP) and merge mode signaling.

10 [0068] The intra prediction unit 206 may perform intra prediction on the current video block. When the intra prediction unit 206 performs intra prediction on the current video block, the intra prediction unit 206 may generate prediction data for the current video block based on decoded samples of other video blocks in the same picture. The prediction data for the current video block may include a predicted video block and various syntax elements.

15 [0069] The residual generation unit 207 may generate residual data for the current video block by subtracting (e.g., indicated by the minus sign) the predicted video block (s) of the current video block from the current video block. The residual data of the current video block may include residual video blocks that correspond to different sample components of the samples in the current video block.

20 [0070] In other examples, there may be no residual data for the current video block for the current video block, for example in a skip mode, and the residual generation unit 207 may not perform the subtracting operation.

25 [0071] The transform processing unit 208 may generate one or more transform coefficient video blocks for the current video block by applying one or more transforms to a residual video block associated with the current video block.

[0072] After the transform processing unit 208 generates a transform coefficient video block associated with the current video block, the quantization unit 209 may quantize the transform coefficient video block associated with the current video block based on one or more quantization parameter (QP) values associated with the current video block.

[0073] The inverse quantization unit 210 and the inverse transform unit 211 may apply inverse quantization and inverse transforms to the transform coefficient video block, respectively, to reconstruct a residual video block from the transform coefficient video block. The reconstruction unit 212 may add the reconstructed residual video block to corresponding samples from one or more predicted video blocks generated by the predication unit 202 to produce a reconstructed video block associated with the current video block for storage in the buffer 213.

[0074] After the reconstruction unit 212 reconstructs the video block, loop filtering operation may be performed to reduce video blocking artifacts in the video block.

[0075] The entropy encoding unit 214 may receive data from other functional components of the video encoder 200. When the entropy encoding unit 214 receives the data, the entropy encoding unit 214 may perform one or more entropy encoding operations to generate entropy encoded data and output a bitstream that includes the entropy encoded data.

[0076] Fig. 3 is a block diagram illustrating an example of a video decoder 300, which may be an example of the video decoder 124 in the system 100 illustrated in Fig. 1, in accordance with some embodiments of the present disclosure.

[0077] The video decoder 300 may be configured to perform any or all of the techniques of this disclosure. In the example of Fig. 3, the video decoder 300 includes a plurality of functional components. The techniques described in this disclosure may be shared among the various components of the video decoder 300. In some examples, a processor may be configured to perform any or all of the techniques described in this disclosure.

[0078] In the example of Fig. 3, the video decoder 300 includes an entropy decoding unit 301, a motion compensation unit 302, an intra prediction unit 303, an inverse quantization unit 304, an inverse transformation unit 305, and a reconstruction unit 306 and a buffer 307. The video decoder 300 may, in some examples, perform a decoding pass generally reciprocal to the encoding pass described with respect to video encoder 200.

[0079] The entropy decoding unit 301 may retrieve an encoded bitstream. The encoded bitstream may include entropy coded video data (e.g., encoded blocks of video data). The entropy decoding unit 301 may decode the entropy coded video data, and from the entropy decoded video data, the motion compensation unit 302 may determine motion information including motion vectors, motion vector precision, reference picture list indexes, and other motion information. The motion compensation unit 302 may, for example, determine such information by performing

the AMVP and merge mode. AMVP is used, including derivation of several most probable candidates based on data from adjacent PBs and the reference picture. Motion information typically includes the horizontal and vertical motion vector displacement values, one or two reference picture indices, and, in the case of prediction regions in B slices, an identification of which reference picture list is associated with each index. As used herein, in some aspects, a “merge mode” may refer to deriving the motion information from spatially or temporally neighboring blocks.

[0080] The motion compensation unit 302 may produce motion compensated blocks, possibly performing interpolation based on interpolation filters. Identifiers for interpolation filters to be used with sub-pixel precision may be included in the syntax elements.

[0081] The motion compensation unit 302 may use the interpolation filters as used by the video encoder 200 during encoding of the video block to calculate interpolated values for sub-integer pixels of a reference block. The motion compensation unit 302 may determine the interpolation filters used by the video encoder 200 according to the received syntax information and use the interpolation filters to produce predictive blocks.

[0082] The motion compensation unit 302 may use at least part of the syntax information to determine sizes of blocks used to encode frame(s) and/or slice(s) of the encoded video sequence, partition information that describes how each macroblock of a picture of the encoded video sequence is partitioned, modes indicating how each partition is encoded, one or more reference frames (and reference frame lists) for each inter-encoded block, and other information to decode the encoded video sequence. As used herein, in some aspects, a “slice” may refer to a data structure that can be decoded independently from other slices of the same picture, in terms of entropy coding, signal prediction, and residual signal reconstruction. A slice can either be an entire picture or a region of a picture.

[0083] The intra prediction unit 303 may use intra prediction modes for example received in the bitstream to form a prediction block from spatially adjacent blocks. The inverse quantization unit 304 inverse quantizes, i.e., de-quantizes, the quantized video block coefficients provided in the bitstream and decoded by entropy decoding unit 301. The inverse transform unit 305 applies an inverse transform.

[0084] The reconstruction unit 306 may obtain the decoded blocks, e.g., by summing the residual blocks with the corresponding prediction blocks generated by the motion compensation

unit 302 or intra-prediction unit 303. If desired, a deblocking filter may also be applied to filter the decoded blocks in order to remove blockiness artifacts. The decoded video blocks are then stored in the buffer 307, which provides reference blocks for subsequent motion compensation/intra predication and also produces decoded video for presentation on a display device.

[0085] Some exemplary embodiments of the present disclosure will be described in detailed hereinafter. It should be understood that section headings are used in the present document to facilitate ease of understanding and do not limit the embodiments disclosed in a section to only that section. Furthermore, while certain embodiments are described with reference to Versatile Video Coding or other specific video codecs, the disclosed techniques are applicable to other video coding technologies also. Furthermore, while some embodiments describe video coding steps in detail, it will be understood that corresponding steps decoding that undo the coding will be implemented by a decoder. Furthermore, the term video processing encompasses video coding or compression, video decoding or decompression and video transcoding in which video pixels are represented from one compressed format into another compressed format or at a different compressed bitrate.

1 Summary

This disclosure is related to video coding technologies. Specifically, it is about generating prediction blocks from more than one composition, wherein each composition may obtained from different coding techniques. It may be applied to the existing video coding standard like HEVC, VVC, and etc. It may be also applicable to future video coding standards or video codec.

2 Background

Video coding standards have evolved primarily through the development of the well-known ITU-T and ISO/IEC standards. The ITU-T produced H.261 and H.263, ISO/IEC produced MPEG-1 and MPEG-4 Visual, and the two organizations jointly produced the H.262/MPEG-2 Video and H.264/MPEG-4 Advanced Video Coding (AVC) and H.265/HEVC standards. Since H.262, the video coding standards are based on the hybrid video coding structure wherein temporal prediction plus transform coding are utilized. To explore the future video coding technologies beyond HEVC, the Joint Video Exploration Team (JVET) was founded by VCEG and MPEG jointly in 2015. The JVET meeting is concurrently held once every quarter, and the new video coding standard was officially named as Versatile Video Coding (VVC) in the April 2018 JVET meeting, and the first version of VVC test model (VTM) was released at that time. The VVC

working draft and test model VTM are then updated after every meeting. The VVC project achieved technical completion (FDIS) at the July 2020 meeting.

2.1 Coding tools

2.1.1. Intra prediction

5 2.1.1.1. Intra mode coding with 67 intra prediction modes

To capture the arbitrary edge directions presented in natural video, the number of directional intra modes in VVC is extended from 33, as used in HEVC, to 65. Fig. 4 illustrates a schematic diagram 400 of intra prediction modes. The new directional modes not in HEVC are depicted as dotted arrows in Fig. 4, and the planar and DC modes remain the same. These denser directional intra 10 prediction modes apply for all block sizes and for both luma and chroma intra predictions.

In VVC, several conventional angular intra prediction modes are adaptively replaced with wide-angle intra prediction modes for the non-square blocks.

In HEVC, every intra-coded block has a square shape and the length of each of its side is a power of 2. Thus, no division operations are required to generate an intra-predictor using DC mode. In 15 VVC, blocks can have a rectangular shape that necessitates the use of a division operation per block in the general case. To avoid division operations for DC prediction, only the longer side is used to compute the average for non-square blocks.

2.1.1.2 Intra mode coding

To keep the complexity of the most probable mode (MPM) list generation low, an intra mode 20 coding method with 6 MPMs is used by considering two available neighboring intra modes. The following three aspects are considered to construct the MPM list:

- Default intra modes
- Neighbouring intra modes
- Derived intra modes.

25 A unified 6-MPM list is used for intra blocks irrespective of whether MRL and ISP coding tools are applied or not. The MPM list is constructed based on intra modes of the left and above neighboring block. Suppose the mode of the left is denoted as *Left* and the mode of the above block is denoted as *Above*, the unified MPM list is constructed as follows:

- When a neighboring block is not available, its intra mode is set to Planar by default.
- 30 – If both modes *Left* and *Above* are non-angular modes:

- MPM list $\rightarrow$ {Planar, DC, V, H, $V - 4$, $V + 4$ }
- If one of modes Left and Above is angular mode, and the other is non-angular:
 - Set a mode Max as the larger mode in Left and Above
 - MPM list $\rightarrow$ {Planar, Max, DC, $\text{Max} - 1$, $\text{Max} + 1$, $\text{Max} - 2$ }
- 5 – If Left and Above are both angular and they are different:
 - Set a mode Max as the larger mode in Left and Above
 - if the difference of mode Left and Above is in the range of 2 to 62, inclusive
 - MPM list $\rightarrow$ {Planar, Left, Above, DC, $\text{Max} - 1$, $\text{Max} + 1$ }
 - Otherwise
 - 10 – MPM list $\rightarrow$ {Planar, Left, Above, DC, $\text{Max} - 2$, $\text{Max} + 2$ }
- If Left and Above are both angular and they are the same:
 - MPM list $\rightarrow$ {Planar, Left, $\text{Left} - 1$, $\text{Left} + 1$, DC, $\text{Left} - 2$ }

Besides, the first bin of the mpm index codeword is CABAC context coded. In total three contexts are used, corresponding to whether the current intra block is MRL enabled, ISP enabled, or a normal intra block.

During 6 MPM list generation process, pruning is used to remove duplicated modes so that only unique modes can be included into the MPM list. For entropy coding of the 61 non-MPM modes, a Truncated Binary Code (TBC) is used.

2.1.1.3 Wide-angle intra prediction for non-square blocks

- 20 Conventional angular intra prediction directions are defined from 45 degrees to -135 degrees in clockwise direction. In VVC, several conventional angular intra prediction modes are adaptively replaced with wide-angle intra prediction modes for non-square blocks. The replaced modes are signalled using the original mode indexes, which are remapped to the indexes of wide angular modes after parsing. The total number of intra prediction modes is unchanged, i.e., 67, and the
- 25 intra mode coding method is unchanged.

Fig. 5A illustrates a schematic diagram 500 of top reference.

Fig. 5B illustrates a schematic diagram 550 of left reference.

To support these prediction directions, the top reference with length $2W+1$ is defined as reference as shown in Fig. 5A, and the left reference with length $2H+1$ is defined as reference as shown in

30 Fig. 5B.

The number of replaced modes in wide-angular direction mode depends on the aspect ratio of a block.

The replaced intra prediction modes are illustrated in Table 1.

Table 1 – Intra prediction modes replaced by wide-angular modes

Aspect ratio	Replaced intra prediction modes
$W / H == 16$	Modes 12, 13, 14, 15
$W / H == 8$	Modes 12, 13
$W / H == 4$	Modes 2, 3, 4, 5, 6, 7, 8, 9, 10, 11
$W / H == 2$	Modes 2, 3, 4, 5, 6, 7,
$W / H == 1$	None
$W / H == 1/2$	Modes 61, 62, 63, 64, 65, 66
$W / H == 1/4$	Mode 57, 58, 59, 60, 61, 62, 63, 64, 65, 66
$W / H == 1/8$	Modes 55, 56
$W / H == 1/16$	Modes 53, 54, 55, 56

- 5 Fig. 6 illustrates a schematic diagram 600 of discontinuity in case of directions beyond 45° . As shown in Fig. 6, two vertically-adjacent predicted samples may use two non-adjacent reference samples in the case of wide-angle intra prediction. Hence, low-pass reference samples filter and side smoothing are applied to the wide-angle prediction to reduce the negative effect of the increased gap Δp_a . If a wide-angle mode represents a non-fractional offset. There are 8 modes in
- 10 the wide-angle modes satisfy this condition, which are $[-14, -12, -10, -6, 72, 76, 78, 80]$. When a block is predicted by these modes, the samples in the reference buffer are directly copied without applying any interpolation. With this modification, the number of samples needed to be smoothing is reduced. Besides, it aligns the design of non-fractional modes in the conventional prediction modes and wide-angle modes.
- 15 In VVC, 4:2:2 and 4:4:4 chroma formats are supported as well as 4:2:0. Chroma derived mode (DM) derivation table for 4:2:2 chroma format was initially ported from HEVC extending the number of entries from 35 to 67 to align with the extension of intra prediction modes. Since HEVC

specification does not support prediction angle below -135 degree and above 45 degree, luma intra prediction modes ranging from 2 to 5 are mapped to 2. Therefore chroma DM derivation table for 4:2:2: chroma format is updated by replacing some values of the entries of the mapping table to convert prediction angle more precisely for chroma blocks.

5 **2.1.1.4 Mode dependent intra smoothing (MDIS)**

Four-tap intra interpolation filters are utilized to improve the directional intra prediction accuracy. In HEVC, a two-tap linear interpolation filter has been used to generate the intra prediction block in the directional prediction modes (i.e., excluding Planar and DC predictors). In VVC, simplified 6-bit 4-tap Gaussian interpolation filter is used for only directional intra modes. Non-directional intra prediction process is unmodified. The selection of the 4-tap filters is performed according to the MDIS condition for directional intra prediction modes that provide non-fractional displacements, i.e. to all the directional modes excluding the following: 2, HOR_IDX, DIA_IDX, VER_IDX, 66.

Depending on the intra prediction mode, the following reference samples processing is performed:

- 15 – The directional intra-prediction mode is classified into one of the following groups:
 - vertical or horizontal modes (HOR_IDX, VER_IDX),
 - diagonal modes that represent angles which are multiple of 45 degree (2, DIA_IDX, VDIA_IDX),
 - remaining directional modes;
- 20 – If the directional intra-prediction mode is classified as belonging to group A, then then no filters are applied to reference samples to generate predicted samples;
- Otherwise, if a mode falls into group B, then a [1, 2, 1] reference sample filter may be applied (depending on the MDIS condition) to reference samples to further copy these filtered values into an intra predictor according to the selected direction, but no
- 25 interpolation filters are applied;
- Otherwise, if a mode is classified as belonging to group C, then only an intra reference sample interpolation filter is applied to reference samples to generate a predicted sample that falls into a fractional or integer position between reference samples according to a selected direction (no reference sample filtering is performed).

2.1.1.5 Position dependent intra prediction combination

In VVC, the results of intra prediction of DC, planar and several angular modes are further modified by a position dependent intra prediction combination (PDPC) method. PDPC is an intra prediction method which invokes a combination of the un-filtered boundary reference samples and HEVC style intra prediction with filtered boundary reference samples. PDPC is applied to the following intra modes without signalling: planar, DC, horizontal, vertical, bottom-left angular mode and its *eight* adjacent angular modes, and top-right angular mode and its *eight* adjacent angular modes.

The prediction sample $pred(x', y')$ is predicted using an intra prediction mode (DC, planar, angular) and a linear combination of reference samples according to the Equation 3-8 as follows:

$$pred(x', y') = (wL \times R_{-1, y'} + wT \times R_{x', -1} - wTL \times R_{-1, -1} + (64 - wL - wT + wTL) \times pred(x', y') + 32) \gg 6 \quad (2-1)$$

where $R_{x, -1}$, $R_{-1, y}$ represent the reference samples located at the top and left boundaries of current sample (x, y) , respectively, and $R_{-1, -1}$ represents the reference sample located at the top-left corner of the current block.

If PDPC is applied to DC, planar, horizontal, and vertical intra modes, additional boundary filters are not needed, as required in the case of HEVC DC mode boundary filter or horizontal/vertical mode edge filters. PDPC process for DC and Planar modes is identical and clipping operation is avoided. For angular modes, pdpc scale factor is adjusted such that range check is not needed and condition on angle to enable pdpc is removed (scale ≥ 0 is used). In addition, PDPC weight is based on 32 in all angular mode cases. The PDPC weights are dependent on prediction modes and are shown in Table 2. PDPC is applied to the block with both width and height greater than or equal to 4.

Fig. 7A illustrates a schematic diagram 700 of the definition of samples used by PDPC applied to diagonal top-right intra mode. Fig. 7B illustrates a schematic diagram 720 of the definition of samples used by PDPC applied to diagonal bottom-left intra mode. Fig. 7C illustrates a schematic diagram 740 of the definition of samples used by PDPC applied to adjacent diagonal top-right intra mode. Fig. 7D illustrates a schematic diagram 760 of the definition of samples used by PDPC applied to adjacent diagonal bottom-left intra mode. Figs. 7A-7D illustrate the definition of reference samples ($R_{x, -1}$, $R_{-1, y}$ and $R_{-1, -1}$) for PDPC applied over various prediction modes. The prediction sample $pred(x', y')$ is located at (x', y') within the prediction block. As an example, the coordinate x of the reference sample $R_{x, -1}$ is given by: $x = x' + y' + 1$, and the coordinate y of the reference sample $R_{-1, y}$ is similarly given by: $y = x' + y' + 1$ for the diagonal modes. For the other

annular mode, the reference samples R_{x-1} and $R_{-1,y}$ could be located in fractional sample position. In this case, the sample value of the nearest integer sample location is used.

Table 2 - Example of PDPC weights according to prediction modes

Prediction modes	wT	wL	wTL
Diagonal top-right	$16 \gg ((y' \ll 1) \gg \text{shift})$	$16 \gg ((x' \ll 1) \gg \text{shift})$	0
Diagonal bottom-left	$16 \gg ((y' \ll 1) \gg \text{shift})$	$16 \gg ((x' \ll 1) \gg \text{shift})$	0
Adjacent diagonal top-right	$32 \gg ((y' \ll 1) \gg \text{shift})$	0	0
Adjacent diagonal bottom-left	0	$32 \gg ((x' \ll 1) \gg \text{shift})$	0

2.1.1.6 Multiple reference line (MRL) intra prediction

- 5 Multiple reference line (MRL) intra prediction uses more reference lines for intra prediction. Fig. 8 illustrates example diagram 800 of four reference lines neighboring to a prediction block. In Fig. 8, an example of 4 reference lines is depicted, where the samples of segments A and F are not fetched from reconstructed neighbouring samples but padded with the closest samples from Segment B and E, respectively. HEVC intra-picture prediction uses the nearest reference line (i.e.,
10 reference line 0). In MRL, 2 additional lines (reference line 1 and reference line 3) are used.

The index of selected reference line (mrl_idx) is signalled and used to generate intra predictor. For reference line idx , which is greater than 0, only include additional reference line modes in MPM list and only signal mpm index without remaining mode. The reference line index is signalled before intra prediction modes, and Planar mode is excluded from intra prediction modes
15 in case a nonzero reference line index is signalled.

MRL is disabled for the first line of blocks inside a CTU to prevent using extended reference samples outside the current CTU line. Also, PDPC is disabled when additional line is used. For MRL mode, the derivation of DC value in DC intra prediction mode for non-zero reference line indices is aligned with that of reference line index 0. MRL requires the storage of 3 neighboring
20 luma reference lines with a CTU to generate predictions. The Cross-Component Linear Model (CCLM) tool also requires 3 neighboring luma reference lines for its downsampling filters. The

definition of MLR to use the same 3 lines is aligned as CCLM to reduce the storage requirements for decoders.

2.1.1.7 Intra sub-partitions (ISP)

The intra sub-partitions (ISP) divides luma intra-predicted blocks vertically or horizontally into 2 or 4 sub-partitions depending on the block size. For example, minimum block size for ISP is 4x8 (or 8x4). If block size is greater than 4x8 (or 8x4) then the corresponding block is divided by 4 sub-partitions. It has been noted that the $M \times 128$ (with $M \leq 64$) and $128 \times N$ (with $N \leq 64$) ISP blocks could generate a potential issue with the 64×64 VDPU. For example, an $M \times 128$ CU in the single tree case has an $M \times 128$ luma TB and two corresponding $\frac{M}{2} \times 64$ chroma TBs.

If the CU uses ISP, then the luma TB will be divided into four $M \times 32$ TBs (only the horizontal split is possible), each of them smaller than a 64×64 block. However, in the current design of ISP chroma blocks are not divided. Therefore, both chroma components will have a size greater than a 32×32 block. Analogously, a similar situation could be created with a $128 \times N$ CU using ISP. Hence, these two cases are an issue for the 64×64 decoder pipeline. For this reason, the CU sizes that can use ISP is restricted to a maximum of 64×64 . Fig. 9A illustrates an example diagram 910 illustrating examples of sub-partitions for 4x8 and 8x4 CUs. Fig. 9B illustrates an example diagram 920 illustrating examples of sub-partitions for CUs other than 4x8, 8x4 and 4x4. All sub-partitions fulfill the condition of having at least 16 samples.

In ISP, the dependence of $1 \times N/2 \times N$ subblock prediction on the reconstructed values of previously decoded $1 \times N/2 \times N$ subblocks of the coding block is not allowed so that the minimum width of prediction for subblocks becomes four samples. For example, an $8 \times N$ ($N > 4$) coding block that is coded using ISP with vertical split is split into two prediction regions each of size $4 \times N$ and four transforms of size $2 \times N$. Also, a $4 \times N$ coding block that is coded using ISP with vertical split is predicted using the full $4 \times N$ block; four transform each of $1 \times N$ is used. Although the transform sizes of $1 \times N$ and $2 \times N$ are allowed, it is asserted that the transform of these blocks in $4 \times N$ regions can be performed in parallel. For example, when a $4 \times N$ prediction region contains four $1 \times N$ transforms, there is no transform in the horizontal direction; the transform in the vertical direction can be performed as a single $4 \times N$ transform in the vertical direction. Similarly, when a $4 \times N$ prediction region contains two $2 \times N$ transform blocks, the transform operation of the two $2 \times N$ blocks in each direction (horizontal and vertical) can be conducted in parallel. Thus, there is no delay added in processing these smaller blocks than processing 4×4 regular-coded intra blocks.

Table 3 – Entropy coding coefficient group size

Block Size	Coefficient group Size
$1 \times N, N \geq 16$	1×16
$N \times 1, N \geq 16$	16×1
$2 \times N, N \geq 8$	2×8
$N \times 2, N \geq 8$	8×2
All other possible $M \times N$ cases	4×4

For each sub-partition, reconstructed samples are obtained by adding the residual signal to the prediction signal. Here, a residual signal is generated by the processes such as entropy decoding, inverse quantization and inverse transform. Therefore, the reconstructed sample values of each sub-partition are available to generate the prediction of the next sub-partition, and each sub-partition is processed repeatedly. In addition, the first sub-partition to be processed is the one containing the top-left sample of the CU and then continuing downwards (horizontal split) or rightwards (vertical split). As a result, reference samples used to generate the sub-partitions prediction signals are only located at the left and above sides of the lines. All sub-partitions share the same intra mode. The followings are summary of interaction of ISP with other coding tools.

- Multiple Reference Line (MRL): if a block has an MRL index other than 0, then the ISP coding mode will be inferred to be 0 and therefore ISP mode information will not be sent to the decoder.
- Entropy coding coefficient group size: the sizes of the entropy coding subblocks have been modified so that they have 16 samples in all possible cases, as shown in Table 3. Note that the new sizes only affect blocks produced by ISP in which one of the dimensions is less than 4 samples. In all other cases coefficient groups keep the 4×4 dimensions.
- CBF coding: it is assumed to have at least one of the sub-partitions has a non-zero CBF. Hence, if n is the number of sub-partitions and the first $n - 1$ sub-partitions have produced a zero CBF, then the CBF of the n -th sub-partition is inferred to be 1.
- MPM usage: the MPM flag will be inferred to be one in a block coded by ISP mode, and the MPM list is modified to exclude the DC mode and to prioritize horizontal intra modes for the ISP horizontal split and vertical intra modes for the vertical one.
- Transform size restriction: all ISP transforms with a length larger than 16 points uses the DCT-II.

- PDPC: when a CU uses the ISP coding mode, the PDPC filters will not be applied to the resulting sub-partitions.
- MTS flag: if a CU uses the ISP coding mode, the MTS CU flag will be set to 0 and it will not be sent to the decoder. Therefore, the encoder will not perform RD tests for the different available transforms for each resulting sub-partition. The transform choice for the ISP mode will instead be fixed and selected according the intra mode, the processing order and the block size utilized. Hence, no signalling is required. For example, let t_H and t_V be the horizontal and the vertical transforms selected respectively for the $w \times h$ sub-partition, where w is the width and h is the height. Then the transform is selected according to the following rules:

- If $w = 1$ or $h = 1$, then there is no horizontal or vertical transform respectively.
- If $w = 2$ or $w > 32$, $t_H = \text{DCT-II}$
- If $h = 2$ or $h > 32$, $t_V = \text{DCT-II}$
- Otherwise, the transform is selected as in Table 4.

Table 4– Transform selection depends on intra mode

Intra mode	t_H	t_V
Planar Ang. 31,32,34,36,37	DST-VII	DST-VII
DC Ang. 33, 35	DCT-II	DCT-II
Ang. 2, 4, 6...28,30 Ang. 39,41,43...63,65	DST-VII	DCT-II
Ang. 3,5,7...27,29 Ang. 38,40,42...64,66	DCT-II	DST-VII

In ISP mode, all 67 intra modes are allowed. PDPC is also applied if corresponding width and height is at least 4 samples long. In addition, the condition for intra interpolation filter selection doesn't exist anymore, and Cubic (DCT-IF) filter is always applied for fractional position interpolation in ISP mode.

2.1.1.8 Matrix weighted Intra Prediction (MIP)

Matrix weighted intra prediction (MIP) method is a newly added intra prediction technique into VVC. For predicting the samples of a rectangular block of width W and height H , matrix weighted intra prediction (MIP) takes one line of H reconstructed neighbouring boundary samples left of the block and one line of W reconstructed neighbouring boundary samples above the block as input. If the reconstructed samples are unavailable, they are generated as it is done in the conventional intra prediction. Fig. 10 illustrates a schematic diagram 1000 of matrix weighted intra prediction process. The generation of the prediction signal is based on the following three steps, which are averaging, matrix vector multiplication and linear interpolation as shown in Fig. 10.

Averaging neighboring samples

Among the boundary samples, four samples or eight samples are selected by averaging based on block size and shape. Specifically, the input boundaries $bdry_{red}^{top}$ and $bdry_{red}^{left}$ are reduced to smaller boundaries $bdry_{red}^{top}$ and $bdry_{red}^{left}$ by averaging neighboring boundary samples according to predefined rule depends on block size. Then, the two reduced boundaries $bdry_{red}^{top}$ and $bdry_{red}^{left}$ are concatenated to a reduced boundary vector $bdry_{red}$ which is thus of size four for blocks of shape 4×4 and of size eight for blocks of all other shapes. If *mode* refers to the MIP-mode, this concatenation is defined as follows:

$$bdry_{red} = \begin{cases} [bdry_{red}^{top}, bdry_{red}^{left}] & \text{for } W = H = 4 \text{ and } mode < 18 \\ [bdry_{red}^{left}, bdry_{red}^{top}] & \text{for } W = H = 4 \text{ and } mode \geq 18 \\ [bdry_{red}^{top}, bdry_{red}^{left}] & \text{for } \max(W, H) = 8 \text{ and } mode < 10 \\ [bdry_{red}^{left}, bdry_{red}^{top}] & \text{for } \max(W, H) = 8 \text{ and } mode \geq 10 \\ [bdry_{red}^{top}, bdry_{red}^{left}] & \text{for } \max(W, H) > 8 \text{ and } mode < 6 \\ [bdry_{red}^{left}, bdry_{red}^{top}] & \text{for } \max(W, H) > 8 \text{ and } mode \geq 6. \end{cases} \quad (2-2)$$

Matrix Multiplication

A matrix vector multiplication, followed by addition of an offset, is carried out with the averaged samples as an input. The result is a reduced prediction signal on a subsampled set of samples in the original block. Out of the reduced input vector $bdry_{red}$ a reduced prediction signal $pred_{red}$, which is a signal on the downsampled block of width W_{red} and height H_{red} is generated. Here, W_{red} and H_{red} are defined as:

$$W_{red} = \begin{cases} 4 & \text{for } \max(W, H) \leq 8 \\ \min(W, 8) & \text{for } \max(W, H) > 8 \end{cases} \quad (2-3)$$

$$H_{red} = \begin{cases} 4 & \text{for } \max(W, H) \leq 8 \\ \min(H, 8) & \text{for } \max(W, H) > 8 \end{cases} \quad (2-4)$$

The reduced prediction signal $pred_{red}$ is computed by calculating a matrix vector product and adding an offset:

$$pred_{red} = A \cdot bdry_{red} + b.$$

- 5 Here, A is a matrix that has $W_{red} \cdot H_{red}$ rows and 4 columns if $W = H = 4$ and 8 columns in all other cases. b is a vector of size $W_{red} \cdot H_{red}$. The matrix A and the offset vector b are taken from one of the sets S_0, S_1, S_2 . One defines an index $idx = idx(W, H)$ as follows:

$$idx(W, H) = \begin{cases} 0 & \text{for } W = H = 4 \\ 1 & \text{for } \max(W, H) = 8 \\ 2 & \text{for } \max(W, H) > 8. \end{cases} \quad (2-5)$$

- Here, each coefficient of the matrix A is represented with 8 bit precision. The set S_0 consists of
 10 16 matrices $A_0^i, i \in \{0, \dots, 15\}$ each of which has 16 rows and 4 columns and 16 offset vectors $b_0^i, i \in \{0, \dots, 15\}$ each of size 16. Matrices and offset vectors of that set are used for blocks of size 4×4 . The set S_1 consists of 8 matrices $A_1^i, i \in \{0, \dots, 7\}$, each of which has 16 rows and 8 columns and 8 offset vectors $b_1^i, i \in \{0, \dots, 7\}$ each of size 16. The set S_2 consists of 6 matrices $A_2^i, i \in \{0, \dots, 5\}$, each of which has 64 rows and 8 columns and of 6 offset vectors $b_2^i, i \in \{0, \dots, 5\}$
 15 of size 64.

Interpolation

- The prediction signal at the remaining positions is generated from the prediction signal on the subsampled set by linear interpolation which is a single step linear interpolation in each direction. The interpolation is performed firstly in the horizontal direction and then in the vertical direction
 20 regardless of block shape or block size.

Signaling of MIP mode and harmonization with other coding tools

- For each Coding Unit (CU) in intra mode, a flag indicating whether an MIP mode is to be applied or not is sent. If an MIP mode is to be applied, MIP mode ($predModelIntra$) is signaled. For an MIP mode, a transposed flag ($isTransposed$), which determines whether the mode is transposed,
 25 and MIP mode Id ($modelId$), which determines which matrix is to be used for the given MIP mode is derived as follows

$$\begin{aligned} isTransposed &= predModelIntra \& 1 \\ modelId &= predModelIntra \gg 1 \end{aligned} \quad (2-6)$$

MIP coding mode is harmonized with other coding tools by considering following aspects:

- LFNST is enabled for MIP on large blocks. Here, the LFNST transforms of planar mode are used
- The reference sample derivation for MIP is performed exactly as for the conventional intra prediction modes
- For the upsampling step used in the MIP-prediction, original reference samples are used instead of downsampled ones
- Clipping is performed before upsampling and not after upsampling
- MIP is allowed up to 64x64 regardless of the maximum transform size
- The number of MIP modes is 32 for sizeId=0, 16 for sizeId=1 and 12 for sizeId=2

2.1.2 Inter prediction

For each inter-predicted CU, motion parameters consisting of motion vectors, reference picture indices and reference picture list usage index, and additional information needed for the new coding feature of VVC to be used for inter-predicted sample generation. The motion parameter can be signalled in an explicit or implicit manner. When a CU is coded with skip mode, the CU is associated with one PU and has no significant residual coefficients, no coded motion vector delta or reference picture index. A merge mode is specified whereby the motion parameters for the current CU are obtained from neighbouring CUs, including spatial and temporal candidates, and additional schedules introduced in VVC. The merge mode can be applied to any inter-predicted CU, not only for skip mode. The alternative to merge mode is the explicit transmission of motion parameters, where motion vector, corresponding reference picture index for each reference picture list and reference picture list usage flag and other needed information are signalled explicitly per each CU.

Beyond the inter coding features in HEVC, VVC includes a number of new and refined inter prediction coding tools listed as follows:

- Extended merge prediction
- Merge mode with MVD (MMVD)
- Symmetric MVD (SMVD) signalling
- Affine motion compensated prediction
- Subblock-based temporal motion vector prediction (SbTMVP)
- Adaptive motion vector resolution (AMVR)

- Motion field storage: 1/16th luma sample MV storage and 8x8 motion field compression
- Bi-prediction with CU-level weight (BCW)
- Bi-directional optical flow (BDOF)
- Decoder side motion vector refinement (DMVR)
- 5 – Geometric partitioning mode (GPM)
- Combined inter and intra prediction (CIIP)

The following text provides the details on those inter prediction methods specified in VVC.

2.1.2.1 Extended merge prediction

10 In VVC, the merge candidate list is constructed by including the following five types of candidates in order:

- 1) Spatial MVP from spatial neighbour CUs
- 2) Temporal MVP from collocated CUs
- 3) History-based MVP from an FIFO table
- 4) Pairwise average MVP
- 15 5) Zero MVs.

The size of merge list is signalled in sequence parameter set header and the maximum allowed size of merge list is 6. For each CU code in merge mode, an index of best merge candidate is encoded using truncated unary binarization (TU). The first bin of the merge index is coded with context and bypass coding is used for other bins.

20 The derivation process of each category of merge candidates is provided in this session. As done in HEVC, VVC also supports parallel derivation of the merging candidate lists for all CUs within a certain size of area.

2.1.2.2 Spatial candidates derivation

25 The derivation of spatial merge candidates in VVC is same to that in HEVC except the positions of first two merge candidates are swapped. Fig. 11 illustrates a schematic diagram 1100 of positions of spatial merge candidates. A maximum of four merge candidates are selected among candidates located in the positions depicted in Fig. 11. The order of derivation is B₀, A₀, B₁, A₁ and B₂. Position B₂ is considered only when one or more than one CUs of position B₀, A₀, B₁, A₁ are not available (e.g. because it belongs to another slice or tile) or is intra coded. After candidate at position A₁ is added, the addition of the remaining candidates is subject to a redundancy check

30 which ensures that candidates with same motion information are excluded from the list so that

coding efficiency is improved. To reduce computational complexity, not all possible candidate pairs are considered in the mentioned redundancy check. Fig. 12 illustrates a schematic diagram 1200 of candidate pairs considered for redundancy check of spatial merge candidates. Instead only the pairs linked with an arrow in Fig. 12 are considered and a candidate is only added to the list if the corresponding candidate used for redundancy check has not the same motion information.

2.1.2.3 Temporal candidates derivation

In this step, only one candidate is added to the list. Particularly, in the derivation of this temporal merge candidate, a scaled motion vector is derived based on co-located CU belonging to the collocated reference picture. The reference picture list to be used for derivation of the co-located CU is explicitly signalled in the slice header. Fig. 13 illustrates a schematic diagram 1300 of motion vector scaling for temporal merge candidate. The scaled motion vector for temporal merge candidate is obtained as illustrated by the dotted line in Fig. 13, which is scaled from the motion vector of the co-located CU using the POC distances, t_b and t_d , where t_b is defined to be the POC difference between the reference picture of the current picture and the current picture and t_d is defined to be the POC difference between the reference picture of the co-located picture and the co-located picture. The reference picture index of temporal merge candidate is set equal to zero.

Fig. 14 illustrates a schematic diagram 1400 of candidate positions for temporal merge candidates, C_0 and C_1 . The position for the temporal candidate is selected between candidates C_0 and C_1 , as depicted in Fig. 14. If CU at position C_0 is not available, is intra coded, or is outside of the current row of CTUs, position C_1 is used. Otherwise, position C_0 is used in the derivation of the temporal merge candidate.

2.1.2.4 History-based merge candidates derivation

The history-based MVP (HMVP) merge candidates are added to merge list after the spatial MVP and TMVP. In this method, the motion information of a previously coded block is stored in a table and used as MVP for the current CU. The table with multiple HMVP candidates is maintained during the encoding/decoding process. The table is reset (emptied) when a new CTU row is encountered. Whenever there is a non-subblock inter-coded CU, the associated motion information is added to the last entry of the table as a new HMVP candidate.

The HMVP table size S is set to be 6, which indicates up to 6 History-based MVP (HMVP) candidates may be added to the table. When inserting a new motion candidate to the table, a constrained first-in-first-out (FIFO) rule is utilized wherein redundancy check is firstly applied to

find whether there is an identical HMVP in the table. If found, the identical HMVP is removed from the table and all the HMVP candidates afterwards are moved forward.

HMVP candidates could be used in the merge candidate list construction process. The latest several HMVP candidates in the table are checked in order and inserted to the candidate list after the TMVP candidate. Redundancy check is applied on the HMVP candidates to the spatial or temporal merge candidate.

To reduce the number of redundancy check operations, the following simplifications are introduced:

1. Number of HMPV candidates is used for merge list generation is set as $(N \leq 4) ? M : (8 - N)$, wherein N indicates number of existing candidates in the merge list and M indicates number of available HMVP candidates in the table.
2. Once the total number of available merge candidates reaches the maximally allowed merge candidates minus 1, the merge candidate list construction process from HMVP is terminated.

2.1.2.5 Pair-wise average merge candidates derivation

Pairwise average candidates are generated by averaging predefined pairs of candidates in the existing merge candidate list, and the predefined pairs are defined as $\{(0, 1), (0, 2), (1, 2), (0, 3), (1, 3), (2, 3)\}$, where the numbers denote the merge indices to the merge candidate list. The averaged motion vectors are calculated separately for each reference list. If both motion vectors are available in one list, these two motion vectors are averaged even when they point to different reference pictures; if only one motion vector is available, use the one directly; if no motion vector is available, keep this list invalid.

When the merge list is not full after pair-wise average merge candidates are added, the zero MVPs are inserted in the end until the maximum merge candidate number is encountered.

2.1.2.6 Merge estimation region

Merge estimation region (MER) allows independent derivation of merge candidate list for the CUs in the same merge estimation region (MER). A candidate block that is within the same MER to the current CU is not included for the generation of the merge candidate list of the current CU.

In addition, the updating process for the history-based motion vector predictor candidate list is updated only if $(x_{Cb} + cbWidth) \gg \text{Log2ParMrgLevel}$ is greater than $x_{Cb} \gg \text{Log2ParMrgLevel}$ and $(y_{Cb} + cbHeight) \gg \text{Log2ParMrgLevel}$ is great than $y_{Cb} \gg \text{Log2ParMrgLevel}$ and where (x_{Cb}, y_{Cb}) is the top-left luma sample position of the

current CU in the picture and (cbWidth, cbHeight) is the CU size. The MER size is selected at encoder side and signalled as log2_parallel_merge_level_minus2 in the sequence parameter set.

2.1.3 Merge mode with MVD (MMVD)

In addition to merge mode, where the implicitly derived motion information is directly used for prediction samples generation of the current CU, the merge mode with motion vector differences (MMVD) is introduced in VVC. A MMVD flag is signalled right after sending a skip flag and merge flag to specify whether MMVD mode is used for a CU.

In MMVD, after a merge candidate is selected, it is further refined by the signalled MVDs information. The further information includes a merge candidate flag, an index to specify motion magnitude, and an index for indication of motion direction. In MMVD mode, one for the first two candidates in the merge list is selected to be used as MV basis. The merge candidate flag is signalled to specify which one is used.

Distance index specifies motion magnitude information and indicate the pre-defined offset from the starting point. Fig. 15A illustrates a schematic diagram 1500 of MMVD Search Point for L0 reference. Fig. 15B illustrates a schematic diagram 1550 of MMVD Search Point for L1 reference. As shown in Figs. 15A-15B, an offset is added to either horizontal component or vertical component of starting MV. The relation of distance index and pre-defined offset is specified in Table 5.

Table 5 – The relation of distance index and pre-defined offset

Distance IDX	0	1	2	3	4	5	6	7
Offset (in unit of luma sample)	1/4	1/2	1	2	4	8	16	32

Direction index represents the direction of the MVD relative to the starting point. The direction index can represent of the four directions as shown in Table 6. It's noted that the meaning of MVD sign could be variant according to the information of starting MVs. When the starting MVs is an un-prediction MV or bi-prediction MVs with both lists point to the same side of the current picture (i.e. POCs of two references are both larger than the POC of the current picture, or are both smaller than the POC of the current picture), the sign in Table 6 specifies the sign of MV offset added to the starting MV. When the starting MVs is bi-prediction MVs with the two MVs point to the different sides of the current picture (i.e. the POC of one reference is larger than the POC of the current picture, and the POC of the other reference is smaller than the POC of the current picture),

the sign in Table 6 specifies the sign of MV offset added to the list0 MV component of starting MV and the sign for the list1 MV has opposite value.

Table 6 – Sign of MV offset specified by direction index

Direction IDX	00	01	10	11
x-axis	+	–	N/A	N/A
y-axis	N/A	N/A	+	–

2.1.3.1 Bi-prediction with CU-level weight (BCW)

- 5 In HEVC, the bi-prediction signal is generated by averaging two prediction signals obtained from two different reference pictures and/or using two different motion vectors. In VVC, the bi-prediction mode is extended beyond simple averaging to allow weighted averaging of the two prediction signals.

$$P_{\text{bi-pred}} = ((8 - w) * P_0 + w * P_1 + 4) \gg 3 \quad (2-7)$$

- 10 Five weights are allowed in the weighted averaging bi-prediction, $w \in \{-2, 3, 4, 5, 10\}$. For each bi-predicted CU, the weight w is determined in one of two ways: 1) for a non-merge CU, the weight index is signalled after the motion vector difference; 2) for a merge CU, the weight index is inferred from neighbouring blocks based on the merge candidate index. BCW is only applied to CUs with 256 or more luma samples (i.e., CU width times CU height is greater than or equal to 256). For low-delay pictures, all 5 weights are used. For non-low-delay pictures, only 3 weights
15 ($w \in \{3, 4, 5\}$) are used.

- At the encoder, fast search algorithms are applied to find the weight index without significantly increasing the encoder complexity. These algorithms are summarized as follows. For further details readers are referred to the VTM software and document. When combined with AMVR, unequal weights are only conditionally checked for 1-pel and 4-pel motion
20 vector precisions if the current picture is a low-delay picture.
- When combined with affine, affine ME will be performed for unequal weights if and only if the affine mode is selected as the current best mode.
- When the two reference pictures in bi-prediction are the same, unequal weights are only conditionally checked.

- Unequal weights are not searched when certain conditions are met, depending on the POC distance between current picture and its reference pictures, the coding QP, and the temporal level.

The BCW weight index is coded using one context coded bin followed by bypass coded bins. The first context coded bin indicates if equal weight is used; and if unequal weight is used, additional bins are signalled using bypass coding to indicate which unequal weight is used.

Weighted prediction (WP) is a coding tool supported by the H.264/AVC and HEVC standards to efficiently code video content with fading. Support for WP was also added into the VVC standard. WP allows weighting parameters (weight and offset) to be signalled for each reference picture in each of the reference picture lists L0 and L1. Then, during motion compensation, the weight(s) and offset(s) of the corresponding reference picture(s) are applied. WP and BCW are designed for different types of video content. In order to avoid interactions between WP and BCW, which will complicate VVC decoder design, if a CU uses WP, then the BCW weight index is not signalled, and w is inferred to be 4 (i.e. equal weight is applied). For a merge CU, the weight index is inferred from neighbouring blocks based on the merge candidate index. This can be applied to both normal merge mode and inherited affine merge mode. For constructed affine merge mode, the affine motion information is constructed based on the motion information of up to 3 blocks. The BCW index for a CU using the constructed affine merge mode is simply set equal to the BCW index of the first control point MV.

In VVC, CIIP and BCW cannot be jointly applied for a CU. When a CU is coded with CIIP mode, the BCW index of the current CU is set to 2, e.g. equal weight.

2.1.3.2 Bi-directional optical flow (BDOF)

The bi-directional optical flow (BDOF) tool is included in VVC. BDOF, previously referred to as BIO, was included in the JEM. Compared to the JEM version, the BDOF in VVC is a simpler version that requires much less computation, especially in terms of number of multiplications and the size of the multiplier.

BDOF is used to refine the bi-prediction signal of a CU at the 4×4 subblock level. BDOF is applied to a CU if it satisfies all the following conditions:

- The CU is coded using “true” bi-prediction mode, i.e., one of the two reference pictures is prior to the current picture in display order and the other is after the current picture in display order

- The distances (i.e. POC difference) from two reference pictures to the current picture are same
 - Both reference pictures are short-term reference pictures.
 - The CU is not coded using affine mode or the ATMVP merge mode
- 5 – CU has more than 64 luma samples
- Both CU height and CU width are larger than or equal to 8 luma samples
 - BCW weight index indicates equal weight
 - WP is not enabled for the current CU
 - CIIP mode is not used for the current CU
- 10 BDOF is only applied to the luma component. As its name indicates, the BDOF mode is based on the optical flow concept, which assumes that the motion of an object is smooth. For each 4×4 subblock, a motion refinement (v_x, v_y) is calculated by minimizing the difference between the L0 and L1 prediction samples. The motion refinement is then used to adjust the bi-predicted sample values in the 4x4 subblock. The following steps are applied in the BDOF process.
- 15 First, the horizontal and vertical gradients, $\frac{\partial I^{(k)}}{\partial x}(i, j)$ and $\frac{\partial I^{(k)}}{\partial y}(i, j)$, $k = 0, 1$, of the two prediction signals are computed by directly calculating the difference between two neighboring samples, i.e.,

$$\begin{aligned}\frac{\partial I^{(k)}}{\partial x}(i, j) &= (I^{(k)}(i+1, j) \gg \text{shift1}) - (I^{(k)}(i-1, j) \gg \text{shift1}) \\ \frac{\partial I^{(k)}}{\partial y}(i, j) &= (I^{(k)}(i, j+1) \gg \text{shift1}) - (I^{(k)}(i, j-1) \gg \text{shift1})\end{aligned}\quad (2-8)$$

where $I^{(k)}(i, j)$ are the sample value at coordinate (i, j) of the prediction signal in list k , $k = 0, 1$, and shift1 is calculated based on the luma bit depth, bitDepth, as $\text{shift1} = \max(6, \text{bitDepth}-6)$.

Then, the auto- and cross-correlation of the gradients, S_1, S_2, S_3, S_5 and S_6 , are calculated as

$$\begin{aligned}S_1 &= \sum_{(i,j) \in \Omega} \text{Abs}(\psi_x(i, j)), \quad S_3 = \sum_{(i,j) \in \Omega} \theta(i, j) \cdot \text{Sign}(\psi_x(i, j)) \\ S_2 &= \sum_{(i,j) \in \Omega} \psi_x(i, j) \cdot \text{Sign}(\psi_y(i, j)) \\ S_5 &= \sum_{(i,j) \in \Omega} \text{Abs}(\psi_y(i, j)), \quad S_6 = \sum_{(i,j) \in \Omega} \theta(i, j) \cdot \text{Sign}(\psi_y(i, j))\end{aligned}\quad (2-9)$$

20 where

$$\psi_x(i, j) = \left(\frac{\partial I^{(1)}}{\partial x}(i, j) + \frac{\partial I^{(0)}}{\partial x}(i, j) \right) \gg n_a \quad (2-10)$$

$$\psi_y(i, j) = \left(\frac{\partial I^{(1)}}{\partial y}(i, j) + \frac{\partial I^{(0)}}{\partial y}(i, j) \right) \gg n_a$$

$$\theta(i, j) = (I^{(1)}(i, j) \gg n_b) - (I^{(0)}(i, j) \gg n_b)$$

where Ω is a 6×6 window around the 4×4 subblock, and the values of n_a and n_b are set equal to $\min(1, \text{bitDepth} - 11)$ and $\min(4, \text{bitDepth} - 8)$, respectively.

The motion refinement (v_x, v_y) is then derived using the cross- and auto-correlation terms using the following:

$$v_x = S_1 > 0 ? \text{clip3} \left(-th'_{BIO}, th'_{BIO}, -((S_3 \cdot 2^{n_b - n_a}) \gg \lfloor \log_2 S_1 \rfloor) \right) : 0$$

$$v_y = S_5 > 0 ? \text{clip3} \left(-th'_{BIO}, th'_{BIO}, - \left((S_6 \cdot 2^{n_b - n_a} - ((v_x S_{2,m}) \ll n_{S_2} + v_x S_{2,s}) / 2 \right) \gg \lfloor \log_2 S_5 \rfloor \right) : 0 \quad (2-11)$$

- 5 where $S_{2,m} = S_2 \gg n_{S_2}$, $S_{2,s} = S_2 \& (2^{n_{S_2}} - 1)$, $th'_{BIO} = 2^{\max(5, BD-7)}$. $\lfloor \cdot \rfloor$ is the floor function, and $n_{S_2} = 12$.

Based on the motion refinement and the gradients, the following adjustment is calculated for each sample in the 4×4 subblock:

$$b(x, y) = \text{rnd} \left(\left(v_x \left(\frac{\partial I^{(1)}(x, y)}{\partial x} - \frac{\partial I^{(0)}(x, y)}{\partial x} \right) + v_y \left(\frac{\partial I^{(1)}(x, y)}{\partial y} - \frac{\partial I^{(0)}(x, y)}{\partial y} \right) + 1 \right) / 2 \right) \quad (2-12)$$

- 10 Finally, the BDOF samples of the CU are calculated by adjusting the bi-prediction samples as follows:

$$\text{pred}_{BDOF}(x, y) = (I^{(0)}(x, y) + I^{(1)}(x, y) + b(x, y) + o_{offset}) \gg \text{shift} \quad (2-13)$$

These values are selected such that the multipliers in the BDOF process do not exceed 15-bit, and the maximum bit-width of the intermediate parameters in the BDOF process is kept within 32-bit.

- 15 In order to derive the gradient values, some prediction samples $I^{(k)}(i, j)$ in list k ($k = 0, 1$) outside of the current CU boundaries need to be generated. Fig. 16 illustrates a schematic diagram 1600 of an extended CU region used in BDOF. As depicted in Fig. 16, the BDOF in VVC uses one extended row/column around the CU's boundaries. In order to control the computational complexity of generating the out-of-boundary prediction samples, prediction samples in the extended area (white positions) are generated by taking the reference samples at the nearby integer positions (using floor() operation on the coordinates) directly without interpolation, and the

normal 8-tap motion compensation interpolation filter is used to generate prediction samples within the CU (gray positions). These extended sample values are used in gradient calculation only. For the remaining steps in the BDOF process, if any sample and gradient values outside of the CU boundaries are needed, they are padded (i.e. repeated) from their nearest neighbors.

- 5 When the width and/or height of a CU are larger than 16 luma samples, it will be split into subblocks with width and/or height equal to 16 luma samples, and the subblock boundaries are treated as the CU boundaries in the BDOF process. The maximum unit size for BDOF process is limited to 16x16. For each subblock, the BDOF process could be skipped. When the SAD of between the initial L0 and L1 prediction samples is smaller than a threshold, the BDOF process is not applied to the subblock. The threshold is set equal to $(8 * W * (H > 1))$, where W indicates the subblock width, and H indicates subblock height. To avoid the additional complexity of SAD calculation, the SAD between the initial L0 and L1 prediction samples calculated in DVMR process is re-used here.

- 15 If BCW is enabled for the current block, i.e., the BCW weight index indicates unequal weight, then bi-directional optical flow is disabled. Similarly, if WP is enabled for the current block, i.e., the luma_weight_lx_flag is 1 for either of the two reference pictures, then BDOF is also disabled. When a CU is coded with symmetric MVD mode or CIIP mode, BDOF is also disabled.

2.1.4 Symmetric MVD coding

- 20 In VVC, besides the normal unidirectional prediction and bi-directional prediction mode MVD signalling, symmetric MVD mode for bi-predictional MVD signalling is applied. In the symmetric MVD mode, motion information including reference picture indices of both list-0 and list-1 and MVD of list-1 are not signaled but derived.

The decoding process of the symmetric MVD mode is as follows:

- 25 1) At slice level, variables BiDirPredFlag, RefIdxSymL0 and RefIdxSymL1 are derived as follows:
- If mvd_l1_zero_flag is 1, BiDirPredFlag is set equal to 0.
 - Otherwise, if the nearest reference picture in list-0 and the nearest reference picture in list-1 form a forward and backward pair of reference pictures or a backward and forward pair of reference pictures, BiDirPredFlag is set to 1, and both list-0 and list-1 reference pictures are short-term reference pictures. Otherwise BiDirPredFlag is set to 0.

- 30 2) At CU level, a symmetrical mode flag indicating whether symmetrical mode is used or not is explicitly signaled if the CU is bi-prediction coded and BiDirPredFlag is equal to 1.

When the symmetrical mode flag is true, only mvp_l0_flag, mvp_l1_flag and MVD0 are explicitly signaled. The reference indices for list-0 and list-1 are set equal to the pair of reference

pictures, respectively. MVD1 is set equal to (- MVD0). The final motion vectors are shown in below formula.

$$\begin{cases} (mvx_0, mvy_0) = (mvp_x_0 + mvd_x_0, mvp_y_0 + mvd_y_0) \\ (mvx_1, mvy_1) = (mvp_x_1 - mvd_x_0, mvp_y_1 - mvd_y_0) \end{cases} \quad (2-14)$$

Fig. 17 illustrates a schematic diagram of an illustration for symmetrical MVD mode. In the encoder, symmetric MVD motion estimation starts with initial MV evaluation. A set of initial MV candidates comprising of the MV obtained from uni-prediction search, the MV obtained from bi-prediction search and the MVs from the AMVP list. For example, there are a List-0 reference picture 1710 and a List-1 reference picture 1730 for the current picture 1720. The one with the lowest rate-distortion cost is chosen to be the initial MV for the symmetric MVD motion search.

2.1.5 Decoder side motion vector refinement (DMVR)

In order to increase the accuracy of the MVs of the merge mode, a bilateral-matching based decoder side motion vector refinement is applied in VVC. In bi-prediction operation, a refined MV is searched around the initial MVs in the reference picture list L0 and reference picture list L1. The BM method calculates the distortion between the two candidate blocks in the reference picture list L0 and list L1. Fig. 18 illustrates a decoding side motion vector refinement. As illustrated in Fig. 18, the SAD between the blocks 1810 and 1812 based on each MV candidate around the initial MV is calculated, where the block 1810 is in a reference picture 1801 in the list L0 and the block 1812 is in a reference picture 1803 in the List L1 for the current picture 1802. The MV candidate with the lowest SAD becomes the refined MV and used to generate the bi-predicted signal.

In VVC, the DMVR can be applied for the CUs which are coded with following modes and features:

- CU level merge mode with bi-prediction MV
- One reference picture is in the past and another reference picture is in the future with respect to the current picture
- The distances (i.e. POC difference) from two reference pictures to the current picture are same
- Both reference pictures are short-term reference pictures
- CU has more than 64 luma samples
- Both CU height and CU width are larger than or equal to 8 luma samples
- BCW weight index indicates equal weight
- WP is not enabled for the current block

- CIIP mode is not used for the current block.

The refined MV derived by DMVR process is used to generate the inter prediction samples and also used in temporal motion vector prediction for future pictures coding. While the original MV is used in deblocking process and also used in spatial motion vector prediction for future CU coding.

The additional features of DMVR are mentioned in the following sub-clauses.

2.1.5.1 Searching scheme

In DVMR, the search points are surrounding the initial MV and the MV offset obey the MV difference mirroring rule. In other words, any points that are checked by DMVR, denoted by candidate MV pair (MV0, MV1) obey the following two equations:

$$MV0' = MV0 + MV_offset \quad (2-15)$$

$$MV1' = MV1 - MV_offset \quad (2-16)$$

Where *MV_offset* represents the refinement offset between the initial MV and the refined MV in one of the reference pictures. The refinement search range is two integer luma samples from the initial MV. The searching includes the integer sample offset search stage and fractional sample refinement stage.

25 points full search is applied for integer sample offset searching. The SAD of the initial MV pair is first calculated. If the SAD of the initial MV pair is smaller than a threshold, the integer sample stage of DMVR is terminated. Otherwise SADs of the remaining 24 points are calculated and checked in raster scanning order. The point with the smallest SAD is selected as the output of integer sample offset searching stage. To reduce the penalty of the uncertainty of DMVR refinement, it is proposed to favor the original MV during the DMVR process. The SAD between the reference blocks referred by the initial MV candidates is decreased by 1/4 of the SAD value.

The integer sample search is followed by fractional sample refinement. To save the calculational complexity, the fractional sample refinement is derived by using parametric error surface equation, instead of additional search with SAD comparison. The fractional sample refinement is conditionally invoked based on the output of the integer sample search stage. When the integer sample search stage is terminated with center having the smallest SAD in either the first iteration or the second iteration search, the fractional sample refinement is further applied.

In parametric error surface based sub-pixel offsets estimation, the center position cost and the costs at four neighboring positions from the center are used to fit a 2-D parabolic error surface equation of the following form

$$E(x, y) = A(x - x_{min})^2 + B(y - y_{min})^2 + C \quad (2-17)$$

- 5 where (x_{min}, y_{min}) corresponds to the fractional position with the least cost and C corresponds to the minimum cost value. By solving the above equations by using the cost value of the five search points, the (x_{min}, y_{min}) is computed as:

$$x_{min} = (E(-1,0) - E(1,0)) / (2(E(-1,0) + E(1,0) - 2E(0,0))) \quad (2-18)$$

$$y_{min} = (E(0,-1) - E(0,1)) / (2(E(0,-1) + E(0,1) - 2E(0,0))) \quad (2-19)$$

- 10 The value of x_{min} and y_{min} are automatically constrained to be between -8 and 8 since all cost values are positive and the smallest value is $E(0,0)$. This corresponds to half pel offset with 1/16th-pel MV accuracy in VVC. The computed fractional (x_{min}, y_{min}) are added to the integer distance refinement MV to get the sub-pixel accurate refinement delta MV.

2.1.5.2 Bilinear-interpolation and sample padding

- 15 In VVC, the resolution of the MVs is 1/16 luma samples. The samples at the fractional position are interpolated using a 8-tap interpolation filter. In DMVR, the search points are surrounding the initial fractional-pel MV with integer sample offset, therefore the samples of those fractional position need to be interpolated for DMVR search process. To reduce the calculation complexity, the bi-linear interpolation filter is used to generate the fractional samples for the searching process
- 20 in DMVR. Another important effect is that by using bi-linear filter is that with 2-sample search range, the DMVR does not access more reference samples compared to the normal motion compensation process. After the refined MV is attained with DMVR search process, the normal 8-tap interpolation filter is applied to generate the final prediction. In order to not access more reference samples to normal MC process, the samples, which is not needed for the interpolation
- 25 process based on the original MV but is needed for the interpolation process based on the refined MV, will be padded from those available samples.

2.1.5.3 Maximum DMVR processing unit

- When the width and/or height of a CU are larger than 16 luma samples, it will be further split into subblocks with width and/or height equal to 16 luma samples. The maximum unit size for DMVR
- 30 searching process is limit to 16x16.

2.1.6 Combined inter and intra prediction (CIIP)

In VVC, when a CU is coded in merge mode, if the CU contains at least 64 luma samples (that is, CU width times CU height is equal to or larger than 64), and if both CU width and CU height are less than 128 luma samples, an additional flag is signalled to indicate if the combined inter/intra prediction (CIIP) mode is applied to the current CU. Fig. 19 illustrates a schematic diagram 1900 of top and left neighboring blocks used in CIIP weight derivation. As its name indicates, the CIIP prediction combines an inter prediction signal with an intra prediction signal. The inter prediction signal in the CIIP mode P_{inter} is derived using the same inter prediction process applied to regular merge mode; and the intra prediction signal P_{intra} is derived following the regular intra prediction process with the planar mode. Then, the intra and inter prediction signals are combined using weighted averaging, where the weight value is calculated depending on the coding modes of the top and left neighbouring blocks (depicted in Fig. 19) as follows:

- If the top neighbor is available and intra coded, then set isIntraTop to 1, otherwise set isIntraTop to 0;
- If the left neighbor is available and intra coded, then set isIntraLeft to 1, otherwise set isIntraLeft to 0;
- If (isIntraLeft + isIntraTop) is equal to 2, then wt is set to 3;
- Otherwise, if (isIntraLeft + isIntraTop) is equal to 1, then wt is set to 2;
- Otherwise, set wt to 1.

The CIIP prediction is formed as follows:

$$P_{CIIP} = ((4 - wt) * P_{inter} + wt * P_{intra} + 2) \gg 2 \quad (2-20)$$

2.1.7 Geometric partitioning mode (GPM)

In VVC, a geometric partitioning mode is supported for inter prediction. The geometric partitioning mode is signalled using a CU-level flag as one kind of merge mode, with other merge modes including the regular merge mode, the MMVD mode, the CIIP mode and the subblock merge mode. In total 64 partitions are supported by geometric partitioning mode for each possible CU size $w \times h = 2^m \times 2^n$ with $m, n \in \{3 \dots 6\}$ excluding 8x64 and 64x8.

Fig. 20 illustrates a schematic diagram 2000 of examples of the GPM splits grouped by identical angles. When this mode is used, a CU is split into two parts by a geometrically located straight line (as shown in Fig. 20). The location of the splitting line is mathematically derived from the angle and offset parameters of a specific partition. Each part of a geometric partition in the CU is inter-predicted using its own motion; only uni-prediction is allowed for each partition, that is, each part has one motion vector and one reference index. The uni-prediction motion constraint is

applied to ensure that same as the conventional bi-prediction, only two motion compensated prediction are needed for each CU.

If geometric partitioning mode is used for the current CU, then a geometric partition index indicating the partition mode of the geometric partition (angle and offset), and two merge indices (one for each partition) are further signalled. The number of maximum GPM candidate size is signalled explicitly in SPS and specifies syntax binarization for GPM merge indices. After predicting each of part of the geometric partition, the sample values along the geometric partition edge are adjusted using a blending processing with adaptive weights. This is the prediction signal for the whole CU, and transform and quantization process will be applied to the whole CU as in other prediction modes. Finally, the motion field of a CU predicted using the geometric partition modes is stored.

2.1.7.1 Uni-prediction candidate list construction

The uni-prediction candidate list is derived directly from the merge candidate list constructed according to the extended merge prediction process. Denote n as the index of the uni-prediction motion in the geometric uni-prediction candidate list. The LX motion vector of the n -th extended merge candidate, with X equal to the parity of n , is used as the n -th uni-prediction motion vector for geometric partitioning mode. Fig. 21 illustrates a schematic diagram 2100 of uni-prediction MV selection for geometric partitioning mode. These motion vectors are marked with “x” in Fig. 21. In case a corresponding LX motion vector of the n -th extended merge candidate does not exist, the $L(1 - X)$ motion vector of the same candidate is used instead as the uni-prediction motion vector for geometric partitioning mode.

2.1.7.2 Blending along the geometric partitioning edge

After predicting each part of a geometric partition using its own motion, blending is applied to the two prediction signals to derive samples around geometric partition edge. The blending weight for each position of the CU are derived based on the distance between individual position and the partition edge.

The distance for a position (x, y) to the partition edge are derived as:

$$d(x, y) = (2x + 1 - w) \cos(\varphi_i) + (2y + 1 - h) \sin(\varphi_i) - \rho_j \quad (2-21)$$

$$\rho_j = \rho_{x,j} \cos(\varphi_i) + \rho_{y,j} \sin(\varphi_i) \quad (2-22)$$

$$\rho_{x,j} = \begin{cases} 0 & i \% 16 = 8 \text{ or } (i \% 16 \neq 0 \text{ and } h \geq w) \\ \pm(j \times w) \gg 2 & \text{otherwise} \end{cases} \quad (2-23)$$

$$\rho_{y,j} = \begin{cases} \pm(j \times h) \gg 2 & i \% 16 = 8 \text{ or } (i \% 16 \neq 0 \text{ and } h \geq w) \\ 0 & \text{otherwise} \end{cases} \quad (2-24)$$

where i, j are the indices for angle and offset of a geometric partition, which depend on the signaled geometric partition index. The sign of $\rho_{x,j}$ and $\rho_{y,j}$ depend on angle index i .

The weights for each part of a geometric partition are derived as following:

$$wIdxL(x, y) = partIdx ? 32 + d(x, y) : 32 - d(x, y) \quad (2-25)$$

$$w_0(x, y) = \frac{Clip3(0, 8, (wIdxL(x, y) + 4) \gg 3)}{8} \quad (2-26)$$

$$w_1(x, y) = 1 - w_0(x, y) \quad (2-27)$$

The $partIdx$ depends on the angle index i . Fig. 22 illustrates a schematic diagram 2200 of exemplified generation of a bending weight w_0 using geometric partitioning mode.

2.1.7.3 Motion field storage for geometric partitioning mode

Mv1 from the first part of the geometric partition, Mv2 from the second part of the geometric partition and a combined Mv of Mv1 and Mv2 are stored in the motion filed of a geometric partitioning mode coded CU.

The stored motion vector type for each individual position in the motion filed are determined as:

$$sType = abs(motionIdx) < 32 ? 2 : (motionIdx \leq 0 ? (1 - partIdx) : partIdx) \quad (2-43)$$

where $motionIdx$ is equal to $d(4x + 2, 4y + 2)$, which is recalculated from equation (2-36). The $partIdx$ depends on the angle index i .

If $sType$ is equal to 0 or 1, Mv0 or Mv1 are stored in the corresponding motion field, otherwise if $sType$ is equal to 2, a combined Mv from Mv0 and Mv2 are stored. The combined Mv are generated using the following process:

- 1) If Mv1 and Mv2 are from different reference picture lists (one from L0 and the other from L1), then Mv1 and Mv2 are simply combined to form the bi-prediction motion vectors.
- 2) Otherwise, if Mv1 and Mv2 are from the same list, only uni-prediction motion Mv2 is stored.

2.1.8 Geometric partitioning mode (GPM) with merge motion vector differences (MMVD)

GPM in VVC is extended by applying motion vector refinement on top of the existing GPM uni-directional MVs. A flag is first signalled for a GPM CU, to specify whether this mode is used. If the mode is used, each geometric partition of a GPM CU can further decide whether to signal MVD or not. If MVD is signalled for a geometric partition, after a GPM merge candidate is

selected, the motion of the partition is further refined by the signalled MVDs information. All other procedures are kept the same as in GPM.

The MVD is signaled as a pair of distance and direction, similar as in MMVD. There are nine candidate distances ($\frac{1}{4}$ -pel, $\frac{1}{2}$ -pel, 1-pel, 2-pel, 3-pel, 4-pel, 6-pel, 8-pel, 16-pel), and eight candidate directions (four horizontal/vertical directions and four diagonal directions) involved in GPM with MMVD (GPM-MMVD). In addition, when `pic_fpel_mmvd_enabled_flag` is equal to 1, the MVD is left shifted by 2 as in MMVD.

2.1.9 Geometric partitioning mode (GPM) with template matching (TM)

Template matching is applied to GPM. When GPM mode is enabled for a CU, a CU-level flag is signaled to indicate whether TM is applied to both geometric partitions. Motion information for each geometric partition is refined using TM. When TM is chosen, a template is constructed using left, above or left and above neighboring samples according to partition angle, as shown in Table 7. In Table 7, A represents using above samples, L represents using left samples, and L+A represents using both left and above samples. The motion is then refined by minimizing the difference between the current template and the template in the reference picture using the same search pattern of merge mode with half-pel interpolation filter disabled.

Table 7 Template for the 1st and 2nd geometric partitions

Partition angle	0	2	3	4	5	8	11	12	13	14
1st partition	A	A	A	A	L+A	L+A	L+A	L+A	A	A
2nd partition	L+A	L+A	L+A	L	L	L	L	L+A	L+A	L+A
Partition angle	16	18	19	20	21	24	27	28	29	30
1st partition	A	A	A	A	L+A	L+A	L+A	L+A	A	A
2nd partition	L+A	L+A	L+A	L	L	L	L	L+A	L+A	L+A

A GPM candidate list is constructed as follows:

1. Interleaved List-0 MV candidates and List-1 MV candidates are derived directly from the regular merge candidate list, where List-0 MV candidates are higher priority than List-1 MV candidates. A pruning method with an adaptive threshold based on the current CU size is applied to remove redundant MV candidates.

2. Interleaved List-1 MV candidates and List-0 MV candidates are further derived directly from the regular merge candidate list, where List-1 MV candidates are higher priority than List-0 MV candidates. The same pruning method with the adaptive threshold is also applied to remove redundant MV candidates.

3. Zero MV candidates are padded until the GPM candidate list is full.

The GPM-MMVD and GPM-TM are exclusively enabled to one GPM CU. This is done by firstly signaling the GPM-MMVD syntax. When both two GPM-MMVD control flags are equal to false (i.e., the GPM-MMVD are disabled for two GPM partitions), the GPM-TM flag is signaled to indicate whether the template matching is applied to the two GPM partitions. Otherwise (at least one GPM-MMVD flag is equal to true), the value of the GPM-TM flag is inferred to be false.

2.1.10 Multi-hypothesis prediction (MHP)

The multi-hypothesis prediction is adopted in this contribution. Up to two additional predictors are signalled on top of inter AMVP mode, regular merge mode, and MMVD mode. The resulting overall prediction signal is accumulated iteratively with each additional prediction signal.

$$p_{n+1} = (1 - \alpha_{n+1})p_n + \alpha_{n+1}h_{n+1}$$

The weighting factor α is specified according to the following table:

add_hyp_weight_idx	α
0	1/4
1	-1/8

For inter AMVP mode, MHP is only applied if non-equal weight in BCW is selected in bi-prediction mode.

2.1.11 Decoder side intra mode derivation (DIMD)

- Three angular modes are selected from a Histogram of Gradient (HoG) computed from the neighboring pixels of current block. Once the three modes are selected, their predictors are computed normally and then their weighted average is used as the final predictor of the block. To determine the weights, corresponding amplitudes in the HoG are used for each of the three modes. The DIMD mode is used as an alternative prediction mode and is always checked in the FullRD mode.

Current version of DIMD has modified some aspects in the signaling, HoG computation and the prediction fusion. The purpose of this modification is to improve the coding performance as well

as addressing the complexity concerns raised during the last meeting (i.e. throughput of 4x4 blocks). The following sections describe the modifications for each aspect.

2.1.11.1 Signalling

Fig. 23 illustrates a schematic diagram 2300 of a proposed intra block decoding process. Fig. 23 shows the order of parsing flags/indices in VTM5, integrated with the proposed DIMD.

As can be seen, the DIMD flag of the block is parsed first using a single CABAC context, which is initialized to the default value of 154.

If flag = 0, then the parsing continues normally.

Else (if flag = 1), only the ISP index is parsed and the following flags/indices are inferred to be zero: BDPCM flag, MIP flag, MRL index. In this case, the entire IPM parsing is also skipped.

During the parsing phase, when a regular non-DIMD block inquires the IPM of its DIMD neighbor, the mode PLANAR_IDX is used as the virtual IPM of the DIMD block.

2.1.11.2 Texture analysis

Fig. 24 illustrates a schematic diagram 2400 of HoG computation from a template of width 3 pixels. The texture analysis of DIMD includes a Histogram of Gradient (HoG) computation (as shown in Fig. 24). The HoG computation is carried out by applying horizontal and vertical Sobel filters on pixels in a template of width 3 around the block. Except, if above template pixels fall into a different CTU, then they will not be used in the texture analysis.

Once computed, the IPMs corresponding to two tallest histogram bars are selected for the block.

In previous versions, all pixels in the middle line of the template were involved in the HoG computation. However, the current version improves the throughput of this process by applying the Sobel filter more sparsely on 4x4 blocks. To this aim, only one pixel from left and one pixel from above are used. This is shown in Fig. 24.

In addition to reduction in the number of operations for gradient computation, this property also simplifies the selection of best 2 modes from the HoG, as the resulting HoG cannot have more than two non-zero amplitudes.

2.1.11.3 Prediction fusion

This method uses a fusion of three predictors for each block. However, the choice of prediction modes is different and makes use of the combined hypothesis intra-prediction method, where the

Planar mode is considered to be used in combination with other modes when computing an intra-

predicted candidate. In the current version, the two IPMs corresponding to two tallest HoG bars are combined with the Planar mode.

The prediction fusion is applied as a weighted average of the above three predictors. To this aim, the weight of planar is fixed to 21/64 ($\sim 1/3$). The remaining weight of 43/64 ($\sim 2/3$) is then shared
5 between the two HoG IPMs, proportionally to the amplitude of their HoG bars.

Fig. 25 illustrates a schematic process 2500 of prediction fusion by weighted averaging of two HoG modes and planar.

2.1.12 Template-based intra mode derivation (TIMD)

A TIMD mode is derived from MPMs using the neighbouring template. The TIMD mode is used
10 as an additional intra prediction method for a CU.

2.1.12.1 TIMD mode derivation

For each intra prediction mode in MPMs, The SATD between the prediction and reconstruction samples of the template is calculated. The intra prediction mode with the minimum SATD is selected as the TIMD mode and used for intra prediction of current CU. Position dependent intra
15 prediction combination (PDPC) is included in the derivation of the TIMD mode.

2.1.12.2 TIMD signalling

A flag is signalled in sequence parameter set (SPS) to enable/disable the proposed method. When the flag is true, a CU level flag is signalled to indicate whether the proposed TIMD method is used. The TIMD flag is signalled right after the MIP flag. If the TIMD flag is equal to true, the
20 remaining syntax elements related to luma intra prediction mode, including MRL, ISP, and normal parsing stage for luma intra prediction modes, are all skipped.

2.1.12.3 Modification of MPM list construction in the derivation of TIMD mode

During the construction of MPM list, intra prediction mode of a neighbouring block is derived as Planar when it is inter-coded. To improve the accuracy of MPM list, when a neighbouring block
25 is inter-coded, a propagated intra prediction mode is derived using the motion vector and reference picture and used in the construction of MPM list. This modification is only applied to the derivation of the TIMD mode.

3 Problems

There are several issues in the existing video coding techniques, which would be further improved
30 for higher coding gain.

- (1) The combination of multiple hypothesis prediction (e.g., CIIP, MHP, and etc.) with other coding tools need to be carefully designed.
- (2) The coding methods for generating compositions for a multiple hypothesis prediction block need to be carefully designed.
- 5 (3) Currently, three types of motion would be stored for a GPM mode coded block, where a combined motion is stored for subblock locating at the blending area. Moreover, the storage of intra mode information is not defined if a GPM block contains both inter and intra sub-partitions/regions.

4 Detailed descriptions

- 10 The detailed descriptions below should be considered as examples to explain general concepts. These embodiments should not be interpreted in a narrow way. Furthermore, these embodiments can be combined in any manner.

The terms ‘video unit’ or ‘coding unit’ or ‘block’ may represent a coding tree block (CTB), a coding tree unit (CTU), a coding block (CB), a CU, a PU, a TU, a PB, a TB.

- 15 In this disclosure, regarding “a block coded with mode N”, here “mode N” may be a prediction mode (e.g., MODE_INTRA, MODE_INTER, MODE_PLT, MODE_IBC, and etc.), or a coding technique (e.g., AMVP, Merge, SMVD, BDOF, PROF, DMVR, AMVR, TM, Affine, CIIP, GPM, MMVD, BCW, HMVP, SbTMVP, and etc.).

- 20 A “multiple hypothesis prediction” in this disclosure may refer to any coding tool that combining/blending more than one prediction/composition/hypothesis into one for later reconstruction process. For example, a composition/hypothesis may be INTER mode coded, INTRA mode coded, or any other coding mode/method like CIIP, GPM, MHP, and etc.

In the following discussion, a “base hypothesis” of a multiple hypothesis prediction block may refer to a first hypothesis/prediction with a first set of weighting values.

- 25 In the following discussion, an “additional hypothesis” of a multiple hypothesis prediction block may refer to a second hypothesis/prediction with a second set of weighting values.

The compositions of multiple hypothesis prediction

1. In one example, mode X may NOT be allowed to generate a hypothesis of a multiple hypothesis prediction block coded with multiple hypothesis prediction mode Y.
- 30 1) For example, a base hypothesis of a multiple hypothesis prediction block may not be allowed to be coded by mode X.
- 2) For example, an additional hypothesis of a multiple hypothesis prediction block may not be allowed to be coded by mode X.

- 3) For example, for an X-coded block, it may never signal any block level coding information related to mode Y.
 - 4) For example, X is a palette coded block (e.g., PLT mode).
 - 5) Alternatively, mode X may be allowed to be used to generate a hypothesis of a multiple hypothesis prediction block coded with mode Y.
 - a) For example, X is a Symmetric MVD coding (e.g., SMVD) mode.
 - b) For example, X is based on a template matching based technique.
 - c) For example, X is based on a bilateral matching based technique.
 - d) For example, X is a combined intra and inter prediction (e.g., CIIP) mode.
 - e) For example, X is a geometric partition prediction (e.g., GPM) mode.
 - 6) Mode Y may be CIIP, GPM or MHP.
2. CIIP may be used together with mode X (such as GPM, or MMVD, or affine) for a block.
 - 1) In one example, at least one hypothesis in GPM is generated by CIIP. In other words, at least one hypothesis in GPM is generated as a weighted sum of at least one inter-prediction and one intra-prediction.
 - 2) In one example, at least one hypothesis in CIIP is generated by GPM. In other words, at least one hypothesis in CIIP is generated as a weighted sum of at least two inter-predictions.
 - 3) In one example, at least one hypothesis in CIIP is generated by MMVD.
 - 4) In one example, at least one hypothesis in CIIP is generated by affine prediction.
 - 5) In one example, whether mode X can be used together with CIIP may depend on coding information such as block dimensions.
 - 6) In one example, whether mode X can be used together with CIIP may be signaled from the encoder to the decoder.
 - a) In one example, the signaling may be conditioned by coding information such as block dimensions.
 3. In one example, one or more hypotheses of a multiple hypothesis prediction block may be generated based on position dependent prediction combination (e.g., PDPC).
 - 1) For example, prediction samples of a hypothesis may be processed by PDPC first, before it is used to generate the multiple hypothesis prediction block.
 - 2) For example, a predictor obtained based on PDPC which takes into account the neighboring sample values may be used to generate a hypothesis.
 - 3) For example, a predictor obtained based on gradient based PDPC which takes into account the gradient of neighboring samples may be used to generate a hypothesis.
 - a) For example, a gradient based PDPC may be applied to an intra mode (Planar, DC, Horizontal, Vertical, or diagonal mode) coded hypothesis.
 - 4) For example, a PDPC predictor may be not based on a prediction sample inside the current block.
 - a) For example, a PDPC predictor may be only based on prediction (or reconstruction) samples neighboring the current block.
 - b) For example, a PDPC predictor may be based on both prediction (or reconstruction) samples neighboring the current block and inside the current block.
 4. In one example, a multiple hypothesis predicted block may be generated based on decoder side refinement techniques.

- 1) For example, a decoder side refinement technique may be applied to one or more hypotheses of a multiple hypothesis prediction block.
- 2) For example, a decoder side refinement technique may be applied to a multiple hypothesis prediction block.
- 5 3) For example, the decoder side refinement technique may be based on decoder side template matching (e.g., TM), decoder side bilateral matching (e.g., DMVR), or decoder side bi-directional optical flow (e.g., BDOF) or Prediction Refinement with Optical Flow (PROF).
- 10 4) For example, the multiple hypothesis predicted block may be coded with CIIP, MHP, GPM, or any other multiple hypothesis prediction modes.
- 5) For example, the INTER prediction motion data of a multiple hypothesis block (e.g., CIIP) may be further refined by decoder side template matching (TM), and/or decoder side bilateral matching (DMVR), and/or decoder side bi-directional optical flow (BDOF).
- 15 6) For example, the INTER prediction samples of a multiple hypothesis block (e.g., CIIP) may be further refined by decoder side template matching (TM), and/or decoder side bilateral matching (DMVR), and/or decoder side bi-directional optical flow (BDOF) or Prediction Refinement with Optical Flow (PROF).
- 20 7) For example, the INTRA prediction part of a multiple hypothesis block (e.g., CIIP, MHP, and etc.) may be further refined by decoder side mode derivation (e.g., DIMD), decoder side intra template matching, and etc.
- 8) The refined intra prediction mode/motion information of a multiple hypothesis block may be disallowed to predict the following blocks to be coded/decoded in the same slice/tile/picture/subpicture.
- 25 9) Alternatively, decoder side refinement techniques may be NOT applied to a multiple hypothesis predicted block.
 - a) For example, decoder side refinement techniques may be NOT allowed to an MHP coded block.
- 30 5. For block-based multiple hypothesis prediction-coded blocks (e.g., coded with CIIP, MHP), it is proposed to derive the block into multiple subblocks/subpartitions/partitions
 - 1) In one example, multiple sets of motion information may be signalled/derived.
 - a) In one example, for each subblock/subpartition/partitions, one set of motion may be derived.
 - 35 2) In one example, the final prediction of a subblock/subpartition/partition may be dependent only on the set of motion information associated with it.
 - a) Alternatively, the final prediction of a subblock/subpartition/partition may be dependent only on more than one set of motion information associated with it.
- 40 6. In one example, in case that a multiple hypothesis prediction unit (e.g., coding unit) contains more than one subblock/subpartition/partition wherein the size of each subblock/subpartition/partition is less than the size of the entire multiple hypothesis prediction unit, the following rules may be applied:
 - 1) For example, the multiple hypothesis prediction unit may be partitioned in a uniform way.
 - a) For example, the multiple hypothesis prediction unit may be partitioned in to rectangular or square subblocks.
 - 45 b) For example, the multiple hypothesis prediction unit may be partitioned into $M \times N$ subblocks.
 - i. For example, $M = N$.
 - ii. For example, $M \neq N$.
 - iii. For example, $M = 4$ or 8 or 16 .

- iv. For example, $N = 4$ or 8 or 16 .
- v. For example, M is equal to the width of the entire multiple hypothesis prediction unit, and N is less than the height of the entire multiple hypothesis prediction unit.
- 5 vi. For example, M is less than the width of the entire multiple hypothesis prediction unit, and N is equal to the height of the entire multiple hypothesis prediction unit.
- c) For example, the multiple hypothesis prediction unit may be partitioned into triangle subblocks.
- 10 i. For example, the multiple hypothesis prediction unit may be partitioned into two diagonal triangles.
- 2) For example, the multiple hypothesis prediction unit may be partitioned in a nonuniform/irregular way.
 - a) For example, the multiple hypothesis prediction unit may be partitioned by an oblique line or a straight line (e.g., GPM partition, etc.).
 - 15 b) For example, the multiple hypothesis prediction unit may be partitioned by a curved line.
- 3) For example, whether a subblock/subpartition/partition/hypothesis of a multiple hypothesis prediction unit is intra-coded, may be dependent on the partition information of the multiple hypothesis prediction unit.
- 20 a) For example, it may depend on the angle of the partition line.
 - i. For example, which GPM partition is intra mode coded may be dependent on the GPM partition mode (or GPM partition angle, or GPM partition distance).
 - 25 ii. For example, one or more look-up-table (or mapping table) may be pre-defined for the corresponding relationship between the GPM partition mode (or GPM partition angle, or GPM partition distance) and which subblock/subpartition/partition/hypothesis is intra coded.
- b) For example, it may depend on the number of neighboring samples (outside the entire multiple hypothesis prediction unit) adjacent to the subblock/subpartition/partition/hypothesis (and this also depends on how the multiple hypothesis prediction unit is partitioned).
- 30 4) For example, in case that a subblock/subpartition/partition/hypothesis of the entire multiple hypothesis prediction unit is intra mode coded, what intra modes allowed for the subblock/subpartition/partition/hypothesis may be dependent on the partition information.
- 35 a) For example, whether to use horizontal intra mode, vertical intra mode, diagonal intra mode, or other intra mode may be dependent on the partition information of the multiple hypothesis prediction unit.
- b) For example, a pre-defined intra mode set may be defined depending on whether above and/or left neighbor samples are available for this subblock/subpartition/partition/hypothesis.
- 40 i. For example, horizontal or near horizontal intra modes may be not allowed when a subblock/subpartition/partition/hypothesis doesn't have left neighboring samples outside the entire multiple hypothesis coding unit but adjacent to the current subblock/subpartition/partition/hypothesis (the size of a subblock/subpartition/partition/hypothesis partition is less than the multiple hypothesis coding unit).
- 45 ii. For example, vertical or near vertical intra modes may be not allowed when a subblock/subpartition/partition/hypothesis doesn't have above neighboring

samples outside the entire multiple hypothesis coding unit but adjacent to the current subblock/subpartition/partition/hypothesis.

- c) For example, what intra modes are allowed for a GPM partition may be dependent on the GPM partition mode (or GPM partition angle, or GPM partition distance).
 - i. For example, a pre-defined intra mode set may be defined depending on the GPM partition shape/angle/distance/mode.
 - ii. For example, one or more look-up-table (or mapping table) may be pre-defined for the corresponding relationship between the GPM partition mode (or GPM partition angle, or GPM partition distance) and what intra modes are allowed for the intra coded subblock/subpartition/partition/hypothesis.
 - a) For example, at most one intra mode may be allowed for a GPM partition.
 - b) For example, a set of pre-defined intra modes may be allowed for a GPM partition.
 - iii. Additionally, what intra mode is used for a GPM partition may be dependent on the available neighboring samples outside the entire GPM coding unit but adjacent to the current GPM partition (the size of a GPM partition is less than the GPM coding unit).
 - a) For example, if a GPM partition doesn't have left neighboring samples but have above neighboring samples adjacent to the current GPM partition, horizontal or near horizontal intra modes which predicting from left to right may be allowed for the current GPM partition.
 - b) For example, if a GPM partition doesn't have above neighboring samples but have left neighboring samples adjacent to the current GPM partition, vertical or near vertical intra modes which predicting from up to down may be allowed for the current GPM partition.
 - c) For example, if a GPM partition have neither above neighboring samples nor left neighboring samples adjacent to the current GPM partition, intra mode be NOT allowed for the current GPM partition.
 - i. Alternatively, in such case, a specific intra mode other than horizontal/vertical/near-horizontal/near-vertical intra mode may be allowed for the current GPM partition.
- 5) In one example, the hypothesis prediction unit may not be partitioned into subblock/subpartition/partition in a sharp-cut way. Instead, the way of splitting subblock/subpartition/partition may be used to determine the weighting values for prediction samples in the unit.
 - a) A unit is partitioned into subblock/subpartition/partition in a sharp-cut way if it is partitioned in multiple subblocks/subpartitions/partitions and prediction samples for each subblock/subpartition/partition are derived independently.
 - b) A unit is NOT partitioned into subblock/subpartition/partition in a sharp-cut way if it is partitioned in multiple subblocks/subpartitions/partitions conceptually, but prediction samples for each subblock/subpartition/partition are NOT derived independently.
 - c) In one example, a first weighting value for a first prediction on a first position in a first subblock/subpartition/partition may be larger than a second weighting value for a first prediction on a second position in a second subblock/subpartition/partition.
 - i. For example, the first prediction may be intra-prediction, the first subblock/subpartition/partition may be regarded as an intra-coded subblock/subpartition/partition and the second subblock/subpartition/partition may be regarded as an intra-coded subblock/subpartition/partition.

- d) Alternatively, furthermore, indication of partitioning information is not signalled anymore in such case.
- 6) In one example, the derivation of weighting values used in multiple hypothesis prediction may depend on whether a hypothesis prediction unit (e.g., coding unit) contains more than one subblock/subpartition/partition.
 - a) In one example, the weighting values may be derived on the relative sample positions in each subblock/subpartition/partition.
 - i. In one example, a first weighting value on a first relative sample position in a first subblock/subpartition/partition, may be equal to a second weighting value on the same relative sample position in a second subblock/subpartition/partition.
 - b) Alternatively, the weighting values may be derived toward the relative sample positions in the whole hypothesis prediction unit.
 - c) In one example, different weighing values may be used for different dimensions of subblock/subpartition/partitions.
- 7) The partitioning/weighting values used in the multiple hypothesis prediction-coded blocks may depend on coded information, color component, color formats, etc. al.
 - a) In one example, the chroma components follow the partitioning rules applied to luma component.
 - i. Alternatively, the chroma components have different partitioning rules that are applied to luma component.
 - b) In one example, the chroma components follow the weighting value derivation rules applied to luma component.
 - i. Alternatively, furthermore, the weighting values applied to chroma components may be shared/derived from that for luma component.
- 8) The above methods may be also applied to those bullets mentioned in bullet 5.

CIIP/MHP inter components

- 7. For example, a virtual/generated motion data (e.g., including motion vectors, prediction directions, reference indices, etc.) may be used for multiple hypothesis prediction (e.g., CIIP, MHP, GPM, and etc.)
 - 1) The virtual/generated motion data may be generated in a basic-block by basic-block manner. For example, a basic-block may be a 4×4 block.
 - a) In one example, the motion data of a basic-block may depend on how the hypothesis prediction is conducted on this basic-block, such as the weighting values on this basic-block, the partitioning methods on this basic-block, the motion data of one prediction of the multiple hypothesis predictions on this basic-block and so on.
 - 2) For example, the prediction direction (L0, L1 or bi) may be derived according to pre-defined rules.
 - a) For example, if only motion information for L0 can be found in all hypothesis prediction for a basis-block, the prediction direction of the basis-block may be set to uni-prediction L0.
 - b) For example, if only motion information for L0 can be found in all hypothesis prediction for a basis-block, the prediction direction of the basis-block may be set to uni-prediction L1.
 - c) For example, if motion information for both directions can be found in all hypothesis prediction for a basis-block, the prediction direction of the basis-block may be set to bi.

- 3) For example, the virtual/generated motion may be a bi-predicted motion created according to pre-defined rules.
- a) For example, the virtual/generated BI-motion may be constructed from an L0 motion of a candidate from a first candidate list, and an L1 motion of a candidate from a second candidate list.
- i. For example, the first candidate list and/or the second candidate list may be pre-defined.
- ii. For example, the first candidate list may be AMVP candidate list, MERGE candidate list, a new candidate list constructed based on GPM/AMVP/MERGE candidates, or any other motion candidate lists.
- iii. For example, the second candidate list may be MERGE candidate list, AMVP candidate list, a new candidate list constructed based on GPM/AMVP/MERGE candidates, or any other motion candidate lists.
- iv. Additionally, the first candidate list is different from the second candidate list.
- v. Additionally, the first candidate list may be the same as the second candidate list.
- 4) For example, the virtual/generated motion may be a uni-predicted motion created following pre-defined rules.
- a) For example, the virtual/generated uni-motion may be constructed from L0 or L1 motion of a candidate from a third candidate list.
- i. For example, the third candidate list may be AMVP candidate list, MERGE candidate list, a new candidate list constructed based on GPM/AMVP/MERGE candidates, or any other motion candidate lists.
- 5) For example, if the L0/L1/BI motion is from a MERGE candidate list, a merge candidate index may be signalled.
- a) Alternatively, the merge candidate index may be implicitly derived from a decoder derived method (e.g., template matching based, or bilateral matching based, etc.)
- 6) For example, if the L0/L1/BI motion is from an AMVP candidate list, a motion vector difference (e.g., MVD) may be signalled.
- a) Additionally, an AMVP candidate index may be signalled.
- i. Alternatively, the AMVP candidate index may be implicitly derived from a decoder derived method (e.g., template matching based, or bilateral matching based, etc.)
- b) Alternatively, the motion vector difference may be implicitly derived from a decoder derived method (e.g., template matching based, or bilateral matching based, etc.)
- 7) For example, the virtual/generated motion data may be used to generate a prediction block, and the resultant prediction block may be used to compute the final prediction video unit (e.g., multiple hypothesis prediction block, a new coding mode).
- a) Additionally, a motion/sample refinement may be further applied to the generated prediction block.
- i. For example, the motion/sample refinement may be template matching (TM), bilateral matching, decoder derived motion vector refinement (e.g., DMVR), multi-pass decoder derived motion vector refinement (e.g., MPDMVR), BODF, PROF, and etc.
- 8) For example, the virtual/generated motion data may be used in succeeding procedures such as de-blocking process.
- 9) For example, the virtual/generated motion data may be used to predict motion data in succeeding blocks.

CIIP/MHP intra components

8. For example, the intra part of a multiple hypothesis prediction block (e.g., CIIP, MHP, GPM, etc) may be determined based on a pre-defined rule.

1) For example, the intra part of a multiple hypothesis prediction block (e.g., CIIP, MHP, GPM, etc) may be derived based on a fusion based intra prediction.

a) For example, the fusion based intra prediction may refer to a prediction block blended from more than one intra mode.

b) For example, the fusion based intra prediction may be generated by the first X intra modes from a pre-defined intra mode set.

i. For example, the first X (such as $X > 1$) intra modes may be the modes with lowest cost.

a) Furthermore, the cost may be calculated based on a template matching method, or a bilateral matching method.

i. For example, a template matching based method may be used to sort a set of pre-defined intra modes and select the best X modes as for the intra part of a multiple hypothesis block.

b) Furthermore, the cost may be calculated based on a quality metric (e.g., SAD/SATD/MSE, etc) using information of neighboring samples.

c) Furthermore, the cost may be calculated based on the histogram of gradient (HoG) from neighboring samples.

ii. For example, the pre-defined intra mode set may comprise Planar mode, and/or regular intra modes, and/or intra modes from MPM list, etc.

c) For example, weights for multiple prediction samples blending/fusion may be dependent on the intra prediction angles/directions.

i. Additionally, weights for multiple prediction samples blending/fusion may be dependent on the GPM partition modes, and/or GPM partition angles, and/or GPM partition distances.

d) For example, weights for multiple prediction samples blending/fusion may be block/partition/subblock based (e.g., different block/partition/subblock may have different weights).

i. Alternatively, weights for multiple prediction samples blending/fusion may be sample based (e.g., different weights may be assigned to different samples).

9. For example, the intra part of a multiple hypothesis prediction block (e.g., CIIP, MHP, GPM, etc) may be determined based on decoder-derived method.

1) In one example, it may be determined by decoder intra-prediction mode derivation (DIMD).

2) In one example, it may be determined by template-based intra-prediction mode derivation (TIMD).

Weighting factors design and storage

10. In one example, in case of blending an intra predicted sample with another prediction sample (could be inter coded, or intra code, or a prediction sample blended from others), what blending/fusion weights are used may be dependent on coding information.

1) For example, the rules for deriving blending weights may depend on the prediction modes of the samples being blended.

a) For example, different hypothesis combination (such as “intra + intra”, “intra + inter”, or “inter + inter”) may be different.

- 2) For example, the blending weights of intra and inter/intra may be dependent on the prediction mode of one of the intra predicted sample being used for blending/fusion.
- 3) For example, more than one set of blending/fusion weights may be defined for a specific fusion method, based on what intra mode is used for a video unit.
 - 5 a) For example, different weight sets may be defined based on the classification according to intra mode such as horizontal mode, vertical mode, wide-angle modes, diagonal mode, anti-diagonal mode, intra modes in which the samples are predicted from top and left neighboring samples (e.g., intra mode indices corresponding to angular greater than horizontal, intra mode index less than 18), intra modes in which
 - 10 the samples are predicted from top neighboring samples (e.g., intra mode indices corresponding to angular less than vertical, intra mode index greater than 50), intra modes in which the samples are predicted from left neighboring samples (e.g., intra mode index greater than horizontal (such as 18) but less than vertical (such as 50)), and etc.
 - 15 b) For example, the weight settings may be based on the rule of weights definition/classification in an existing coding tool such as PDPC, CIIP, and etc.
- 4) For example, more than one set of blending/fusion weights may be defined for a specific fusion method, based on which subblock/sub-unit the current sample belongs to.
 - 20 a) For example, different samples may have different weights.
 - b) For example, samples belong to different subblocks may have different weights.
 - c) For example, subblocks may be with non-rectangular shape.
- 5) The weighting values may depend on color components.
 - 25 a) In one example, weighting values on a first (such as chroma) component may be derive based on corresponding weighting values on a second (such as luma) component.
11. For example, intra mode information of a multiple hypothesis prediction block (e.g., GPM, MHP, CIIP, and etc.) may be stored in a basis of $M \times M$ unit (such as $M = 4$, or 8, or 16).
 - 30 a) For example, for an $M \times M$ unit locating at the blending area where all of the subblocks/subpartitions/partitions/hypotheses inside the $M \times M$ unit are INTRA coded, intra mode of which subblock/subpartition/partition/hypothesis is stored may depend on (i) the partition information (e.g., partition angle/distance/mode, etc.); (ii) the size of the subblock/subpartition/partition/hypothesis; (iii) the intra mode information; (iv) pre-defined rules.
 - 35 b) For example, for an $M \times M$ unit locating at the blending area which contain both intra coded and inter coded subblocks/subpartitions/partitions/hypotheses, whether to store motion data or the intra mode information, may be dependent on (i) pre-defined rule; (ii) the intra mode information; (iii) the inter motion data; (iv) the partition information (e.g., partition angle/distance/mode, etc.), (v) the size of the subblock/subpartition/partition/hypothesis.
 - 40 c) For example, the above-mentioned $M \times M$ unit based intra mode storage may be used to a multiple prediction mode which divides a coding unit into more than one subblock/subpartition/partition (e.g., GPM, and etc).
 - d) For example, the above-mentioned $M \times M$ unit based intra mode storage may be used to a multiple prediction mode which doesn't divide a coding unit into
 - 45 subblocks/subpartitions/partitions (e.g., CIIP, MHP, and etc).
 - e) For example, the above-mentioned $M \times M$ unit based intra mode storage may be used to predict intra-prediction mode in succeeding blocks.

General claims

12. Whether to and/or how to apply the disclosed methods above may be signalled at sequence level/group of pictures level/picture level/slice level/tile group level, such as in sequence header/picture header/SPS/VPS/DPS/DCI/PPS/APS/slice header/tile group header.
13. Whether to and/or how to apply the disclosed methods above may be signalled at PB/TB/CB/PU/TU/CU/VPDU/CTU/CTU row/slice/tile/sub-picture/other kinds of region contain more than one sample or pixel.

Whether to and/or how to apply the disclosed methods above may be dependent on coded information, such as block size, colour format, single/dual tree partitioning, colour component, slice/picture type.

In this disclosure, GPM specifies a prediction method that splits a coding unit into at least two subpartitons/partitions, and the splitting line may be an oblique line or a straight line. In addition, each partition of a GPM video unit may use an individual prediction method (e.g., intra, inter, non-inter, L0 prediction, or L1 prediction). Alternatively, at least two intermediate prediction blocks are generated with individual prediction methods, and a final prediction block is generated by a weighted sum of the intermediate prediction blocks, wherein the weighting values are determined based on the splitting method. On the other hand, the transform of a GPM video unit is conducted based on the entire video unit rather than subpartiton/partition. In yet another example, the GPM may generate multiple sets of motion information and the final prediction is based on weighted prediction signals from different sets of motion information; or it may generate the final prediction according to mixed prediction methods (e.g., intra/inter/palette/IBC).

GPM intra-intra prediction, GPM intra-inter prediction

14. In one example, the coding information storage of the intra part of a GPM intra-intra prediction (or a GPM intra-inter prediction) may follow below rules:

- 1) For example, the coding information may be stored on MxN basis.
 - a) For example, M = 2 or 4 or 8 luma samples.
 - b) For example, M may be equal to non-dyadic values.
 - c) For example, N = M.
- 2) For example, the coding information storage of the intra coded partition may be stored based on a zero MV.
- 3) For example, the coding information storage of the intra coded partition may be stored based on a reference index equal to -1.
- 4) For example, the coding information storage of the intra coded partition may be stored based on a reference index equal to a reference index of the current slice/picture.
- 5) For example, the coding information storage of the intra coded partition may be stored based on the real intra prediction mode/angle/direction that used to derive the intra prediction.
 - a) For example, the real intra prediction mode/angle/direction of a GPM partition may not belong to one of the regular intra mode index.

- i. For example, the real intra prediction mode/angle/direction of a GPM partition may be mapped to one of the regular intra mode index for coding information storage.
 - b) Alternatively, the real intra prediction mode/angle/direction that used to derive the intra prediction may not be stored.
 - i. For example, the coding information storage of the intra coded partition may be based on a default inter motion (such as zero MV).
 - c) Alternatively, the coding information storage of the intra coded partition may be based on a default intra mode (not necessarily the intra mode index used for the partition).
 - i. For example, the default intra mode may be Planar mode.
 - 6) For example, at the blending area (e.g., intra-inter fusion area that along the GPM partition line), whether to store intra coded information or inter coded information may be predefined.
 - a) In one example, if a sample belong to a blending area, then more than one weighting values used in GPM for that sample are not equal to 0.
 - b) For example, the intra coded information may be always stored.
 - c) For example, the inter coded information may be always stored.
 - d) Alternatively, whether to store intra or inter coded information may be dependent on the partition information (partition line, partition mode index, partition angle, partition distance, etc).
 - 7) For example, at the blending area (e.g., intra-intra fusion area that along the GPM partition line), the coded information of which partition is stored may be predefined.
 - a) In one example, if a sample belong to a blending area, then more than one weighting values used in GPM for that sample are not equal to 0.
 - b) Alternatively, whether to store the first or the second partition coded information may be dependent on the partition information (partition line, partition mode index, partition angle, partition distance, etc).
 - c) In one example, whether to store the first or the second partition coded information may be dependent on the two intra-prediction modes.
 - 8) For example, the stored coding information of a GPM intra partition may be used by succeeding coded/decoded blocks, such as for MPM list construction of a coding unit succeeding the current GPM block.
 - 9) For example, the stored coding information of a GPM intra partition may be used for deblocking process.
15. In one example, a multiple hypothesis prediction block may be generated based on more than one Intra prediction.
- 1) For example, more than one hypothesis of a multiple hypothesis prediction block (e.g., entire block based, or subblock/partition based) may be intra predicted.
 - a) For example, an MHP block may comprise more than one intra coded hypothesis.
 - b) For example, a CIIP block may comprise at least two intra predictions.
 - c) For example, both partitions of a GPM block may be intra mode coded.
 - d) For example, intra modes of the two partitions may be not allowed to the same.
 - e) For example, the intra mode of one of the two partitions may be signalled in the bitstream.
 - i. For example, the intra mode of the other partition may be implicitly derived.
 - ii. For example, the intra mode of a first partition may be excluded from the coded representation of the second partition.

- f) For example, intra modes of the two partitions may be signalled in the bitstream.
 - g) For example, intra modes of the two partitions may be implicitly derived.
 - h) For example, two intra predictions of the two partitions may be weighted blended.
 - i. For example, two intra predictions of a GPM block may be weighted blended.
 - ii. For example, all samples within a partition may have same weighting factor.
 - iii. For example, different samples may have different weighting factors.
 - iv. For example, the weighting values may depend on the splitting method of the GPM block.
 - v. For example, the weighting values may depend on at least one intra-prediction mode.
- 2) For example, the multiple hypothesis prediction block may be split by one or more oblique or straight partition lines (e.g., a GPM partition line).
- a) For example, the splitting modes (angle, direction, partition mode index) may be signalled in the bitstream.
 - b) For example, the splitting modes may be signalled in the same way as GPM partition mode index.
 - c) For example, the splitting modes may be implicitly derived based on coding information.
- 3) For example, one or more syntax elements (e.g., flag) may be signalled indicating whether the intra prediction of a certain (GPM) partition is derived at the decoder side.
- a) For example, a CU based flag may be signalled for the entire block.
 - b) For example, a partition-based flag may be signalled for a certain partition of the block.
 - c) For example, the decoder derived intra prediction may be DIMD, or TIMD, etc.
16. In one example, a multiple hypothesis prediction block may be allowed for P slice/picture wherein only L0 reference list is available, and/or B slice/picture wherein both L0 and L1 reference lists are available.
- 1) For example, GPM may be allowed for P slice/picture.
 - 2) For example, the two partitions of a GPM block may be an intra prediction and an inter prediction (a.k.a. GPM intra-inter block).
 - a) For example, the inter prediction may be L0 prediction or L1 prediction.
 - b) For example, the intra mode of the partition may be predefined or signalled.
 - 3) For example, the two partitions of a GPM block may be an intra prediction and another intra prediction.
 - a) For example, the intra modes of the two partitions may not be allowed to be the same.
 - 4) For example, the two partitions of a GPM block may be an inter prediction and another inter prediction.
 - a) For example, a GPM block may comprise two L0 predictions.
 - b) For example, a GPM block may comprise two L1 predictions.
 - c) For example, the motion information (e.g., merge index, motion vector, reference index, etc) of the two predictions may not be allowed to be the same.
 - d) For example, when the two partitions are predicted from a same prediction direction (e.g., L0 or L1), the motion vector of the two partitions may be added together or averaged for blended area motion storage.
 - i. For example, if the prediction direction and the reference index of the two partitions are same, motion vectors of the two partitions may be directly added together or averaged for motion storage of the blended area.

- ii. For example, if the prediction direction of the two partitions are same but the reference indexes are different, motion storage of the blended are may be based on a motion vector scaling process.
- e) For example, when the two partitions are predicted from a same prediction direction (e.g., L0 or L1), the motion vector of the partition with a smaller reference index may be stored.
 - f) For example, when the two partitions are predicted from a same prediction direction (e.g., L0 or L1), the motion vector of the partition with a smaller $|MV_x|+|MV_y|$ may be stored.
- 5) For example, a GPM candidate list may be constructed based on regular merge candidates who has a specific prediction direction such as L0.
- 6) For example, a GPM candidate list for P slice may be constructed in a different way of the GPM candidate list for B slice.
 - a) For example, a GPM candidate list for P slice may be a subset of the GPM candidate list for B slice.
- 17. For example, whether CU based GPM template matching syntax elements (e.g., a flag) are signalled or not may be dependent/conditioned on whether intra-inter coding (e.g., one partition is intra coded and the other partition is inter coded) is used for a GPM block.
 - 1) For example, in case that a GPM block is coded by intra-inter prediction, the CU based GPM template matching (in which both GPM partitions are refined by template matching) may be not allowed to be further applied.
 - a) For example, in case that CU based GPM template matching is not allowed for a GPM intra-inter block, the CU level TM based flag is not signalled by inferred to a certain value.
 - 2) Alternatively, whether intra-inter coding is allowed for a GPM block may be dependent/conditioned on whether CU based GPM template matching is used for the block.
 - a) For example, in case that a CU based GPM template matching is used, the GPM intra-inter prediction may be not allowed to be further applied.
 - b) For example, in case that the GPM intra-iter prediction is not allowed for a GPM block, the intra coded information is not signalled in the bitstream.
 - 3) Alternatively, a GPM intra-inter block may be allowed to use partition-based GPM template matching (in which the inter coded GPM partition is allowed to be refined by template matching).
 - a) For example, in case that a partition-based GPM template matching is allowed for a GPM intra-inter block (the partition-based GPM template matching is allowed to be applied to the inter coded partition), a flag may be signalled for the inter coded partition specifying whether the motion of the partition is further refined by template matching.
- 18. In one example, a multiple hypothesis prediction block may be allowed for I slice/picture.
 - 1) For example, GPM may be allowed for I slice/picture.
 - a) For example, a GPM block may comprise two non-Inter predictions.
 - b) For example, the non-Inter prediction may be intra prediction, IBC, or Palette prediction.
 - c) For example, different intra modes may be used for the two partitions of a GPM block.

- d) For example, sample-based weighting factor may be used to blend/fusion the two partitions of a GPM block.
- 2) For example, CIIP may be allowed for I slice/picture.
 - a) For example, a CIIP block may comprise an Intra prediction and a non-Inter prediction.
 - b) For example, the non-Inter prediction may be intra prediction, IBC, or Palette prediction.
 - c) For example, different intra modes may be used for the two predictions of a CIIP block.
 - d) For example, block-based weighting factor may be used to blend/fusion the two predictions of a CIIP block.
- 3) For example, MHP may be allowed for I slice/picture.
 - a) For example, an MHP block may comprise multiple non-Inter predictions.
 - b) For example, the non-Inter prediction may be intra prediction, IBC, or Palette prediction.
 - c) For example, different intra modes may be used for the multiple hypotheses of an MHP block.
 - d) For example, block-based weighting factor may be used to blend/fusion the multiple hypotheses of an MHP block.
- 4) Information for IBC such as BV may be signaled if IBC is involved in GPM/CIIP/MHP.
- 5) Information for Palette such as palette indices may be signaled if Palette is involved in GPM/CIIP/MHP.

Misc.

- 19. In one example, for a specific coding method, the shape of a template used for a video unit may be dependent on the availability of neighboring samples.
 - 1) When above samples are available but left samples are not available (e.g., template exceed the picture left boundary, or the current block locates at the first row of the picture), a template comprises above samples only.
 - 2) When left samples are available but above samples are not available (e.g., template exceed the picture above boundary, or the current block locates at the first column of the picture), a template comprises left samples only.
 - 3) When left samples and above samples are not available (e.g., the current block locates at the first row and first column of the picture), no template is used.
 - 4) Alternatively, a virtual template may be used, in which at least one sample of the template is generated by a specific mean (such as fill with a default sample value dependent on the internal bit depth).
 - a) In one example, padding may be utilized to fill in samples which are unavailable.
 - 5) The template may be used for template matching based MV/BV derivation.
 - 6) The template may be used for template matching based intra-prediction derivation.
- 20. In one example, filter coefficients, clipping values may be allowed to be a value not equal to a power of 2.
 - 1) In one example, the filter coefficients of CCALF may be based on a value not equal to a power of 2.
 - 2) In one example, clipping values (e.g., non-linear clipping in ALF, etc) of a certain coding tool may not be a power of 2.

21. In one example, chroma and luma may share similar filter shape.

1) The filter shape may be the same, however, the filter length may be different.

2) For example, assume $M \times N$ diamond/cross shape filter is used for the luma components of a loop filter (e.g., ALF, CCALF, etc), its associated chroma components may be allowed to use a similar diamond/cross shape filter with a size of $(M \gg \text{SubWidthC}) \times (N \gg \text{SubHeightC})$, wherein SubWidthC and SubHeightC depending on the chroma format sampling structure.

a) For example, SubWidthC = SubHeightC = 2 for 4:2:0 chroma format.

b) For example, SubWidthC = SubHeightC = 1 for 4:4:4 chroma format.

c) For example, SubWidthC = 2 and SubHeightC = 1 for 4:2:2 chroma format.

3) In one example, chroma and luma may share same filter shape if the chroma format is 4:4:4.

General claims

22. Whether to and/or how to apply the disclosed methods above may be signalled at sequence level/group of pictures level/picture level/slice level/tile group level, such as in sequence header/picture header/SPS/VPS/DPS/DCI/PPS/APS/slice header/tile group header.

23. Whether to and/or how to apply the disclosed methods above may be signalled at PB/TB/CB/PU/TU/CU/VPDU/CTU/CTU row/slice/tile/sub-picture/other kinds of region contain more than one sample or pixel.

Whether to and/or how to apply the disclosed methods above may be dependent on coded information, such as block size, colour format, single/dual tree partitioning, colour component, slice/picture type.

It is noted that the terminologies mentioned below are not limited to the specific ones defined in existing standards. Any variance of the coding tool is also applicable. For example, the term 'GPM' may represent a coding method that split one block into two or more sub-regions wherein at least one sub-region couldn't be generated by any of existing partitioning structure (e.g., QT/BT/TT).

In another example, the term 'GPM' may represent a kind of coding block, in which at least one final prediction signal of the coding block is generated by a weighted sum of two or more auxiliary prediction signals associated with the GPM sub-regions. For example, the term 'GPM' may indicate the geometric merge mode (GEO), and/or geometric partition mode (GPM), and/or wedge prediction mode, and/or triangular prediction mode (TPM), and/or a GPM block with motion vector difference (GPM MMVD), and/or a GPM block with template matching based motion refinement (GPM TM), and/or GPM with inter and intra, and/or any variant based on GPM.

Fig. 26 illustrates an example diagram 2600 of subblock based motion/mode information storage of a GPM coded block. In Fig. 26, the prediction samples within dark subblocks (across the GPM splitting line) are blended from sub-region-A and sub-region-B.

Motion/Mode storage for GPM coded blocks

24. In one example, in case that a GPM subblock contains both inter and intra predicted samples (e.g., illustrated as the dark subblocks in Fig. 26, suppose sub-region-A is inter coded, and sub-region-B is intra coded), the subblock may be treated as intra coded subblock in the coding of subsequent video blocks and/or in-loop filtering process.

1) In one example, the motion stored for such GPM subblock may always be perceived as unavailable.

2) For example, the motion vector is stored as zero vector, and the reference index is stored as a certain value (such as -1) indicating there is no reference picture for this subblock.

3) Alternatively, the motion information stored for such GPM subblock may always be equal to the motion information of the inter-coded-sub-region.

a) For example, no matter which of the two sub-regions is inter coded, the motion information stored for such GPM subblock may always be equal to the motion information of the inter-coded sub-region.

4) Alternatively, adaptive/selective motion information storage may be applied to such GPM subblock.

a) For example, whether the stored motion is perceived as unavailable, or equal to the motion of the inter-coded-sub-region, may be dependent on the coded information.

i. Furthermore, the coded information includes but not limited to splitting information (such as GPM partition mode, and/or GPM partition angle, and/or GPM partition direction), and/or weight index, and/or the GPM block/subblock location, and/or the GPM block/subblock dimensions.

5) For example, the stored motion information may be used for succeeding process of the current GPM block such as deblocking.

6) For example, the stored motion information may be used as temporal motion information for future blocks coding/prediction, wherein the future blocks are within succeeding coded pictures in coding order.

7) For example, the stored motion information may be used as spatial motion information for future blocks coding/prediction, wherein the future blocks are within the current picture.

8) For example, the stored motion information may be used for loop-filtering, such as deblocking filtering.

25. In one example, the inter-prediction process to generate the prediction samples of the inter-coded sub-region may follow a rule elaborated below.

1) For example, the inter-coded sub-region may always be uni-directional predicted.

2) Alternatively, the inter-coded sub-region may be bi-directional predicted.

3) Alternatively, furthermore, the inter-coded sub-region may be uni-directional predicted.

4) For example, the above rule may be applicable in case that at least one GPM subblock of the whole GPM block contains both inter and intra predicted samples.

a) Alternatively, furthermore, the above rule may be applicable in case that at least one GPM subblock of the whole GPM block contains intra predicted samples.

- b) Alternatively, furthermore, the above rule may be applicable in case that at least one GPM subblock of the whole GPM block contains inter predicted samples.

26. In one example, in case that a GPM subblock contains both inter and intra predicted samples (e.g., illustrated as the dark subblocks in Fig. 26, suppose sub-region-A is inter coded, and sub-region-B is intra coded), the intra mode information stored for such GPM subblock may always be equal the intra mode information of the intra-coded-sub-region.

1) For example, no matter which of the two sub-regions is intra coded, the intra mode information stored for such GPM subblock may always be equal to the intra mode information of the intra-coded sub-region.

2) Alternatively, the intra mode information stored for such GPM subblock may always be perceived as unavailable.

3) Alternatively, adaptive/selective intra mode information storage may be applied to such GPM subblock.

a) For example, whether the stored intra mode is perceived as unavailable, or equal to the intra mode of the intra-coded-sub-region, may be dependent on the coded information.

i. Furthermore, the coded information includes but not limited to splitting information (such as GPM partition mode, and/or GPM partition angle, and/or GPM partition direction), and/or weight index, and/or the GPM block/subblock location, and/or the GPM block/subblock dimensions.

4) For example, the stored intra mode information may be used for succeeding process of the current GPM block such as deblocking.

5) For example, the stored intra mode information may be used as temporal intra mode information for future blocks coding/prediction (such as TIMD), wherein the future blocks are within succeeding coded pictures in coding order.

6) For example, the stored intra mode information may be used as spatial intra mode information for future blocks coding/prediction, wherein the future blocks are within the current picture.

27. In one example, in case that the prediction of a GPM subblock is blended from more than one inter predicted samples (e.g., illustrated as the dark subblocks in Fig. 26, suppose sub-region-A is inter coded, and sub-region-B is also inter coded), the motion stored for such GPM subblock may always be equal to the motion information of one sub-region.

1) For example, the motion stored for such GPM subblock may comprise motion information from at most one of the two sub-regions.

a. Furthermore, the stored motion information of such GPM subblock may always be uni-directional prediction.

i. Alternatively, the stored motion information of such GPM subblock may be bi-directional predicted (e.g., in case that sub-region-A is bi-directional predicted or sub-region-B is bi-directional predicted).

ii. Alternatively, the stored motion information of such GPM subblock may be uni-directional predicted.

b. Furthermore, in total two types of motion (such as sub-region-A motion, or sub-region-B motion) may be stored for such GPM subblock. The motion storage of such GPM subblock may NOT/NEVER be of a third type such as combining/constructing from both sub-region-A motion and sub-region-B motion.

- 2) For example, whether to store sub-region-A motion or sub-region-B motion (but never a combined one from both motion) for such GPM subblock may be dependent on coding information.

i. Furthermore, the coded information includes but not limited to splitting information (such as GPM partition mode, and/or GPM partition angle, and/or GPM partition direction), and/or weight index, and/or the GPM block/subblock location, and/or the GPM block/subblock dimensions.

28. In one example, in case that the prediction of a GPM subblock is blended from more than one intra predicted samples (e.g., illustrated as the dark subblocks in Fig. 26, suppose sub-region-A is intra coded, and sub-region-B is also intra coded), the intra mode information stored for such GPM subblock may always be equal to the intra mode information of one sub-region.

1) For example, the intra mode information stored for such GPM subblock may always be equal to the intra mode information of sub-region-A.

2) For example, the intra mode information stored for such GPM subblock may always be equal to the intra mode information of sub-region-B.

3) For example, whether to store sub-region-A intra mode information or sub-region-B intra mode information (but never a combined one from both) for such GPM subblock may be dependent on coding information.

i. Furthermore, the coded information includes but not limited to splitting information (such as GPM partition mode, and/or GPM partition angle, and/or GPM partition direction), and/or weight index, and/or the GPM block/subblock location, and/or the GPM block/subblock dimensions.

4) Alternatively, a constructed/converted/mapped intra mode may be stored for such GPM subblock.

5) Alternatively, more than one intra mode may be stored for such GPM subblock.

a) For example, both the intra mode information of sub-region-A and the intra mode information of sub-region-B may be stored for such GPM subblock.

29. For example, the aforementioned GPM block may be a GPM coded block without motion refinement.

30. For example, the aforementioned GPM block may be a GPM coded block with motion refinement.

31. For example, the aforementioned GPM block may be a GPM MMVD block.

32. For example, the aforementioned GPM block may be a GPM TM (template matching) block.

5 Embodiment

[0086] The embodiments of the present disclosure are related to a geometric partitioning tool. As used herein, the term of “geometric partitioning tool” or “geometric partition mode (GPM) tool” refers to any coding tool that splits one block into two or more sub-regions wherein at least one sub-region couldn’t be generated by any of existing partitioning structure (e.g., QT/BT/TT). For example, the term ‘GPM’ may represent a kind of coding block, in which at least one final

prediction signal of the coding block is generated by a weighted sum of two or more auxiliary prediction signals associated with the GPM sub-regions. For example, the term ‘GPM’ may indicate the geometric merge mode (GEO), and/or geometric partition mode (GPM), and/or wedge prediction mode, and/or triangular prediction mode (TPM), and/or a GPM block with motion vector difference (GPM MMVD), and/or a GPM block with template matching based motion refinement (GPM TM), and/or GPM with inter and intra, and/or any variant based on GPM.

[0087] It is noted that the terminologies mentioned below are not limited to the specific ones defined in existing standards. Any variance of the coding tool is also applicable.

[0088] As used herein, the term “block” may represent a coding block (CB), a coding unit (CU), a prediction unit (PU), a transform unit (TU), a prediction block (PB), a transform block (TB).

[0089] Fig. 27 illustrates a flowchart of a method 2700 for video processing in accordance with some embodiments of the present disclosure. The method 2700 may be implemented during a conversion between a target video block of a video and a bitstream of the video. As shown in Fig. 27, the method 2700 starts at 2702, where prediction information of a target subblock in the target video block is obtained based on types of predicted samples in the target subblock. The target video block is coded by a geometric partitioning tool. For example, the obtained prediction information may be stored. The geometric partitioning tool may comprise a geometric merge mode (GEO), a geometric partition mode (GPM), a wedge prediction mode, a triangular prediction mode, a GPM with motion vector difference (GPM MMVD), GPM block with template matching based motion refinement (GPM TM), a GPM with inter and intra, or a variant coding tool based on GPM.

[0090] As used herein, the term of “target video block” may also be referred to as “geometric partitioned block” or “GPM coded block”. For example, the target video block may comprise a geometric partition mode (GPM) coded block without motion refinement, a GPM coded block with motion refinement, a GPM block with motion vector difference (GPM MMVD), or a GPM block with template matching (GPM TM).

[0091] At block 2704, the conversion is performed based on the prediction information of the target subblock. For example, the conversion may include encoding the target video block into the bitstream. For another example, the conversion may include decoding the target video block from the bitstream.

[0092] According to embodiments of the present disclosure, it is proposed that prediction information, such as motion information or intra mode information of a subblock in a target block may be obtained. The obtained prediction information of the subblock may be stored for further coding or prediction process. In this way, the prediction effectiveness and coding efficiency may be improved.

[0093] In some embodiments, if the target subblock comprises an intra predicted sample and an inter predicted sample, intra coded information of the target subblock may be used in at least one of: a coding of a subsequent video block, or an in-loop filtering process. For example, the subblocks across the GPM splitting line as shown in Fig. 26 comprise both intra predicted samples and inter predicted samples. In other words, the target subblock may be treated as an intra coded subblock in the coding of subsequent video blocks and /or in-loop filtering process.

[0094] In some embodiments, at block 2702, if the target subblock comprises an intra predicted sample and an inter predicted sample, motion information of the target subblock may be obtained. Alternatively, or in addition, the obtained motion information may be stored. In some embodiments, the motion information may be unavailable. For example, the motion information may be always perceived as unavailable.

[0095] In some embodiments, the motion information may comprise a zero vector with a corresponding reference index. The corresponding reference index indicates no reference picture being for the target subblock. That is, the motion vector may be stored as zero vector, and the reference index may be stored as a certain value (such as -1) indicating there is no reference picture for this subblock.

[0096] Alternatively, or in addition, in some embodiments, at block 2702, the motion information of the inter predicted sample of the target subblock may be obtained. For example, in some embodiments, the target subblock may comprise an intra region and an inter region. The predicted sample in the intra region is intra predicted, and the predicted sample in the inter region is inter predicted. In such cases, at block 2702, the motion information of the inter region may be obtained. In other words, the motion information stored for the GPM subblock may always be equal the motion information of the inter-coded sub-region.

[0097] In some embodiments, an adaptive or selective motion information storage may be applied to the target subblock. Alternatively, or in addition, whether the motion information for the target subblock is unavailable or equal to motion information of the inter predicted sample

may be determined based on coded information for the target video block. The motion information for the target subblock may be stored based on the determination.

[0098] For example, the coded information for the target video block comprises at least one of: splitting information, a weight index, a location of a GPM block or GPM subblock, or a dimension of the GPM block or GPM subblock. The splitting information may comprise at least one of: a GPM partition mode, a GPM partition angle, or a GPM partition direction.

[0099] In some embodiments, at block 2704, the motion information may be used for a succeeding process in the conversion. The succeeding process may comprise a deblocking process. That is, the obtained motion information or the stored motion information may be used for succeeding process of the current GPM block such as deblocking. Alternatively, or in addition, in some embodiments, at block 2704, the motion information may be used for a loop-filtering in the conversion, such as a deblocking filtering.

[00100] In some embodiments, the obtained motion information may be used as temporal motion information for coding or predicting a further block. The further block may be within succeeding coded pictures of the video in a coding order. For example, the stored motion information may be used as temporal motion information for future blocks coding/prediction, where the future blocks are within succeeding coded pictures in coding order.

[00101] Alternatively, or in addition, in some embodiments, the obtained motion information may be used as spatial motion information for coding or predicting a further block. The further block may be within a current picture of the video. That is to say, the stored motion information may be used as spatial motion information for future blocks coding/prediction, where the future blocks are within the current picture.

[00102] In some embodiments, an inter predicted sample of an inter-coded region of the target video block may be generated based on a predetermined rule. In other words, the inter prediction process to generate the prediction samples of the inter-coded sub-region may follow a rule. For example, the predetermined rule may comprise at least one of: a rule indicating to uni-directionally predict the inter-coded region, or a rule indicating to bi-directionally predict the inter-coded region. For example, the inter-coded sub-region may always be uni-directional predicted. Alternatively, the inter-coded sub-region may be bi-directional predicted. Alternatively, or in addition, the inter-coded sub-region may be uni-directional predicted. In some embodiments, if at least one subblock of the target video block comprises at least one of an inter predicted sample

or an intra predicted sample, the at least one subblock comprising the target subblock, the inter predicted sample may be generated based on the predetermined rule. That is, the above rule may be applicable in case that at least one GPM subblock of the whole GPM block contains both inter and intra predicted samples. The above rule may also be applicable in case that at least one GPM subblock of the whole GPM block contains intra predicted sample. Alternatively, the above rule may be applicable in case that at least one GPM subblock of the whole GPM block contains inter predicted samples.

[00103] In some embodiments, at block 2702, if the target subblock comprises an intra predicted sample and an inter predicted sample, intra mode information of the intra predicted sample may be obtained for the target subblock. For example, intra mode information of an intra-coded region of the target subblock may be obtained. The intra-coded region is intra-coded. In other words, no matter which of the two sub-regions is intra coded, the intra mode information stored for the GPM subblock may always be equal to the intra mode information of the intra-coded sub-region. In some embodiments, the obtained intra mode information may be stored.

[00104] In some embodiments, the intra mode information is unavailable. For example, the intra mode information for the target subblock may be always perceived as unavailable.

[00105] Alternatively, or in addition, in some embodiments, an adaptive or selective intra mode information storage may be applied to the target subblock. For example, in some embodiments, whether the intra mode information for the target subblock is unavailable or equal to intra mode information of the intra predicted sample may be determined based on coded information for the target video block. The intra mode information for the target subblock may be stored based on the determination. For example, the coded information for the target video block comprises at least one of: splitting information, a weight index, a location of a GPM block or GPM subblock, or a dimension of the GPM block or GPM subblock. The splitting information may comprise at least one of: a GPM partition mode, a GPM partition angle, or a GPM partition direction.

[00106] In some embodiments, at block 2704, the intra mode information may be used for a succeeding process in the conversion. For example, the succeeding process may comprise a deblocking process. That is, the intra mode information such the stored intra mode information may be used for succeeding process of the current GPM block such as deblocking.

[00107] Alternatively, or in addition, in some embodiments, the obtained intra mode information may be used as temporal intra mode information for coding or predicting a further

block. The further block may be within succeeding coded pictures of the video in a coding order. Coding or predicting the further block may comprise applying a Template-based intra mode derivation (TIMD). For example, the stored intra mode information may be used as temporal intra mode information for future blocks coding or prediction such as TIMD. The future blocks are within succeeding coded pictures in coding order.

[00108] In some embodiments, the obtained intra mode information may be used as spatial intra mode information for coding or predicting a further block, the further block being within a current picture of the video. For example, the stored intra mode information may be used as spatial intra mode information for future blocks coding or prediction. The future blocks are within the current picture.

[00109] In some embodiments, at block 2702, if the predicted samples comprise at least two inter predicted samples, motion information for the target subblock may be obtained. In some embodiments, the obtained motion information may be stored.

[00110] In some embodiments, motion information may be obtained in an inter-coded region. An inter predicted sample is coded in the inter-coded region. For example, the inter-coded region is one of two inter-coded regions for the target subblock. In other words, the motion information stored for the target subblock may comprise motion information from at most one of the two sub-regions.

[00111] In some embodiments, the motion information may comprise at least one of: a uni-directional prediction, or a bi-directional prediction. For example, the stored motion information for the target subblock may always be uni-directional predicted. Alternatively, the stored motion information for the target subblock may be bi-directional predicted. For example, in case at least one inter-coded region of the target subblock being bi-directional predicted, the motion information comprises a bi-directional prediction. Alternatively, the stored motion information for the target subblock may be uni-directional predicted.

[00112] Alternatively, or in addition, in some embodiments, two types of motion information may be obtained for the target subblock. Each of the two types of motion information is associated with a respective inter-coded region of the target subblock. In other words, in total two types of motion (such as sub-region-A motion, or sub-region-B motion in Fig. 26) may be stored for the target GPM subblock. In some embodiments, a third type of motion information may be absent from the motion information for the target subblock. For example, the third type of motion

information may comprise combining or constructing motion information from first motion information in a first inter-coded region and second motion information in a second inter-coded region. In other words, the motion storage of the target GPM subblock may not or never be of a third type such as combining or constructing from both sub-region-A motion and sub-region-B motion.

[00113] In some embodiments, whether to store first motion information of a first inter-coded region of the target subblock or second motion information of a second inter-coded region of the target subblock may be determined based on coded information for the target subblock. Respective motion information may be stored based on the determination. In some embodiments, combined motion information of the first and second motion information is absent from the stored motion information. For example, the coded information for the target video block may comprise at least one of: splitting information, a weight index, a location of a GPM block or GPM subblock, or a dimension of the GPM block or GPM subblock. The splitting information may comprise at least one of: a GPM partition mode, a GPM partition angle, or a GPM partition direction.

[00114] In some embodiments, at block 2702, if the predicted samples comprise at least two intra predicted samples, intra mode information for the target subblock may be obtained. In some embodiments, the obtained intra mode information may be stored. For example, the intra mode information in an intra-coded region may be obtained. An intra predicted sample is coded in the intra-coded region. The intra-coded region may be one of two intra-coded regions for the target subblock.

[00115] In some embodiments, whether to store first intra mode information of a first intra-coded region of the target subblock or second intra mode information of a second intra-coded region of the target subblock may be determined based on coded information for the target subblock. Respective intra mode information may be stored based on the determination. In some embodiments, combined intra mode information of the first and second intra mode information may be absent from the stored intra mode information. That is, a combined one from both of the first intra mode information and the second intra mode information may never be stored. In some embodiments, the coded information for the target video block may comprise at least one of: splitting information, a weight index, a location of a GPM block or GPM subblock, or a dimension of the GPM block or GPM subblock. The splitting information may comprise at least one of: a GPM partition mode, a GPM partition angle, or a GPM partition direction.

[00116] Alternatively, or in addition, in some embodiments, to obtain the intra mode information, at least one of the following may be obtained for the target subblock: a constructed intra mode, a converted intra mode, or a mapped intra mode. In other words, a constructed or converted or mapped intra mode may be stored for the current GPM subblock.

5 [00117] In some embodiments, intra mode information of more than one intra-coded region of the target subblock may be obtained. For example, the more than one intra-coded region may comprise a first intra-coded region and a second intra-coded region. In other words, both the intra mode information of sub-region-A and the intra mode information of sub-region-B may be stored for the target subblock.

10 [00118] In some embodiments, a bitstream of a video may be stored in a non-transitory computer-readable recording medium. The bitstream of the video can be generated by a method performed by a video processing apparatus. According to the method, prediction information of a target subblock in a target video block of the video may be obtained based on types of predicted samples in the target subblock. The target video block may be coded by a geometric partitioning
15 tool. The bitstream may be generated based on the prediction information of the target subblock.

[00119] In some embodiments, prediction information of a target subblock in a target video block of the video may be obtained based on types of predicted samples in the target subblock. The target video block may be coded by a geometric partitioning tool. A bitstream of the video may be generated based on the prediction information of the target subblock. The bitstream may
20 be stored in a non-transitory computer-readable recording medium.

[00120] According to embodiments of the present disclosure, it is proposed that the prediction information for a subblock in the target video block may be obtained and stored. By using the prediction information such as motion information or intra mode information, the coding process can be improved, thus the effectiveness of the target video block prediction and the coding
25 efficiency may be improved, as well.

[00121] Implementations of the present disclosure can be described in view of the following clauses, the features of which can be combined in any reasonable manner.

[00122] Clause 1. A method for video processing, comprising: obtaining, during a conversion between a target video block of a video and a bitstream of the video, prediction information of a
30 target subblock in the target video block based on types of predicted samples in the target subblock,

the target video block being coded by a geometric partitioning tool; and performing the conversion based on the prediction information of the target subblock.

5 [00123] Clause 2. The method of clause 1, further comprising: if the target subblock comprises an intra predicted sample and an inter predicted sample, using intra coded information of the target subblock in at least one of: a coding of a subsequent video block, or an in-loop filtering process.

[00124] Clause 3. The method of clause 1 or clause 2, wherein obtaining the prediction information of the target subblock based on the types of the predicted samples comprises: if the target subblock comprises an intra predicted sample and an inter predicted sample, obtaining motion information of the target subblock.

10 [00125] Clause 4. The method of clause 3, wherein the motion information is unavailable.

[00126] Clause 5. The method of clause 3, wherein the motion information comprises a zero vector with a corresponding reference index, the corresponding reference index indicating no reference picture being for the target subblock.

15 [00127] Clause 6. The method of clause 3, wherein obtaining the motion information comprises: obtaining the motion information of the inter predicted sample of the target subblock.

[00128] Clause 7. The method of clause 3, wherein the target subblock comprises an intra region and an inter region, the predicted sample in the intra region being intra predicted, and the predicted sample in the inter region being inter predicted, and wherein obtaining the motion information comprises: obtaining the motion information of the inter region.

20 [00129] Clause 8. The method of any of clauses 3-7, further comprising: applying an adaptive or selective motion information storage to the target subblock.

25 [00130] Clause 9. The method of any of clauses 3-8, further comprising: determining, based on coded information for the target video block, whether the motion information for the target subblock is unavailable or equal to motion information of the inter predicted sample; and storing the motion information for the target subblock based on the determination.

[00131] Clause 10. The method of clause 9, wherein the coded information for the target video block comprises at least one of: splitting information, a weight index, a location of a GPM block or GPM subblock, or a dimension of the GPM block or GPM subblock.

[00132] Clause 11. The method of clause 10, wherein the splitting information comprises at least one of: a GPM partition mode, a GPM partition angle, or a GPM partition direction.

[00133] Clause 12. The method of any of clauses 3-11, wherein performing the conversion comprises: using the motion information for a succeeding process in the conversion.

5 [00134] Clause 13. The method of clause 12, wherein the succeeding process comprises a deblocking process.

[00135] Clause 14. The method of any of clauses 3-11, wherein the obtained motion information is used as temporal motion information for coding or predicting a further block, the further block being within succeeding coded pictures of the video in a coding order.

10 [00136] Clause 15. The method of any of clauses 3-11, wherein the obtained motion information is used as spatial motion information for coding or predicting a further block, the further block being within a current picture of the video.

[00137] Clause 16. The method of any of clauses 3-11, wherein performing the conversion comprises: using the motion information for a loop-filtering in the conversion.

15 [00138] Clause 17. The method of clause 16, wherein the loop-filtering comprises a deblocking filtering.

[00139] Clause 18. The method of any of clauses 1-17, further comprising: generating an inter predicted sample of an inter-coded region of the target video block based on a predetermined rule.

20 [00140] Clause 19. The method of clause 18, wherein the predetermined rule comprises at least one of: a rule indicating to uni-directionally predict the inter-coded region, or a rule indicating to bi-directionally predict the inter-coded region.

25 [00141] Clause 20. The method of clause 18 or clause 19, wherein generating the inter predicted sample comprises: if at least one subblock of the target video block comprises at least one of an inter predicted sample or an intra predicted sample, the at least one subblock comprising the target subblock, generating the inter predicted sample based on the predetermined rule.

[00142] Clause 21. The method of any of clauses 1-2, wherein obtaining the prediction information of the target subblock based on the types of the predicted samples comprises: if the target subblock comprises an intra predicted sample and an inter predicted sample, obtaining intra mode information of the intra predicted sample for the target subblock.

[00143] Clause 22. The method of clause 21, wherein obtaining the intra mode information for the target subblock comprises: obtaining intra mode information of an intra-coded region of the target subblock, the intra-coded region being intra-coded.

5 [00144] Clause 23. The method of clause 21 or clause 22, wherein the intra mode information is unavailable.

[00145] Clause 24. The method of any of clauses 21-23, further comprising: applying an adaptive or selective intra mode information storage to the target subblock.

10 [00146] Clause 25. The method of any of clauses 21-24, further comprising: determining, based on coded information for the target video block, whether the intra mode information for the target subblock is unavailable or equal to intra mode information of the intra predicted sample; and storing the intra mode information for the target subblock based on the determination.

[00147] Clause 26. The method of clause 25, wherein the coded information for the target video block comprises at least one of: splitting information, a weight index, a location of a GPM block or GPM subblock, or a dimension of the GPM block or GPM subblock.

15 [00148] Clause 27. The method of clause 26, wherein the splitting information comprises at least one of: a GPM partition mode, a GPM partition angle, or a GPM partition direction.

[00149] Clause 28. The method of any of clauses 21-27, wherein performing the conversion comprises: using the intra mode information for a succeeding process in the conversion.

20 [00150] Clause 29. The method of clause 28, wherein the succeeding process comprises a deblocking process.

[00151] Clause 30. The method of any of clauses 21-27, wherein the obtained intra mode information is used as temporal intra mode information for coding or predicting a further block, the further block being within succeeding coded pictures of the video in a coding order.

25 [00152] Clause 31. The method of clause 30, wherein coding or predicting the further block comprises: applying a Template-based intra mode derivation (TIMD).

[00153] Clause 32. The method of any of clauses 21-27, wherein the obtained intra mode information is used as spatial intra mode information for coding or predicting a further block, the further block being within a current picture of the video.

[00154] Clause 33. The method of clause 1 or clause 2, wherein obtaining the prediction information of the target subblock based on the types of the predicted samples comprises: if the predicted samples comprise at least two inter predicted samples, obtaining motion information for the target subblock.

5 **[00155]** Clause 34. The method of clause 33, wherein obtaining motion information comprises: obtaining motion information in an inter-coded region, an inter predicted sample being coded in the inter-coded region.

[00156] Clause 35. The method of clause 34, wherein the inter-coded region is one of two inter-coded regions for the target subblock.

10 **[00157]** Clause 36. The method of any of clauses 33-35, wherein the motion information comprises at least one of: a uni-directional prediction, or a bi-directional prediction.

[00158] Clause 37. The method of clause 36, wherein in case at least one inter-coded region of the target subblock being bi-directional predicted, the motion information comprises a bi-directional prediction.

15 **[00159]** Clause 38. The method of clause 33, wherein obtaining the motion information comprises: obtaining two types of motion information for the target subblock, each of the two types of motion information associated with a respective inter-coded region of the target subblock.

[00160] Clause 39. The method of clause 38, wherein a third type of motion information is absent from the motion information for the target subblock.

20 **[00161]** Clause 40. The method of clause 39, wherein the third type of motion information comprises combining or constructing motion information from first motion information in a first inter-coded region and second motion information in a second inter-coded region.

[00162] Clause 41. The method of any of clauses 33-40, further comprising: determining, based on coded information for the target subblock, whether to store first motion information of a first
25 inter-coded region of the target subblock or second motion information of a second inter-coded region of the target subblock; and storing respective motion information based on the determination.

[00163] Clause 42. The method of clause 41, wherein combined motion information of the first and second motion information is absent from the stored motion information.

[00164] Clause 43. The method of clause 41 or clause 42, wherein the coded information for the target video block comprises at least one of: splitting information, a weight index, a location of a GPM block or GPM subblock, or a dimension of the GPM block or GPM subblock.

[00165] Clause 44. The method of clause 43, wherein the splitting information comprises at least one of: a GPM partition mode, a GPM partition angle, or a GPM partition direction.

[00166] Clause 45. The method of clause 1 or clause 2, wherein obtaining the prediction information of the target subblock based on the types of the predicted samples comprises: if the predicted samples comprise at least two intra predicted samples, obtaining intra mode information for the target subblock.

[00167] Clause 46. The method of clause 45, wherein obtaining the intra mode information comprises: obtaining the intra mode information in an intra-coded region, an intra predicted sample being coded in the intra-coded region.

[00168] Clause 47. The method of clause 46, wherein the intra-coded region is one of two intra-coded regions for the target subblock.

[00169] Clause 48. The method of any of clauses 45-47, further comprising: determining, based on coded information for the target subblock, whether to store first intra mode information of a first intra-coded region of the target subblock or second intra mode information of a second intra-coded region of the target subblock; and storing respective intra mode information based on the determination.

[00170] Clause 49. The method of clause 48, wherein combined intra mode information of the first and second intra mode information is absent from the stored intra mode information.

[00171] Clause 50. The method of clause 48 or clause 49, wherein the coded information for the target video block comprises at least one of: splitting information, a weight index, a location of a GPM block or GPM subblock, or a dimension of the GPM block or GPM subblock.

[00172] Clause 51. The method of clause 50, wherein the splitting information comprises at least one of: a GPM partition mode, a GPM partition angle, or a GPM partition direction.

[00173] Clause 52. The method of clause 45, wherein obtaining the intra mode information comprises: obtaining at least one of the following for the target subblock: a constructed intra mode, a converted intra mode, or a mapped intra mode.

[00174] Clause 53. The method of clause 45, wherein obtaining the intra mode information comprises: obtaining intra mode information of more than one intra-coded region of the target subblock.

[00175] Clause 54. The method of clause 53, wherein the more than one intra-coded region comprises a first intra-coded region and a second intra-coded region.

[00176] Clause 55. The method of any of clauses 1-54, wherein the target video block comprises one of: a geometric partition mode (GPM) coded block without motion refinement, a geometric partition mode (GPM) coded block with motion refinement, a GPM block with motion vector difference (GPM MMVD), or a GPM block with template matching based motion refinement (GPM TM).

[00177] Clause 56. The method of any of clauses 1-54, wherein the geometric partitioning tool comprises at least one of: a geometric merge mode (GEO), a geometric partition mode (GPM), a wedge prediction mode, a triangular prediction mode, a GPM with motion vector difference (GPM MMVD), a GPM block with template matching based motion refinement (GPM TM), a GPM with inter and intra, or a variant coding tool based on GPM.

[00178] Clause 57. The method of any of clauses 1-56, wherein the conversion includes encoding the target video block into the bitstream.

[00179] Clause 58. The method of any of clauses 1-56, wherein the conversion includes decoding the target video block from the bitstream.

[00180] Clause 59. An apparatus for processing video data comprising a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to perform a method in accordance with any of clauses 1-58.

[00181] Clause 60. A non-transitory computer-readable storage medium storing instructions that cause a processor to perform a method in accordance with any of clauses 1-58.

[00182] Clause 61. A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises: obtaining prediction information of a target subblock in a target video block of the video based on types of predicted samples in the target subblock, the target video block being coded by a geometric partitioning tool; and generating the bitstream based on the prediction information of the target subblock.

[00183] Clause 62. A method for storing a bitstream of a video, comprising: obtaining prediction information of a target subblock in a target video block of the video based on types of predicted samples in the target subblock, the target video block being coded by a geometric partitioning tool; generating the bitstream based on the prediction information of the target subblock; and storing the bitstream in a non-transitory computer-readable recording medium.

Example Device

[00184] Fig. 28 illustrates a block diagram of a computing device 2800 in which various embodiments of the present disclosure can be implemented. The computing device 2800 may be implemented as or included in the source device 110 (or the video encoder 114 or 200) or the destination device 120 (or the video decoder 124 or 300).

[00185] It would be appreciated that the computing device 2800 shown in Fig. 28 is merely for purpose of illustration, without suggesting any limitation to the functions and scopes of the embodiments of the present disclosure in any manner.

[00186] As shown in Fig. 28, the computing device 2800 includes a general-purpose computing device 2800. The computing device 2800 may at least comprise one or more processors or processing units 2810, a memory 2820, a storage unit 2830, one or more communication units 2840, one or more input devices 2850, and one or more output devices 2860.

[00187] In some embodiments, the computing device 2800 may be implemented as any user terminal or server terminal having the computing capability. The server terminal may be a server, a large-scale computing device or the like that is provided by a service provider. The user terminal may for example be any type of mobile terminal, fixed terminal, or portable terminal, including a mobile phone, station, unit, device, multimedia computer, multimedia tablet, Internet node, communicator, desktop computer, laptop computer, notebook computer, netbook computer, tablet computer, personal communication system (PCS) device, personal navigation device, personal digital assistant (PDA), audio/video player, digital camera/video camera, positioning device, television receiver, radio broadcast receiver, E-book device, gaming device, or any combination thereof, including the accessories and peripherals of these devices, or any combination thereof. It would be contemplated that the computing device 2800 can support any type of interface to a user (such as “wearable” circuitry and the like).

[00188] The processing unit 2810 may be a physical or virtual processor and can implement various processes based on programs stored in the memory 2820. In a multi-processor system,

multiple processing units execute computer executable instructions in parallel so as to improve the parallel processing capability of the computing device 2800. The processing unit 2810 may also be referred to as a central processing unit (CPU), a microprocessor, a controller or a microcontroller.

5 [00189] The computing device 2800 typically includes various computer storage medium. Such medium can be any medium accessible by the computing device 2800, including, but not limited to, volatile and non-volatile medium, or detachable and non-detachable medium. The memory 2820 can be a volatile memory (for example, a register, cache, Random Access Memory (RAM)), a non-volatile memory (such as a Read-Only Memory (ROM), Electrically Erasable
10 Programmable Read-Only Memory (EEPROM), or a flash memory), or any combination thereof. The storage unit 2830 may be any detachable or non-detachable medium and may include a machine-readable medium such as a memory, flash memory drive, magnetic disk or another other media, which can be used for storing information and/or data and can be accessed in the computing device 2800.

15 [00190] The computing device 2800 may further include additional detachable/non-detachable, volatile/non-volatile memory medium. Although not shown in Fig. 28, it is possible to provide a magnetic disk drive for reading from and/or writing into a detachable and non-volatile magnetic disk and an optical disk drive for reading from and/or writing into a detachable non-volatile optical disk. In such cases, each drive may be connected to a bus (not shown) via one or more data
20 medium interfaces.

[00191] The communication unit 2840 communicates with a further computing device via the communication medium. In addition, the functions of the components in the computing device 2800 can be implemented by a single computing cluster or multiple computing machines that can communicate via communication connections. Therefore, the computing device 2800 can operate
25 in a networked environment using a logical connection with one or more other servers, networked personal computers (PCs) or further general network nodes.

[00192] The input device 2850 may be one or more of a variety of input devices, such as a mouse, keyboard, tracking ball, voice-input device, and the like. The output device 2860 may be one or more of a variety of output devices, such as a display, loudspeaker, printer, and the like.
30 By means of the communication unit 2840, the computing device 2800 can further communicate with one or more external devices (not shown) such as the storage devices and display device, with one or more devices enabling the user to interact with the computing device 2800, or any

devices (such as a network card, a modem and the like) enabling the computing device 2800 to communicate with one or more other computing devices, if required. Such communication can be performed via input/output (I/O) interfaces (not shown).

[00193] In some embodiments, instead of being integrated in a single device, some or all components of the computing device 2800 may also be arranged in cloud computing architecture. In the cloud computing architecture, the components may be provided remotely and work together to implement the functionalities described in the present disclosure. In some embodiments, cloud computing provides computing, software, data access and storage service, which will not require end users to be aware of the physical locations or configurations of the systems or hardware providing these services. In various embodiments, the cloud computing provides the services via a wide area network (such as Internet) using suitable protocols. For example, a cloud computing provider provides applications over the wide area network, which can be accessed through a web browser or any other computing components. The software or components of the cloud computing architecture and corresponding data may be stored on a server at a remote position. The computing resources in the cloud computing environment may be merged or distributed at locations in a remote data center. Cloud computing infrastructures may provide the services through a shared data center, though they behave as a single access point for the users. Therefore, the cloud computing architectures may be used to provide the components and functionalities described herein from a service provider at a remote location. Alternatively, they may be provided from a conventional server or installed directly or otherwise on a client device.

[00194] The computing device 2800 may be used to implement video encoding/decoding in embodiments of the present disclosure. The memory 2820 may include one or more video coding modules 2825 having one or more program instructions. These modules are accessible and executable by the processing unit 2810 to perform the functionalities of the various embodiments described herein.

[00195] In the example embodiments of performing video encoding, the input device 2850 may receive video data as an input 2870 to be encoded. The video data may be processed, for example, by the video coding module 2825, to generate an encoded bitstream. The encoded bitstream may be provided via the output device 2860 as an output 2880.

[00196] In the example embodiments of performing video decoding, the input device 2850 may receive an encoded bitstream as the input 2870. The encoded bitstream may be processed, for

example, by the video coding module 2825, to generate decoded video data. The decoded video data may be provided via the output device 2860 as the output 2880.

5 [00197] While this disclosure has been particularly shown and described with references to preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the present application as defined by the appended claims. Such variations are intended to be covered by the scope of this present application. As such, the foregoing description of embodiments of the present application is not intended to be limiting.

I/We Claim:

1. A method for video processing, comprising:

obtaining, during a conversion between a target video block of a video and a bitstream of the video, prediction information of a target subblock in the target video block based on types of
5 predicted samples in the target subblock, the target video block being coded by a geometric partitioning tool; and

performing the conversion based on the prediction information of the target subblock.

2. The method of claim 1, further comprising:

10 if the target subblock comprises an intra predicted sample and an inter predicted sample, using intra coded information of the target subblock in at least one of:

a coding of a subsequent video block, or

an in-loop filtering process.

15 3. The method of claim 1 or claim 2, wherein obtaining the prediction information of the target subblock based on the types of the predicted samples comprises:

if the target subblock comprises an intra predicted sample and an inter predicted sample, obtaining motion information of the target subblock.

20 4. The method of claim 3, wherein the motion information is unavailable.

5. The method of claim 3, wherein the motion information comprises a zero vector with a corresponding reference index, the corresponding reference index indicating no reference picture being for the target subblock.

25 6. The method of claim 3, wherein obtaining the motion information comprises:

obtaining the motion information of the inter predicted sample of the target subblock.

7. The method of claim 3, wherein the target subblock comprises an intra region and an inter region, the predicted sample in the intra region being intra predicted, and the predicted sample in the inter region being inter predicted, and

wherein obtaining the motion information comprises:

5 obtaining the motion information of the inter region.

8. The method of any of claims 3-7, further comprising:

applying an adaptive or selective motion information storage to the target subblock.

10 9. The method of any of claims 3-8, further comprising:

determining, based on coded information for the target video block, whether the motion information for the target subblock is unavailable or equal to motion information of the inter predicted sample; and

storing the motion information for the target subblock based on the determination.

15

10. The method of claim 9, wherein the coded information for the target video block comprises at least one of:

splitting information,

a weight index,

20 a location of a GPM block or GPM subblock, or

a dimension of the GPM block or GPM subblock.

11. The method of claim 10, wherein the splitting information comprises at least one of:

a GPM partition mode,

25 a GPM partition angle, or

a GPM partition direction.

12. The method of any of claims 3-11, wherein performing the conversion comprises:

using the motion information for a succeeding process in the conversion.

30

13. The method of claim 12, wherein the succeeding process comprises a deblocking process.

14. The method of any of claims 3-11, wherein the obtained motion information is used as temporal motion information for coding or predicting a further block, the further block being within succeeding coded pictures of the video in a coding order.

15. The method of any of claims 3-11, wherein the obtained motion information is used as spatial motion information for coding or predicting a further block, the further block being within a current picture of the video.

16. The method of any of claims 3-11, wherein performing the conversion comprises: using the motion information for a loop-filtering in the conversion.

17. The method of claim 16, wherein the loop-filtering comprises a deblocking filtering.

18. The method of any of claims 1-17, further comprising:
generating an inter predicted sample of an inter-coded region of the target video block based on a predetermined rule.

19. The method of claim 18, wherein the predetermined rule comprises at least one of:
a rule indicating to uni-directionally predict the inter-coded region, or
a rule indicating to bi-directionally predict the inter-coded region.

20. The method of claim 18 or claim 19, wherein generating the inter predicted sample comprises:

if at least one subblock of the target video block comprises at least one of an inter predicted sample or an intra predicted sample, the at least one subblock comprising the target subblock, generating the inter predicted sample based on the predetermined rule.

21. The method of any of claims 1-2, wherein obtaining the prediction information of the target subblock based on the types of the predicted samples comprises:

if the target subblock comprises an intra predicted sample and an inter predicted sample, obtaining intra mode information of the intra predicted sample for the target subblock.

22. The method of claim 21, wherein obtaining the intra mode information for the target
5 subblock comprises:

obtaining intra mode information of an intra-coded region of the target subblock, the intra-coded region being intra-coded.

23. The method of claim 21 or claim 22, wherein the intra mode information is unavailable.
10

24. The method of any of claims 21-23, further comprising:
applying an adaptive or selective intra mode information storage to the target subblock.

25. The method of any of claims 21-24, further comprising:
15 determining, based on coded information for the target video block, whether the intra mode information for the target subblock is unavailable or equal to intra mode information of the intra predicted sample; and
storing the intra mode information for the target subblock based on the determination.

20 26. The method of claim 25, wherein the coded information for the target video block comprises at least one of:
splitting information,
a weight index,
a location of a GPM block or GPM subblock, or
25 a dimension of the GPM block or GPM subblock.

27. The method of claim 26, wherein the splitting information comprises at least one of:
a GPM partition mode,
a GPM partition angle, or
30 a GPM partition direction.

28. The method of any of claims 21-27, wherein performing the conversion comprises:
using the intra mode information for a succeeding process in the conversion.

5 29. The method of claim 28, wherein the succeeding process comprises a deblocking process.

30. The method of any of claims 21-27, wherein the obtained intra mode information is used
as temporal intra mode information for coding or predicting a further block, the further block
being within succeeding coded pictures of the video in a coding order.

10

31. The method of claim 30, wherein coding or predicting the further block comprises:
applying a Template-based intra mode derivation (TIMD).

32. The method of any of claims 21-27, wherein the obtained intra mode information is used
15 as spatial intra mode information for coding or predicting a further block, the further block being
within a current picture of the video.

33. The method of claim 1 or claim 2, wherein obtaining the prediction information of the
target subblock based on the types of the predicted samples comprises:

20 if the predicted samples comprise at least two inter predicted samples, obtaining motion
information for the target subblock.

34. The method of claim 33, wherein obtaining motion information comprises:

obtaining motion information in an inter-coded region, an inter predicted sample being
25 coded in the inter-coded region.

35. The method of claim 34, wherein the inter-coded region is one of two inter-coded
regions for the target subblock.

30 36. The method of any of claims 33-35, wherein the motion information comprises at least
one of:

a uni-directional prediction, or
a bi-directional prediction.

37. The method of claim 36, wherein in case at least one inter-coded region of the target subblock being bi-directional predicted, the motion information comprises a bi-directional prediction.

38. The method of claim 33, wherein obtaining the motion information comprises:
obtaining two types of motion information for the target subblock, each of the two types of motion information associated with a respective inter-coded region of the target subblock.

39. The method of claim 38, wherein a third type of motion information is absent from the motion information for the target subblock.

40. The method of claim 39, wherein the third type of motion information comprises combining or constructing motion information from first motion information in a first inter-coded region and second motion information in a second inter-coded region.

41. The method of any of claims 33-40, further comprising:
determining, based on coded information for the target subblock, whether to store first motion information of a first inter-coded region of the target subblock or second motion information of a second inter-coded region of the target subblock; and
storing respective motion information based on the determination.

42. The method of claim 41, wherein combined motion information of the first and second motion information is absent from the stored motion information.

43. The method of claim 41 or claim 42, wherein the coded information for the target video block comprises at least one of:
splitting information,
a weight index,

a location of a GPM block or GPM subblock, or
a dimension of the GPM block or GPM subblock.

44. The method of claim 43, wherein the splitting information comprises at least one of:

- 5 a GPM partition mode,
 a GPM partition angle, or
 a GPM partition direction.

45. The method of claim 1 or claim 2, wherein obtaining the prediction information of the
10 target subblock based on the types of the predicted samples comprises:

 if the predicted samples comprise at least two intra predicted samples, obtaining intra mode
information for the target subblock.

46. The method of claim 45, wherein obtaining the intra mode information comprises:

- 15 obtaining the intra mode information in an intra-coded region, an intra predicted sample
being coded in the intra-coded region.

47. The method of claim 46, wherein the intra-coded region is one of two intra-coded
regions for the target subblock.

20

48. The method of any of claims 45-47, further comprising:

 determining, based on coded information for the target subblock, whether to store first intra
mode information of a first intra-coded region of the target subblock or second intra mode
information of a second intra-coded region of the target subblock; and

- 25 storing respective intra mode information based on the determination.

49. The method of claim 48, wherein combined intra mode information of the first and
second intra mode information is absent from the stored intra mode information.

50. The method of claim 48 or claim 49, wherein the coded information for the target video block comprises at least one of:

splitting information,

a weight index,

5 a location of a GPM block or GPM subblock, or

a dimension of the GPM block or GPM subblock.

51. The method of claim 50, wherein the splitting information comprises at least one of:

a GPM partition mode,

10 a GPM partition angle, or

a GPM partition direction.

52. The method of claim 45, wherein obtaining the intra mode information comprises:

obtaining at least one of the following for the target subblock:

15 a constructed intra mode,

a converted intra mode, or

a mapped intra mode.

53. The method of claim 45, wherein obtaining the intra mode information comprises:

20 obtaining intra mode information of more than one intra-coded region of the target subblock.

54. The method of claim 53, wherein the more than one intra-coded region comprises a first intra-coded region and a second intra-coded region.

25 55. The method of any of claims 1-54, wherein the target video block comprises one of:

a geometric partition mode (GPM) coded block without motion refinement,

a geometric partition mode (GPM) coded block with motion refinement,

a GPM block with motion vector difference (GPM MMVD), or

a GPM block with template matching based motion refinement (GPM TM).

30

56. The method of any of claims 1-54, wherein the geometric partitioning tool comprises at least one of:

a geometric merge mode (GEO),

a geometric partition mode (GPM),

5 a wedge prediction mode,

a triangular prediction mode,

a GPM with motion vector difference (GPM MMVD),

a GPM block with template matching based motion refinement (GPM TM),

a GPM with inter and intra, or

10 a variant coding tool based on GPM.

57. The method of any of claims 1-56, wherein the conversion includes encoding the target video block into the bitstream.

15 58. The method of any of claims 1-56, wherein the conversion includes decoding the target video block from the bitstream.

59. An apparatus for processing video data comprising a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor,
20 cause the processor to perform a method in accordance with any of claims 1-58.

60. A non-transitory computer-readable storage medium storing instructions that cause a processor to perform a method in accordance with any of claims 1-58.

25 61. A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises:

obtaining prediction information of a target subblock in a target video block of the video based on types of predicted samples in the target subblock, the target video block being coded by
30 a geometric partitioning tool; and

generating the bitstream based on the prediction information of the target subblock.

62. A method for storing a bitstream of a video, comprising:

obtaining prediction information of a target subblock in a target video block of the video based on types of predicted samples in the target subblock, the target video block being coded by

5 a geometric partitioning tool;

generating the bitstream based on the prediction information of the target subblock; and

storing the bitstream in a non-transitory computer-readable recording medium.

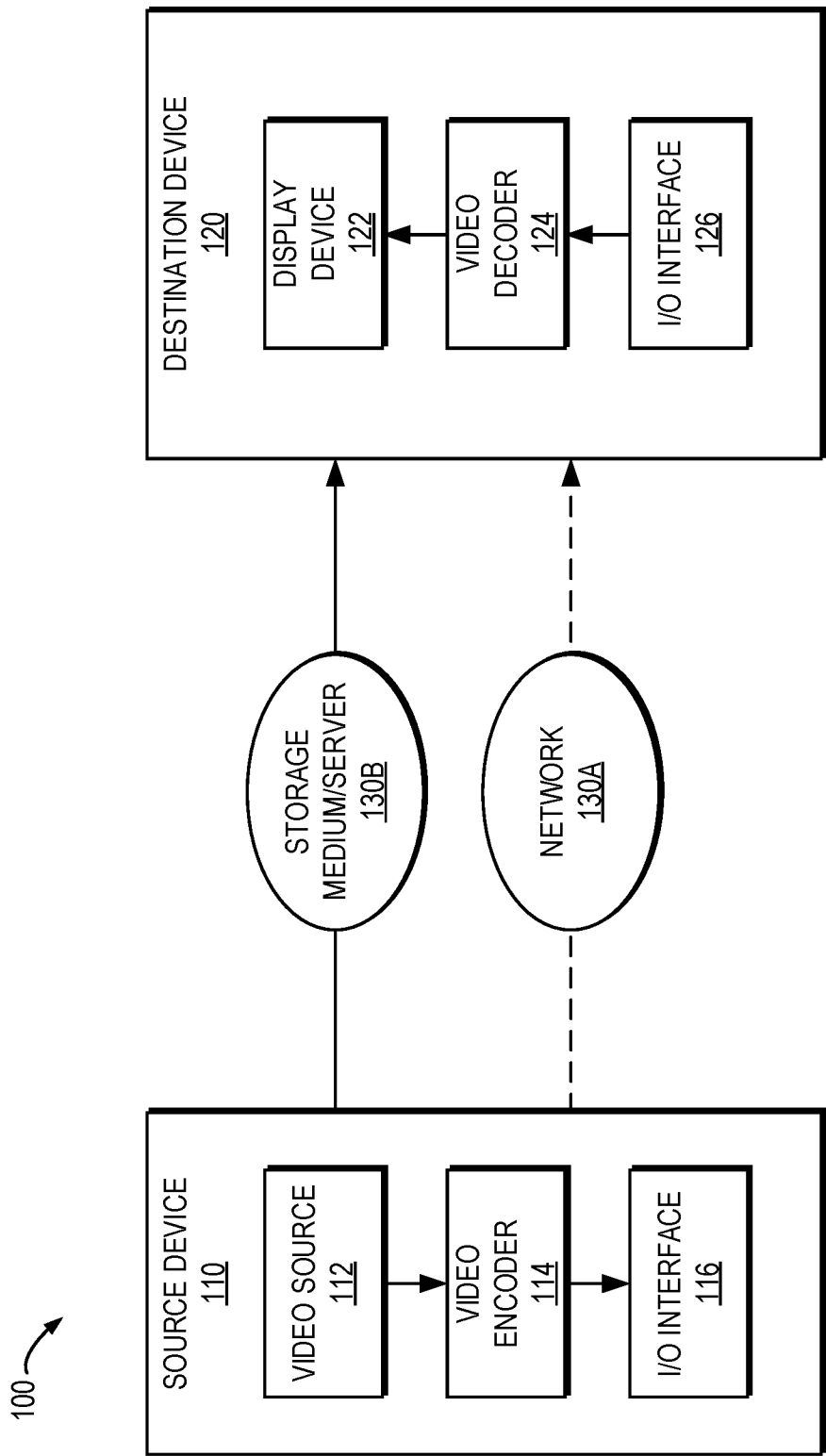


Fig. 1

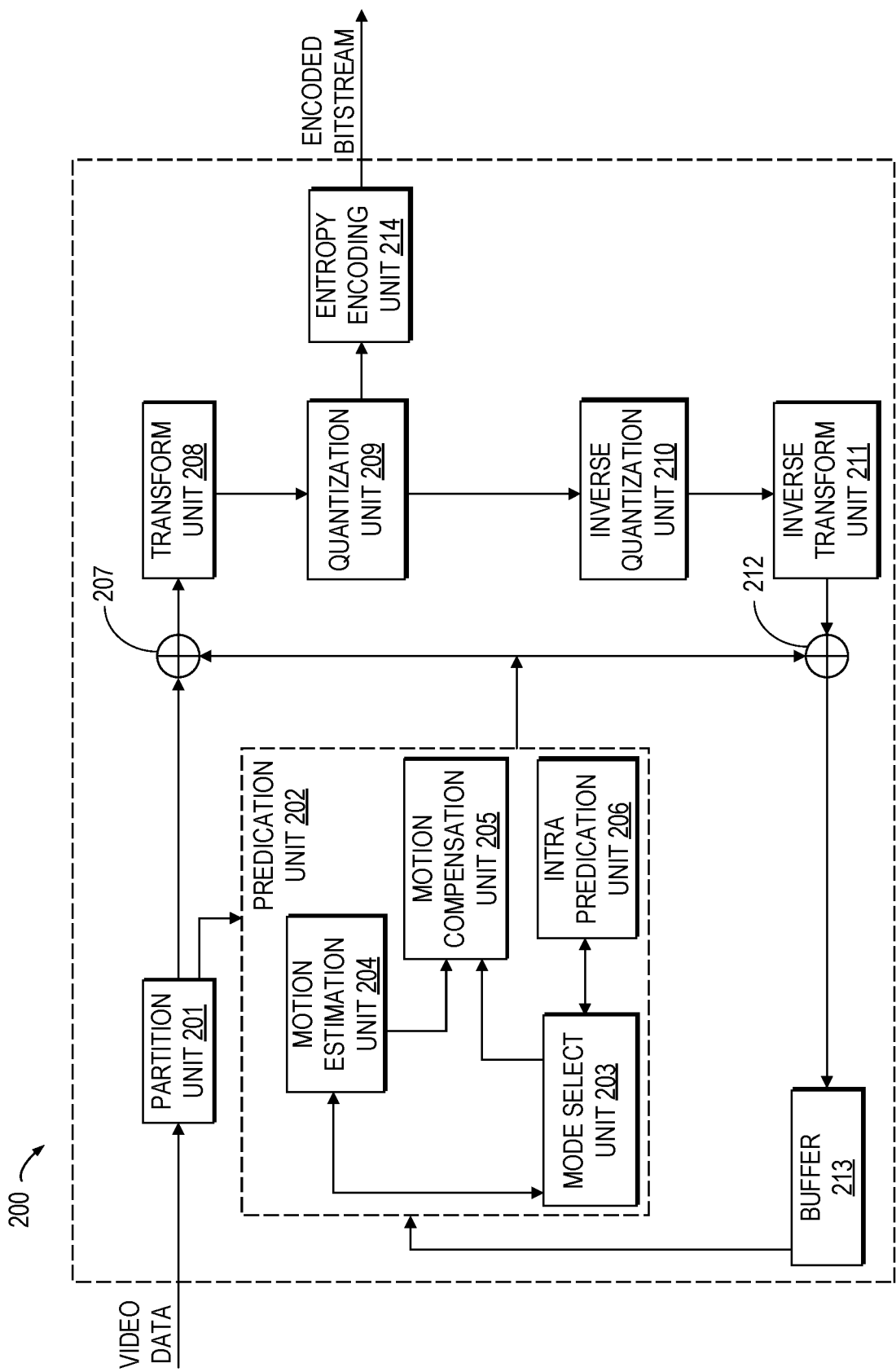


Fig. 2

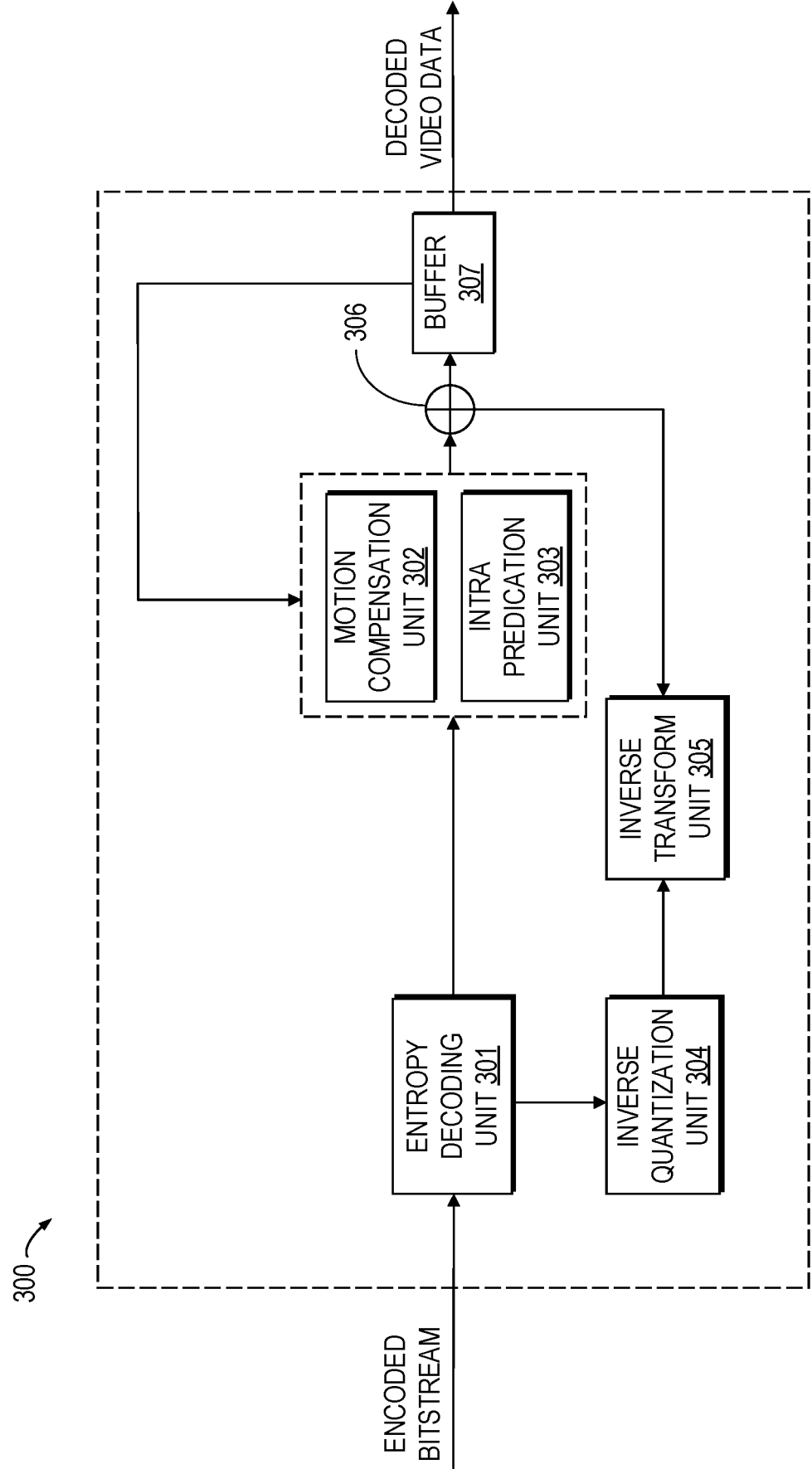


Fig. 3

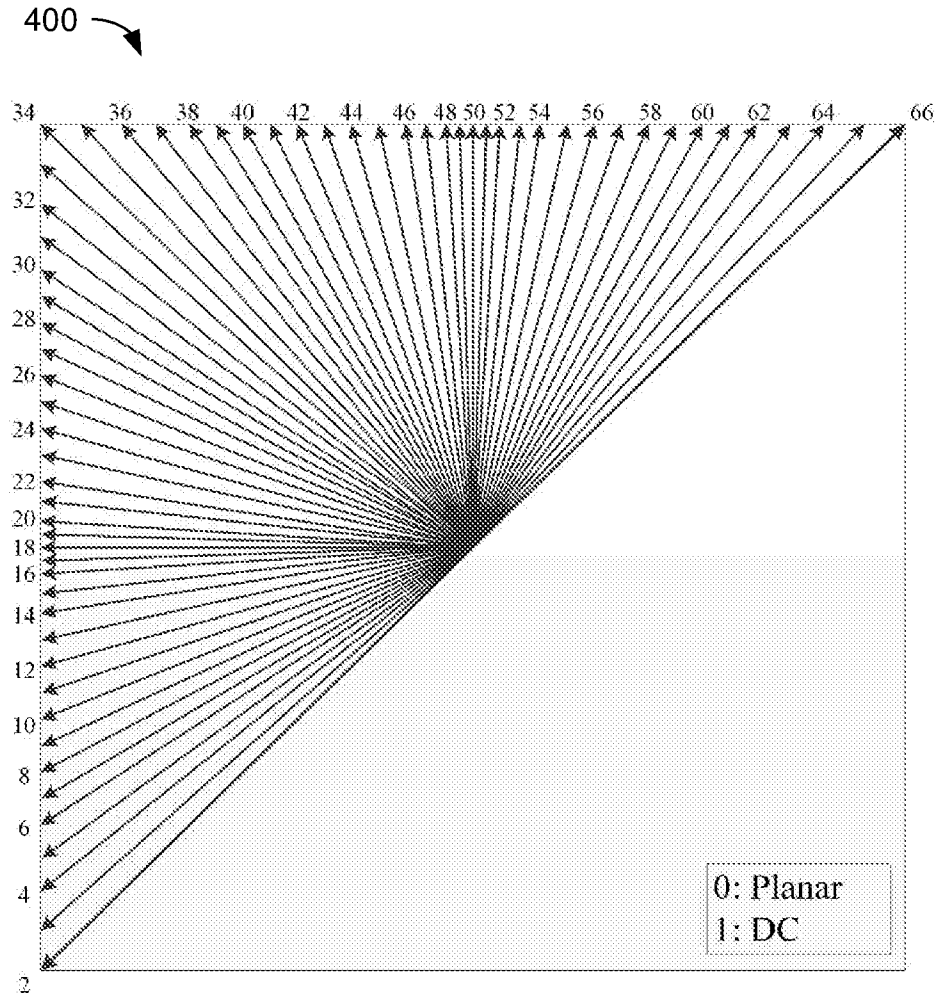


Fig. 4

500

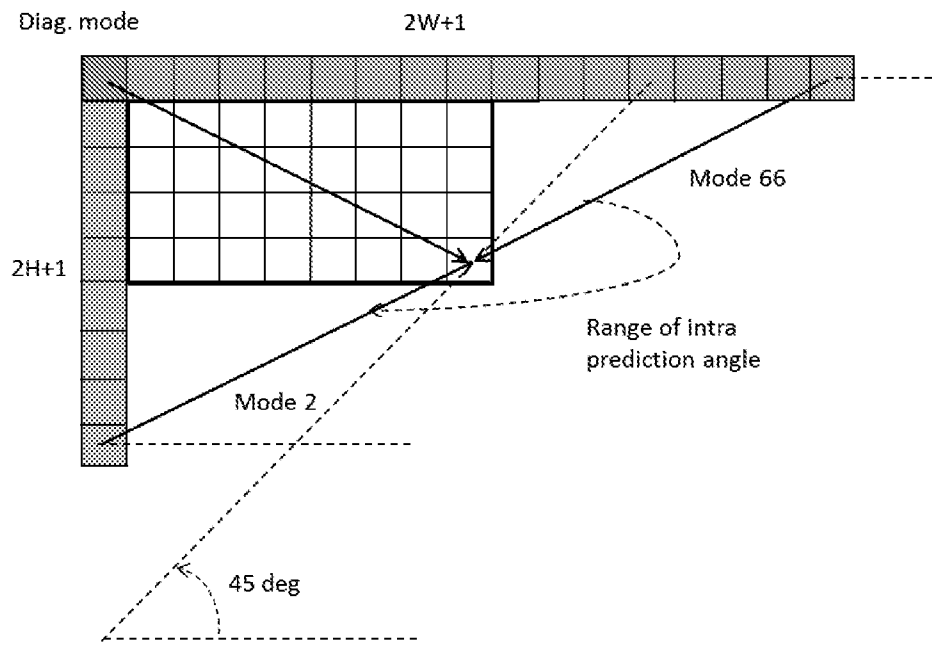


Fig. 5A

550

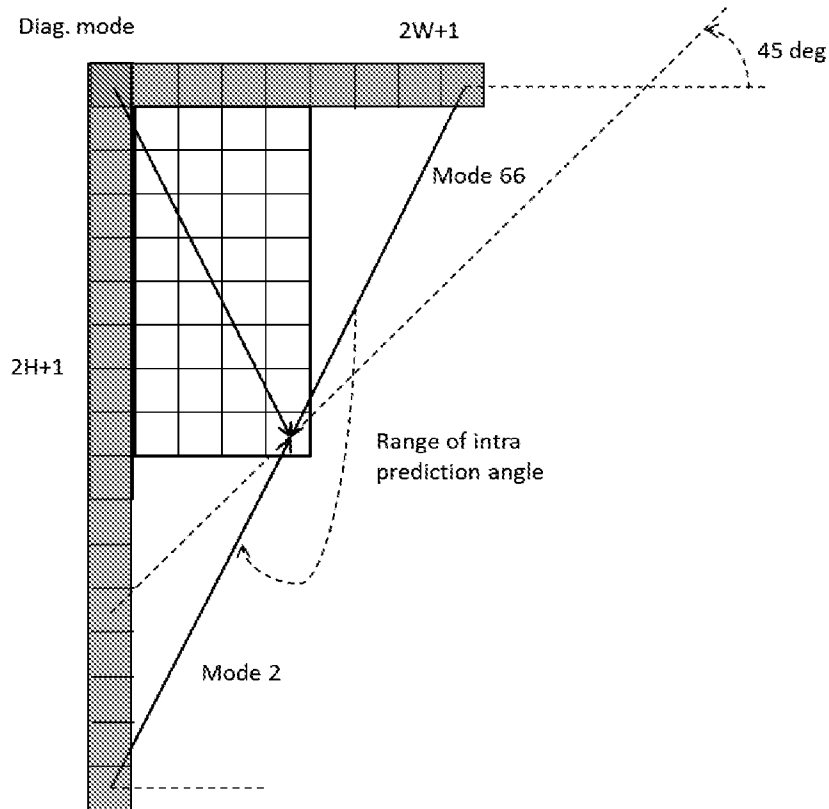


Fig. 5B

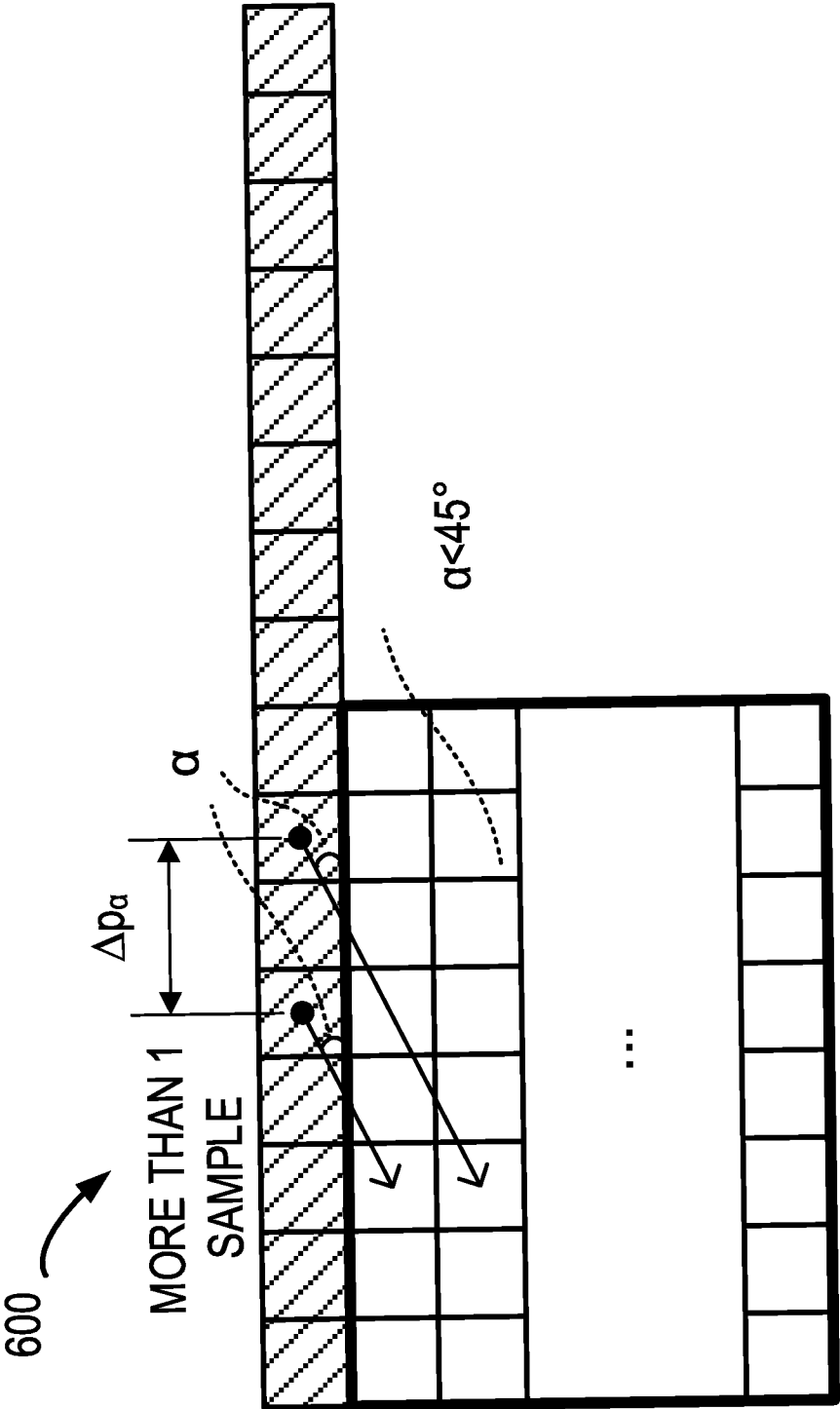


Fig. 6

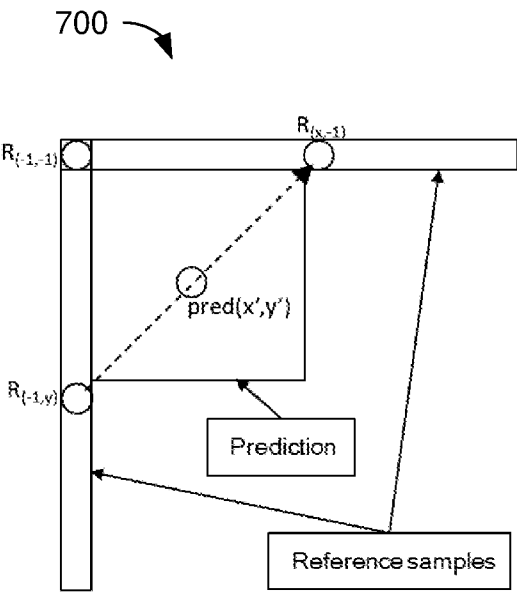


Fig. 7A

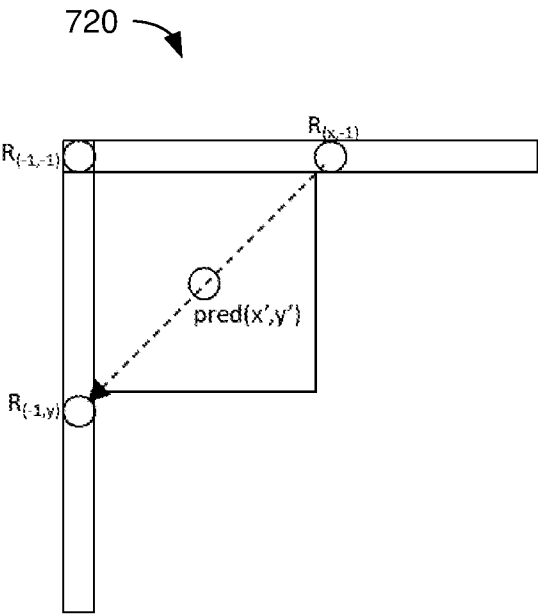


Fig. 7B

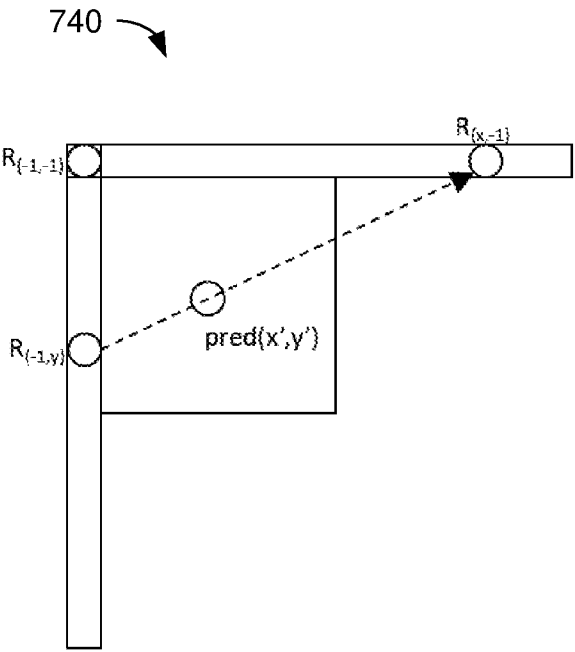


Fig. 7C

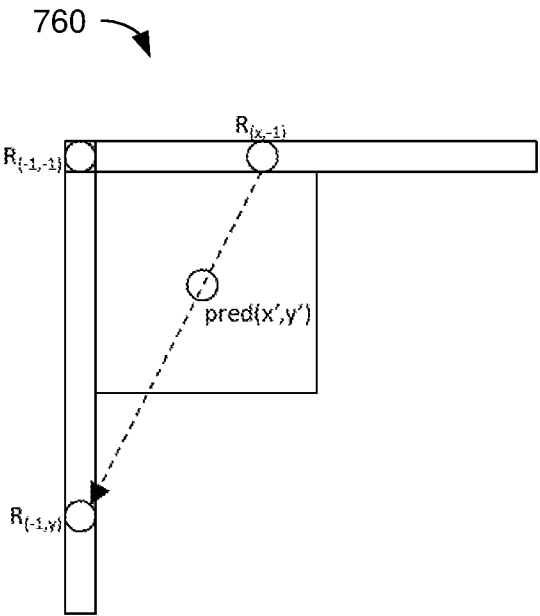


Fig. 7D

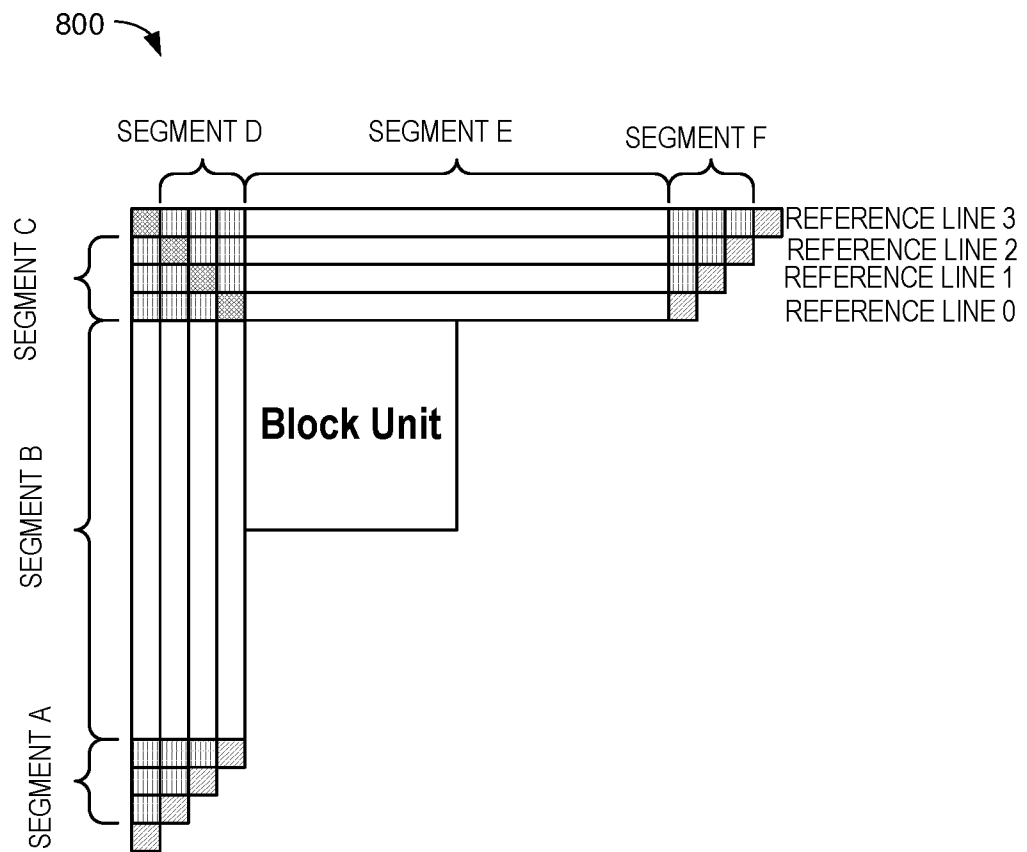


Fig. 8

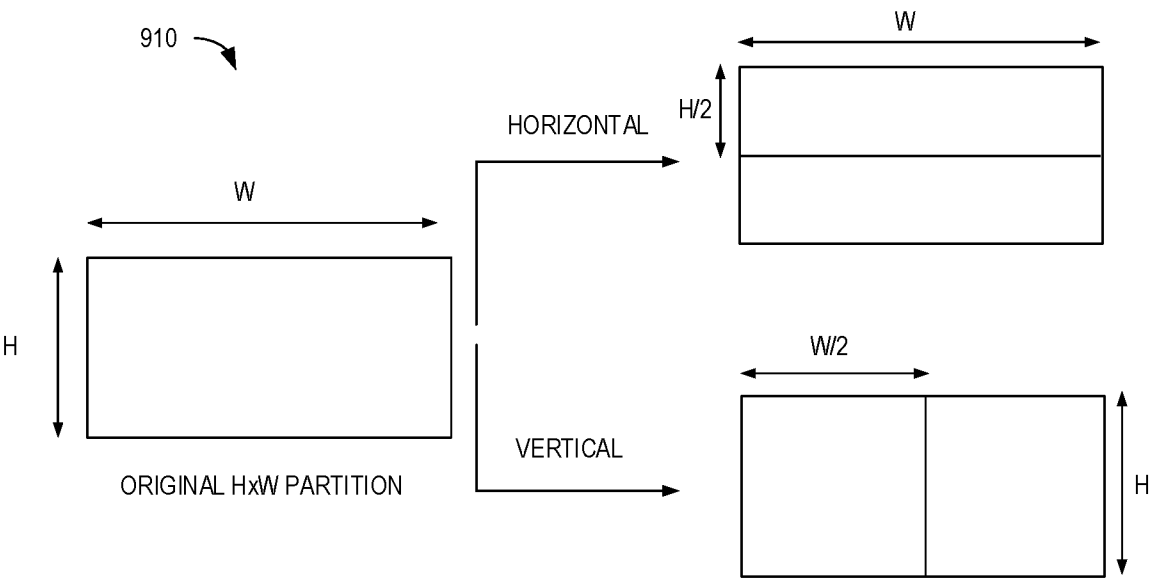


Fig. 9A

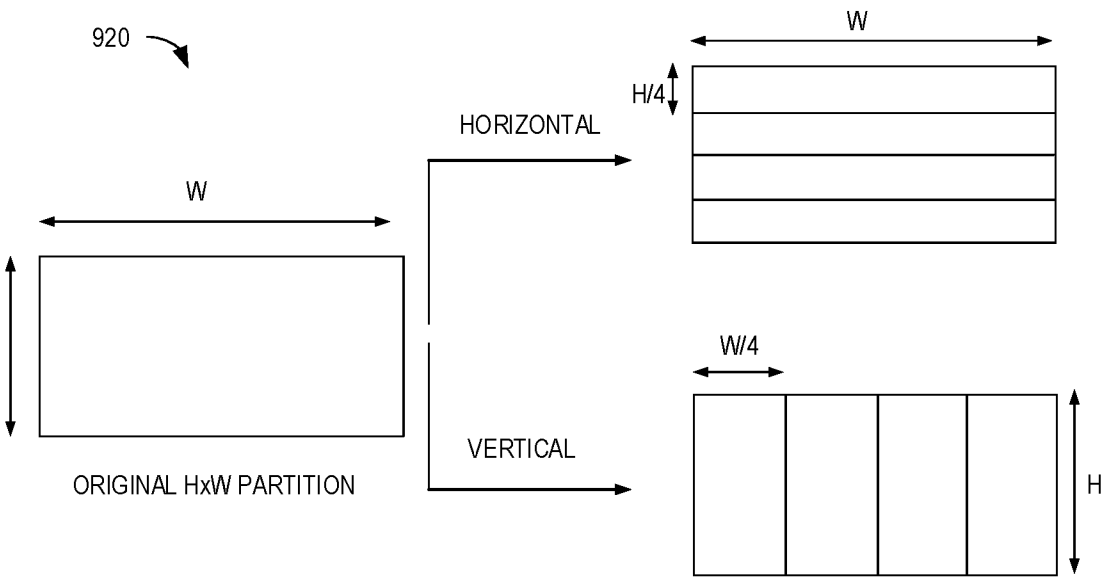


Fig. 9B

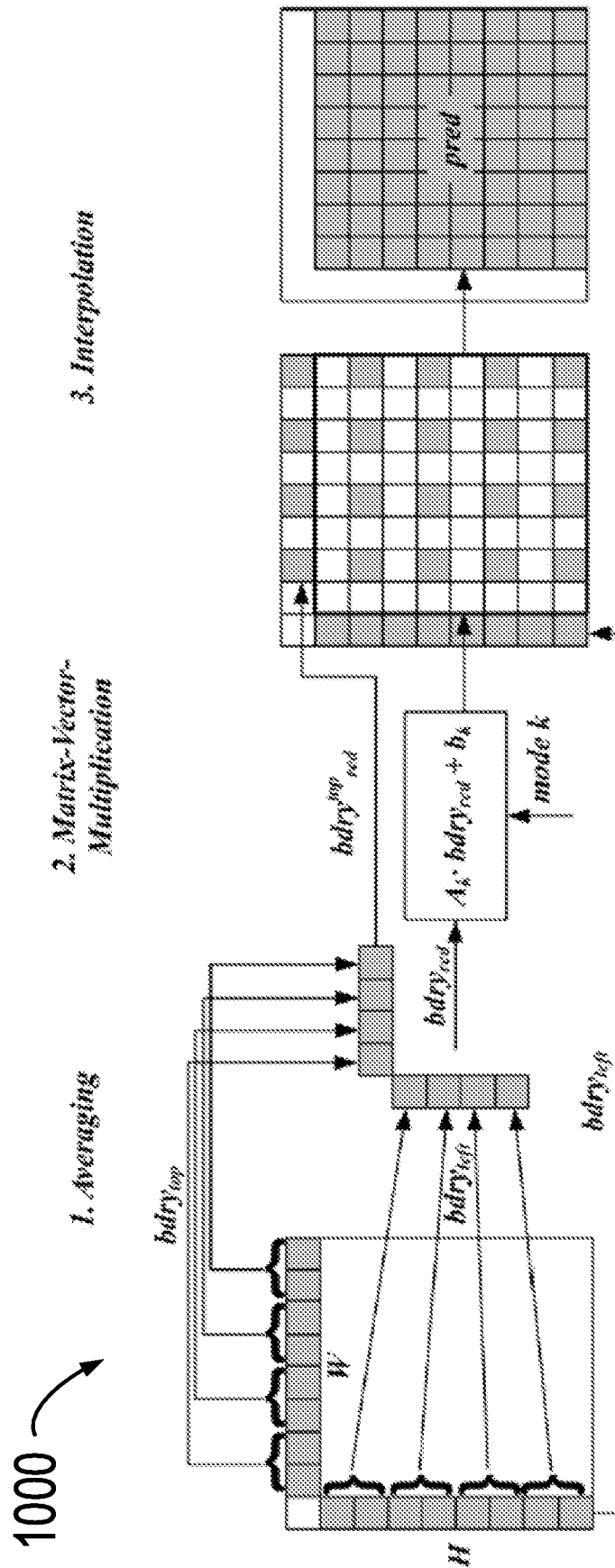


Fig. 10

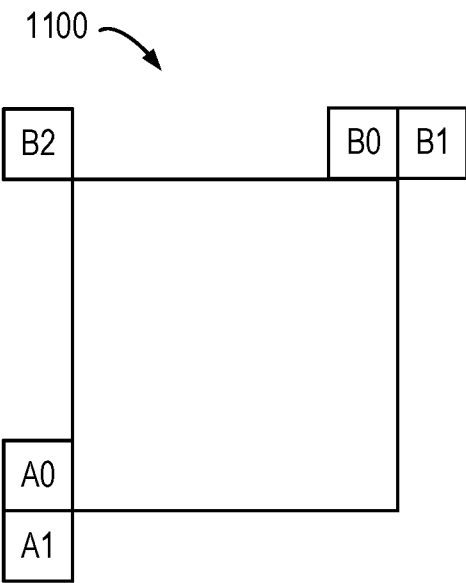


Fig. 11

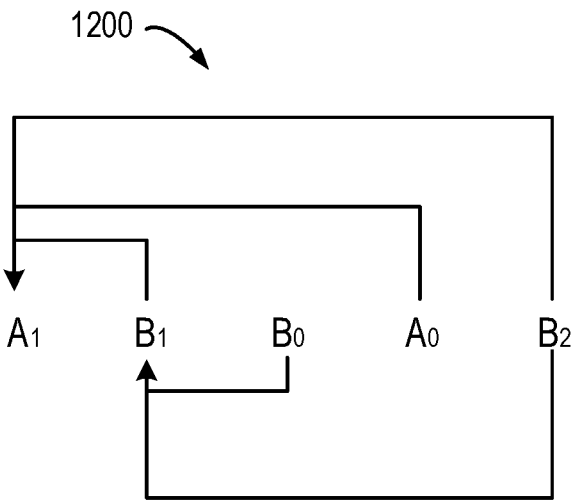


Fig. 12

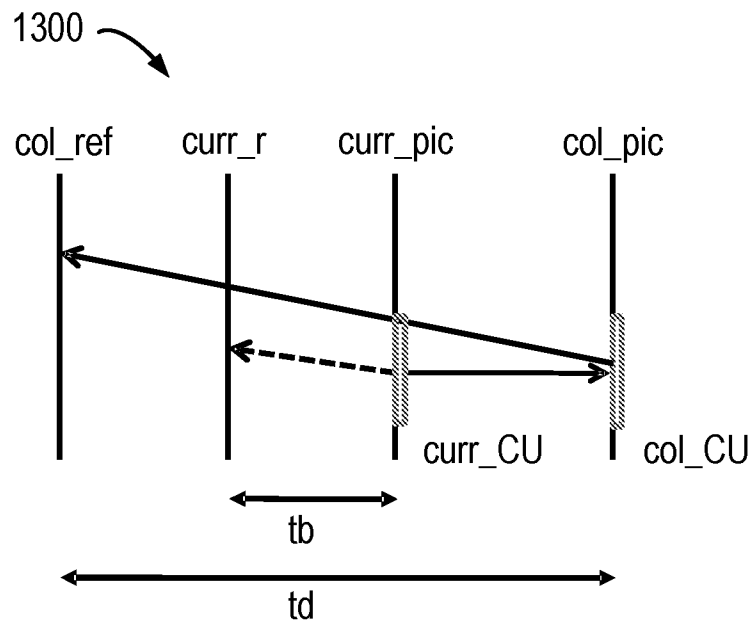


Fig. 13

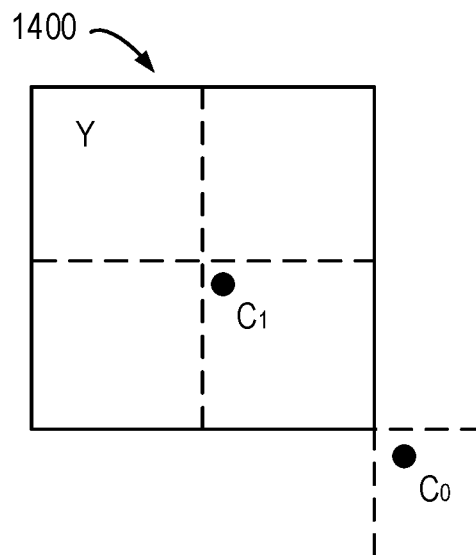


Fig. 14

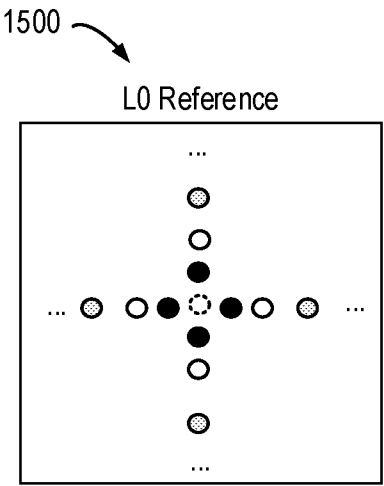


Fig. 15A

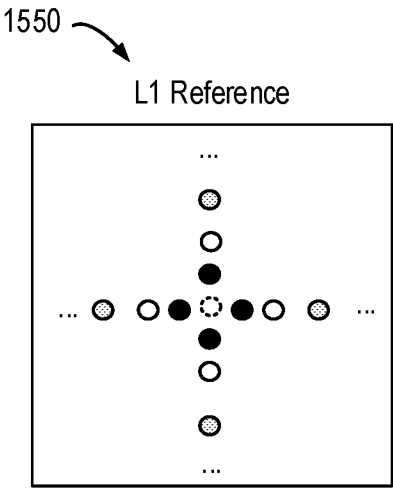


Fig. 15B

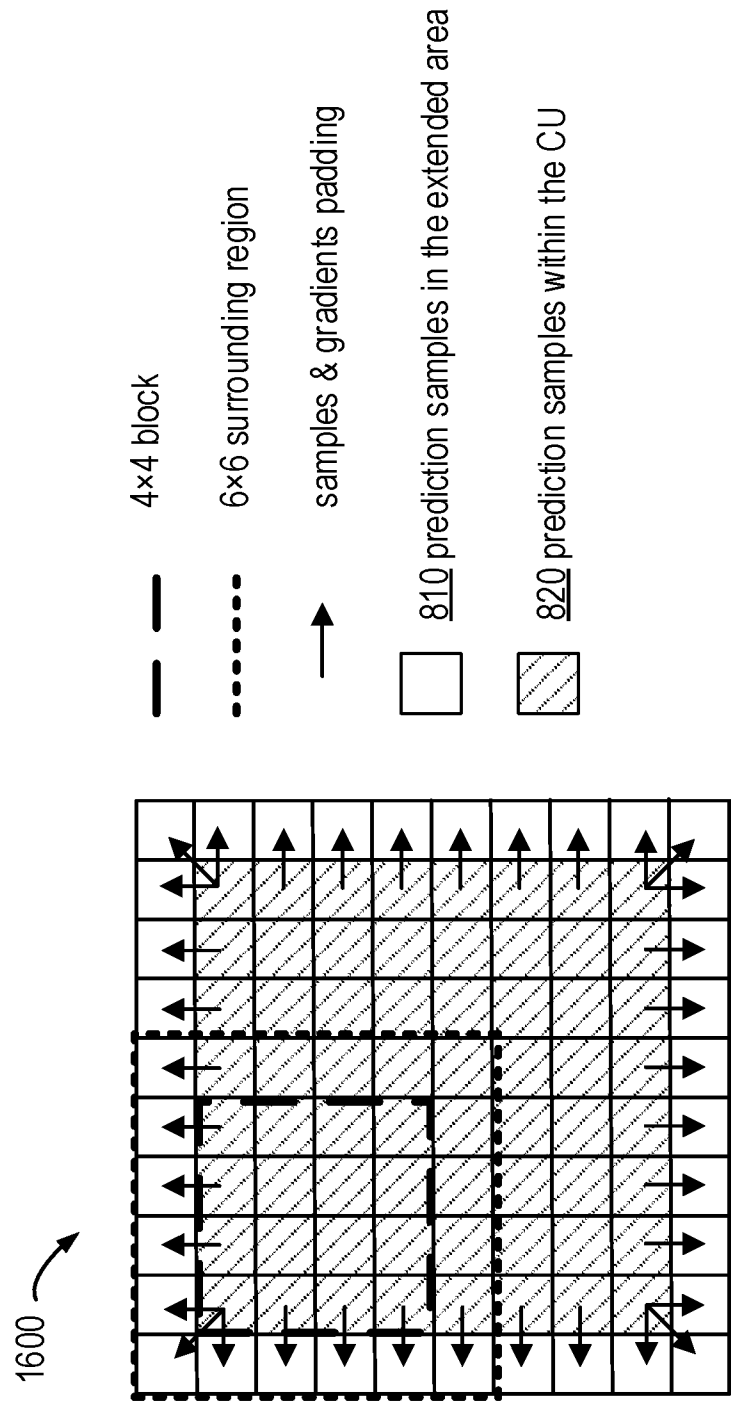


Fig. 16

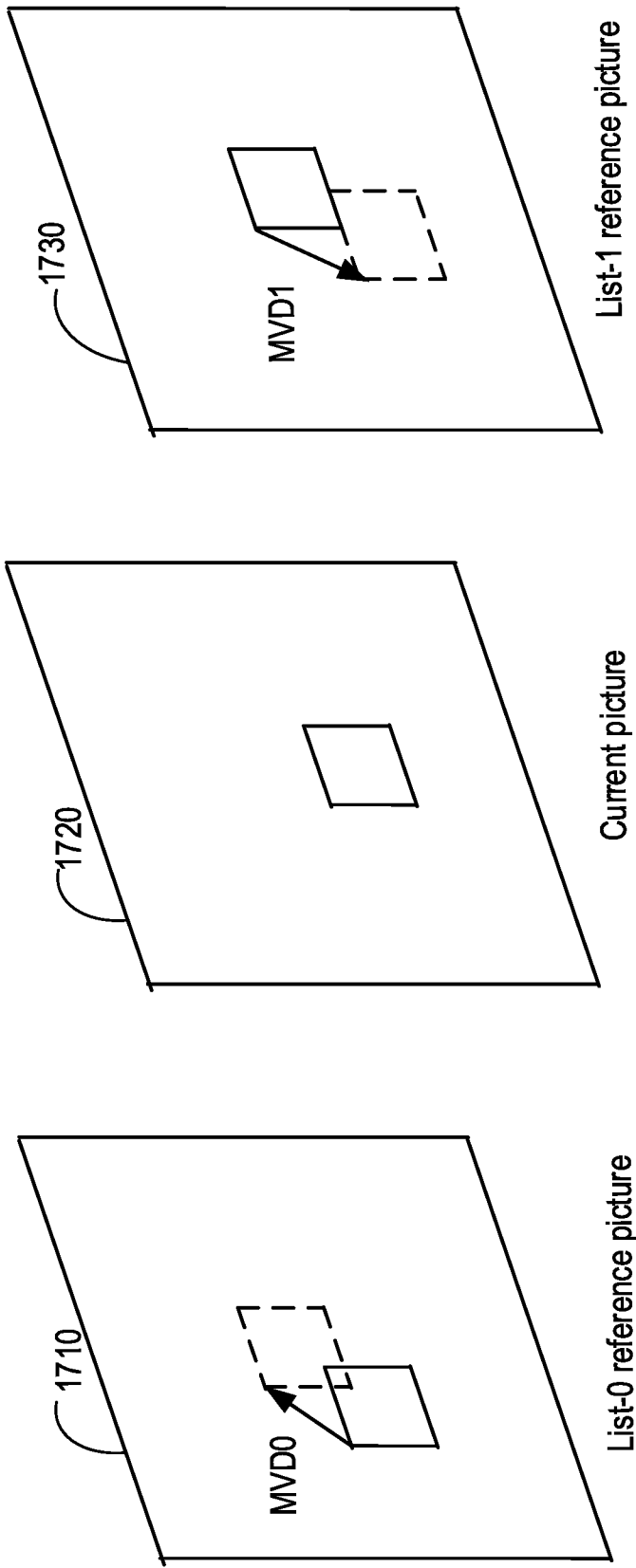


Fig. 17

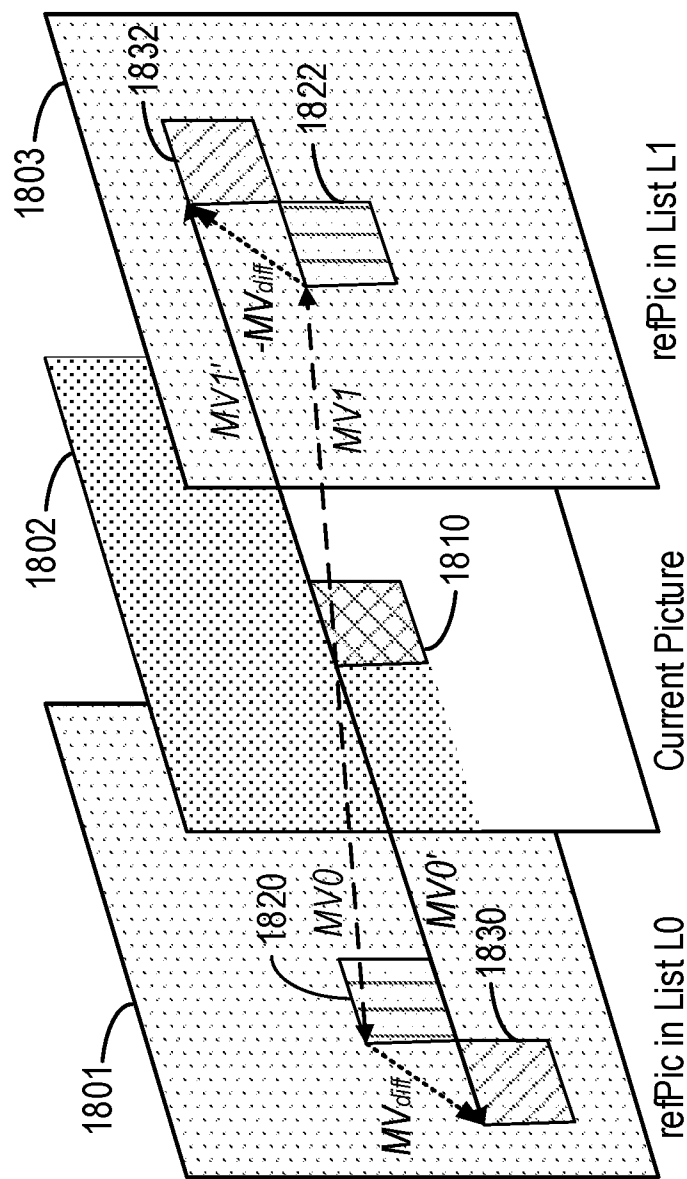


Fig. 18

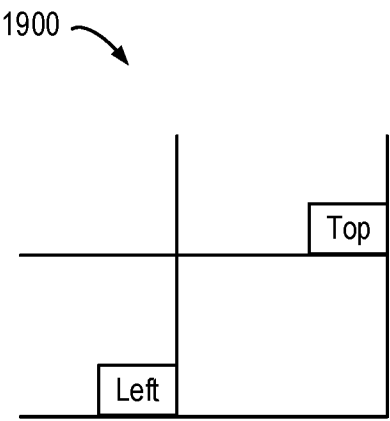


Fig. 19

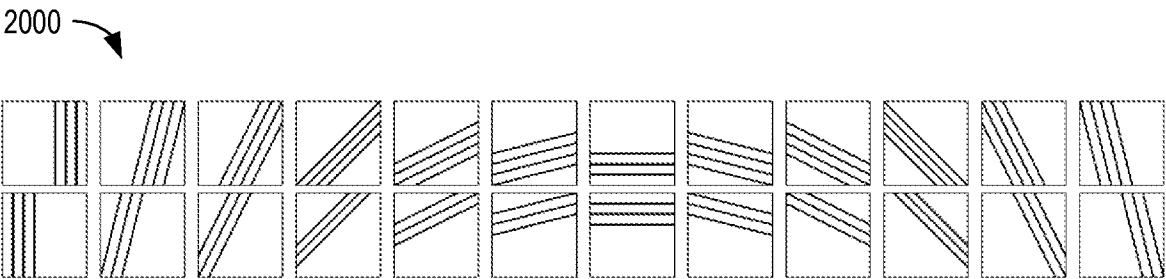


Fig. 20

2110

Merge index	L0 MV	L1 MV
0	x	
1		x
2	x	
3		x
4	x	

Fig. 21

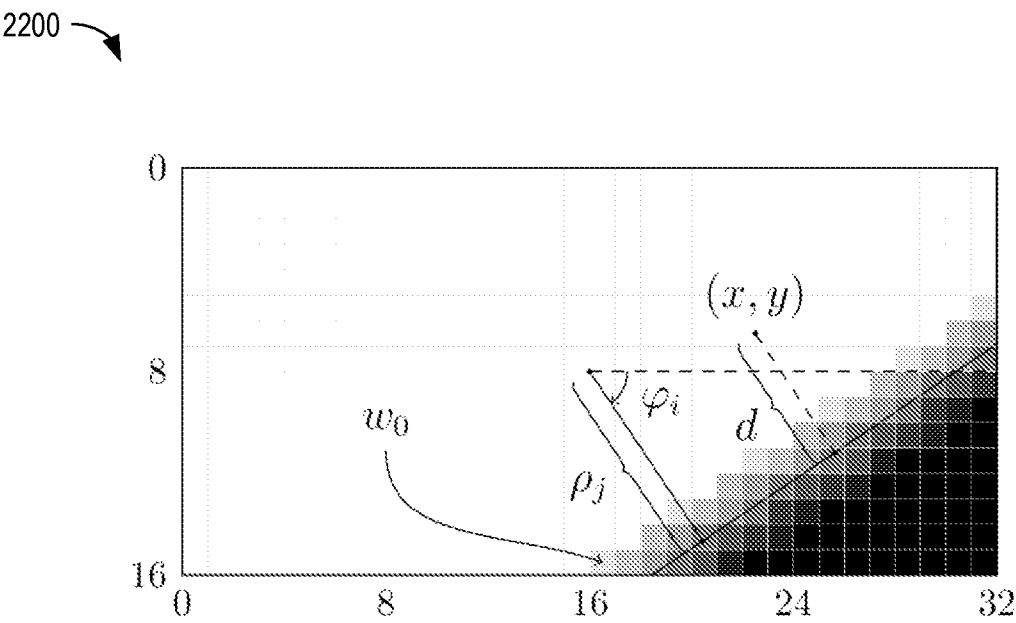


Fig. 22

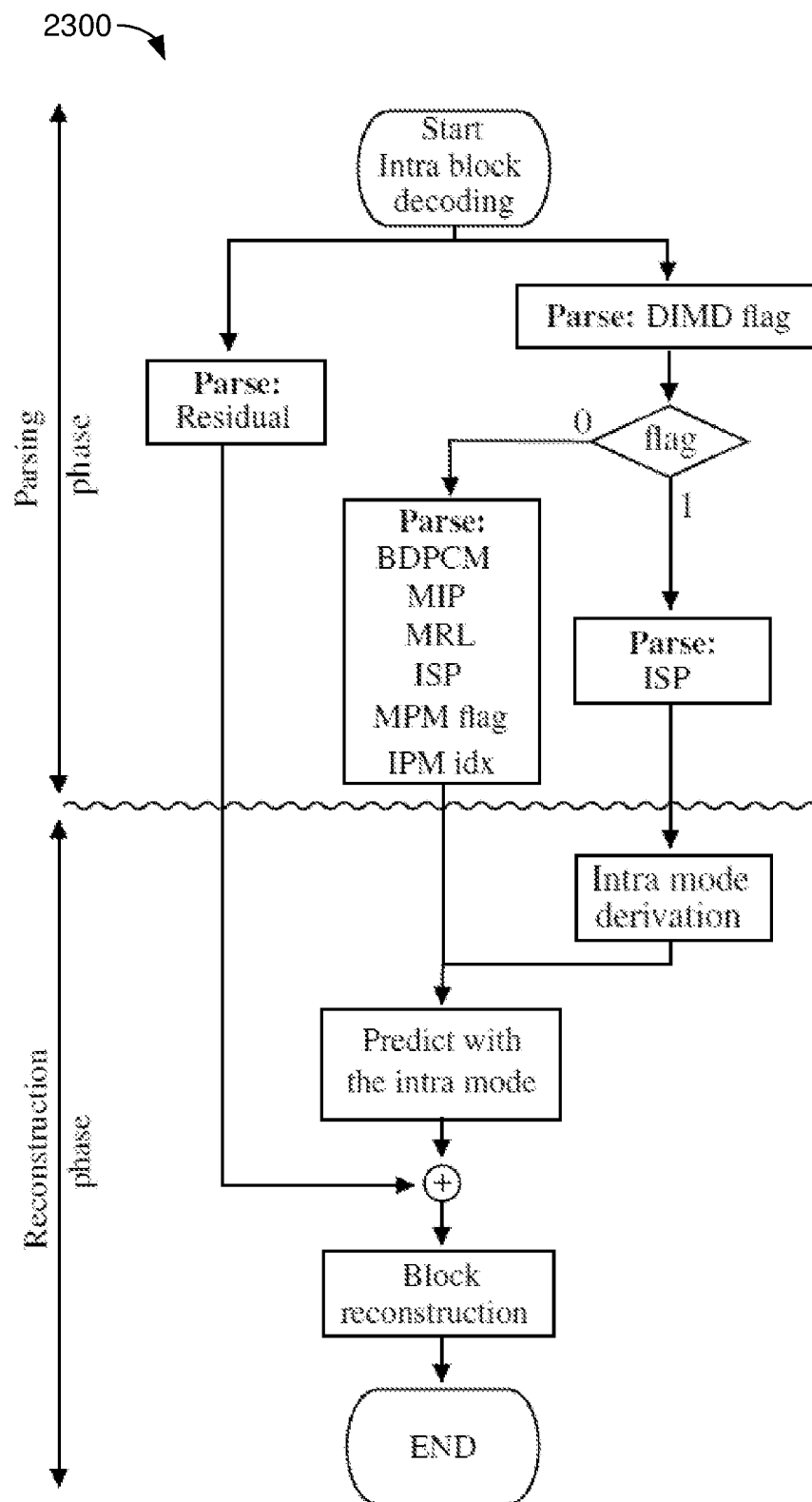


Fig. 23

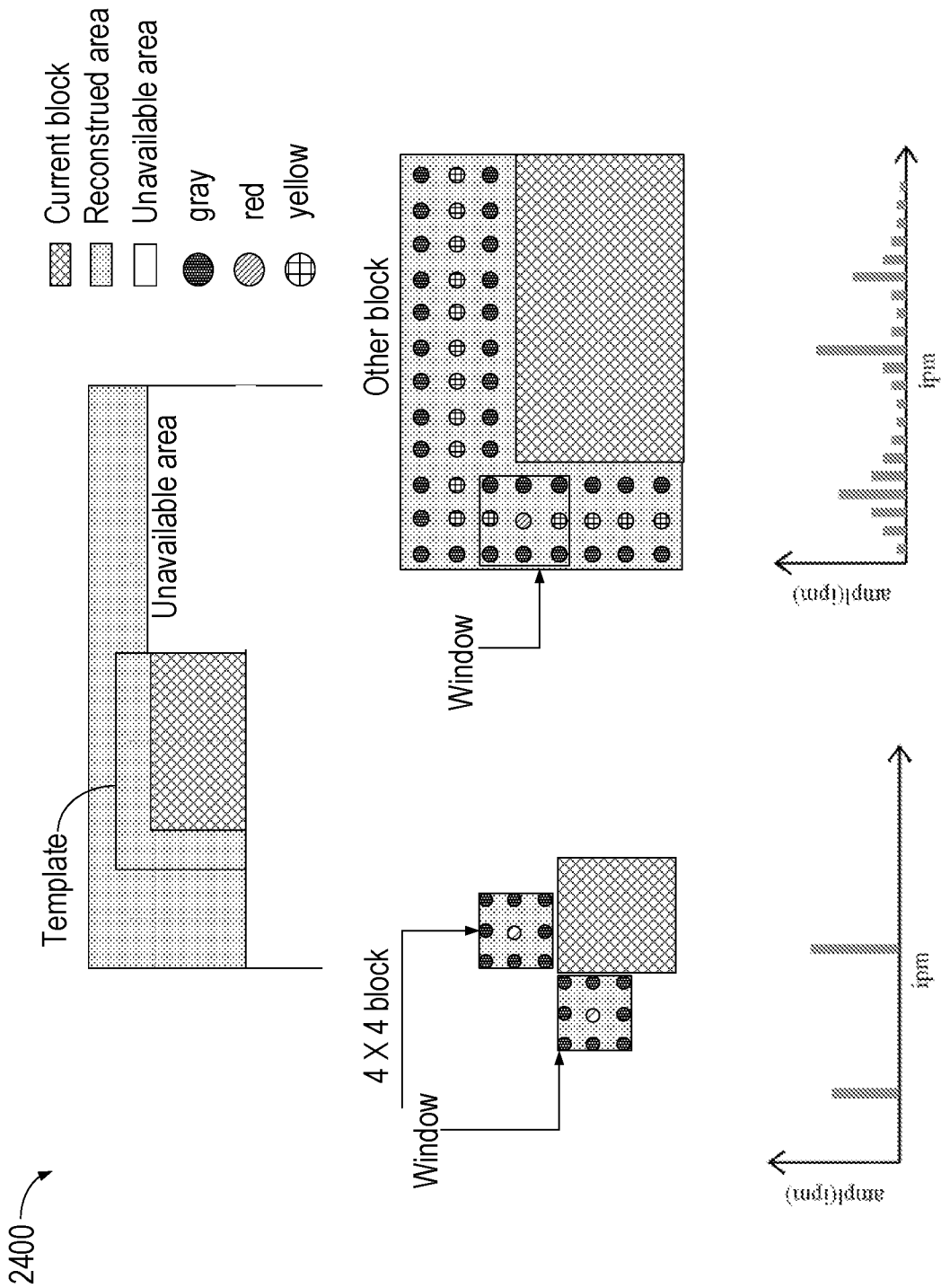


Fig. 24

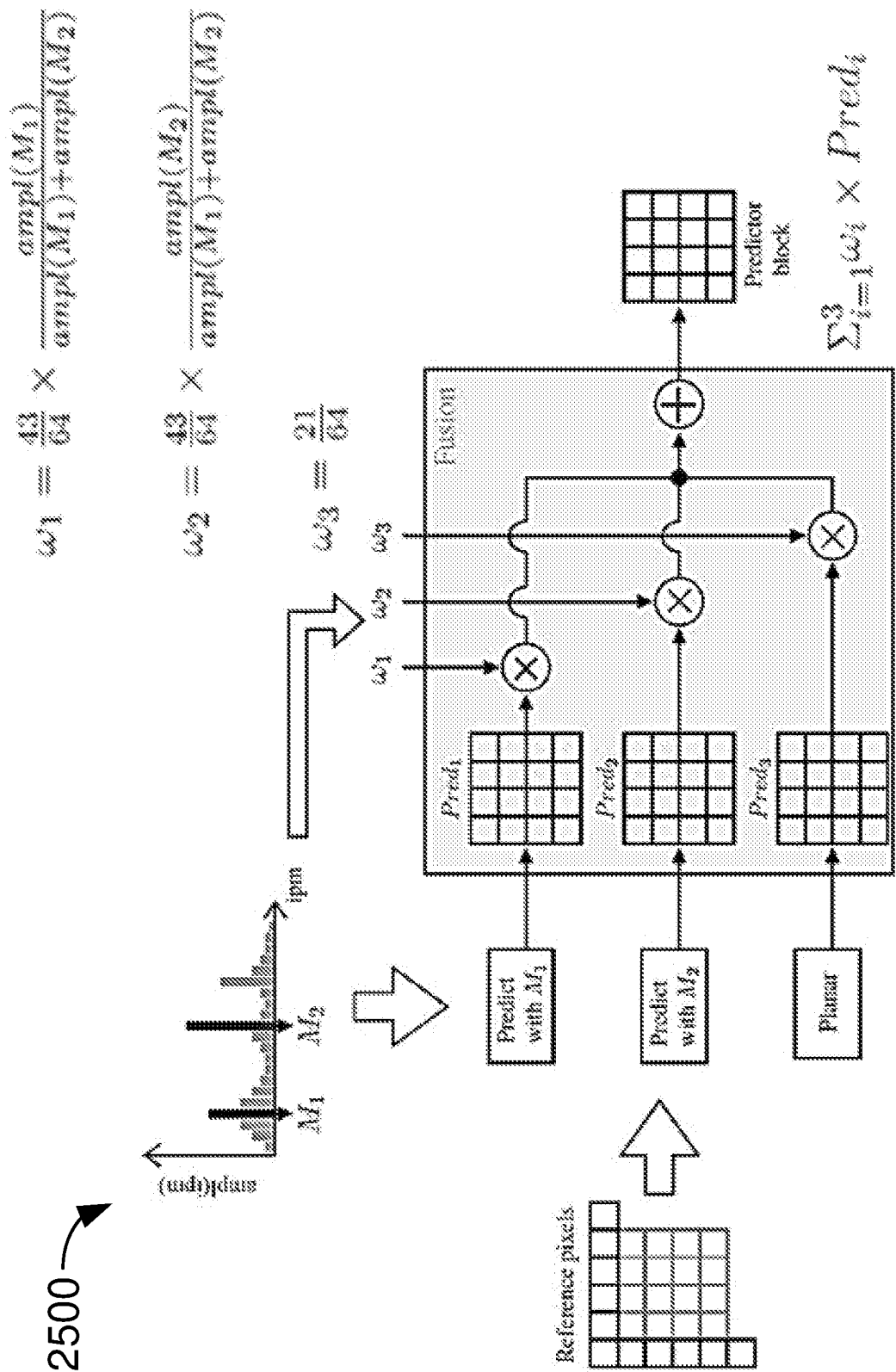
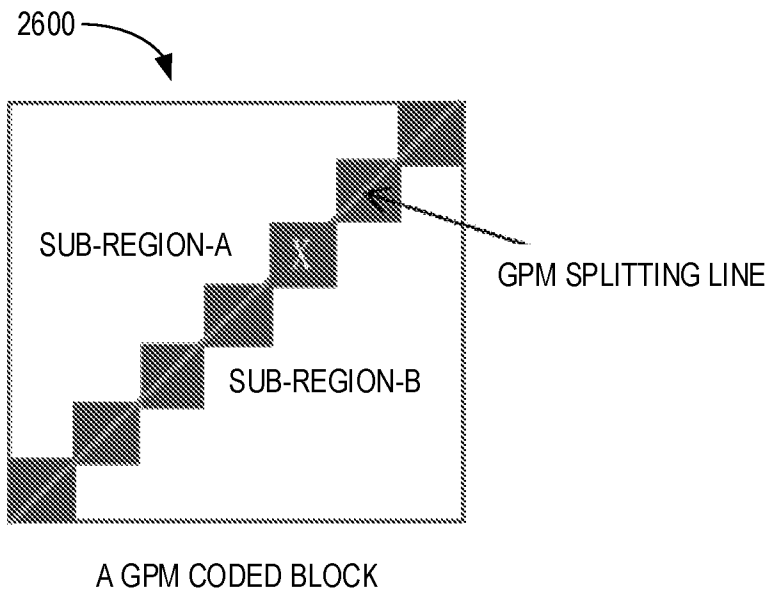
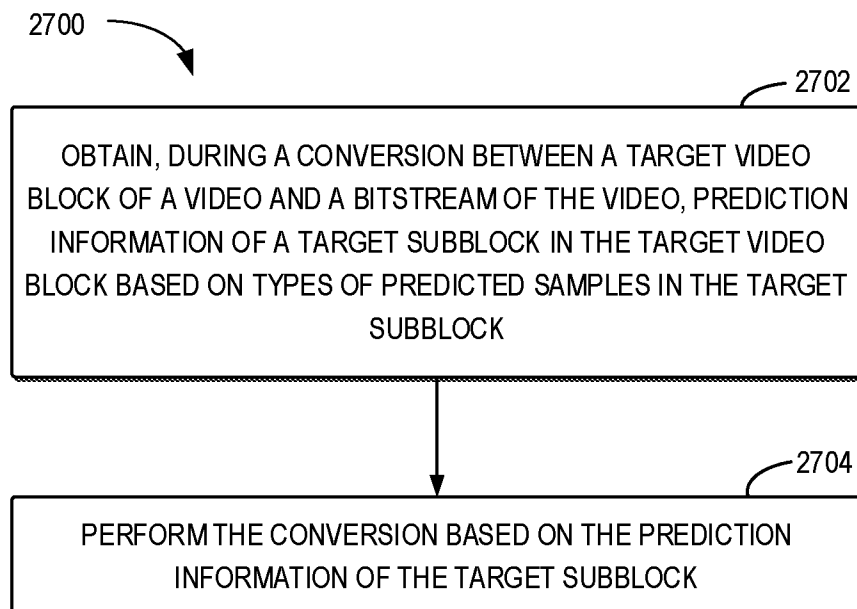
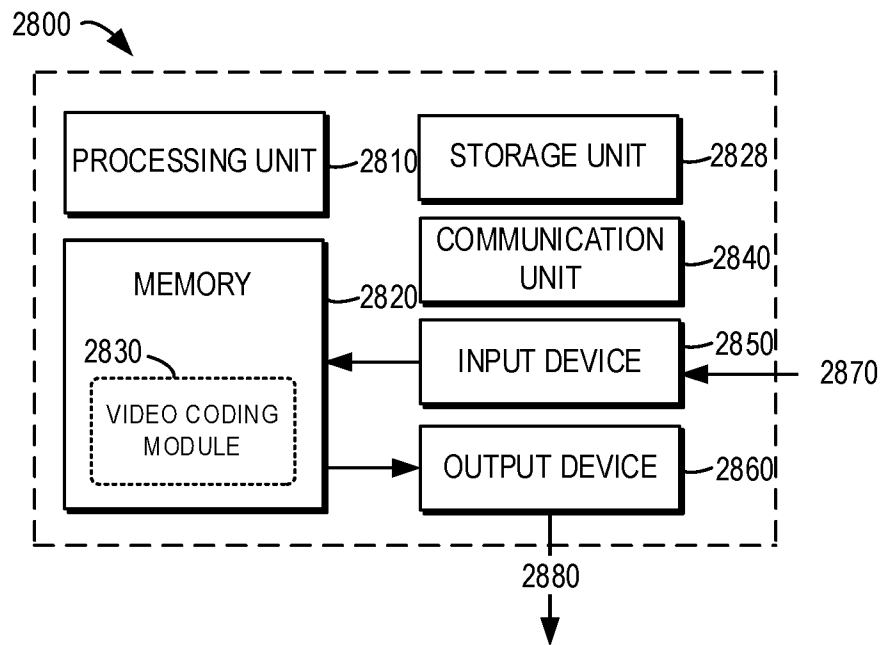


Fig. 25

**Fig. 26**

**Fig. 27**

**Fig. 28**

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2022/122197

A. CLASSIFICATION OF SUBJECT MATTER

H04N 19/119(2014.01)i; H04N 19/159(2014.01)i; H04N 19/169(2014.01)i

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04N

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

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

CNTXT, ENTXT, CNKI, WPABS, ENTXT: video, encode, decode, conversion, block, prediction, samples, geometric, partition, GPM, motion, inter, intra

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2020140862 A1 (BEIJING BYTEDANCE NETWORK TECH CO., LTD. et al.) 09 July 2020 (2020-07-09) the description paragraphs [005]-[00485], figs. 1-28	1-62
X	WO 2020143742 A1 (BEIJING BYTEDANCE NETWORK TECH CO., LTD. et al.) 16 July 2020 (2020-07-16) the description paragraphs [005]-[00368]	1-62
X	WO 2021041681 A1 (QUALCOMM INC.) 04 March 2021 (2021-03-04) the description paragraphs [0047]-[0274]	1-62
A	WO 2021133456 A1 (TENCENT AMERICA LLC.) 01 July 2021 (2021-07-01) the whole document	1-62
A	WO 2020251421 A2 (HUAWEI TECH CO., LTD. et al.) 17 December 2020 (2020-12-17) the whole document	1-62



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

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

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

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

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

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

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

“&” document member of the same patent family

Date of the actual completion of the international search

08 December 2022

Date of mailing of the international search report

30 December 2022

Name and mailing address of the ISA/CN

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

Facsimile No. (86-10)62019451

Authorized officer

XU, Liyuan

Telephone No. 86-010-62411404

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2022/122197

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
WO	2020140862	A1	09 July 2020	US	2022132120	A1	28 April 2022
				CN	113170166	A	23 July 2021
WO	2020143742	A1	16 July 2020	CN	113170139	A	23 July 2021
WO	2021041681	A1	04 March 2021	EP	4022921	A1	06 July 2022
				US	2021067776	A1	04 March 2021
				CN	114303371	A	08 April 2022
				TW	202123701	A	16 June 2021
WO	2021133456	A1	01 July 2021	EP	4082212	A1	02 November 2022
				KR	20210135311	A	12 November 2021
				CA	3134981	A1	01 July 2021
				SG	11202110603Q	A	28 October 2021
				US	2022210427	A1	30 June 2022
				CN	114651447	A	21 June 2022
				JP	2022526161	A	23 May 2022
				AU	2020414351	A1	28 October 2021
				US	2021195200	A1	24 June 2021
WO	2020251421	A2	17 December 2020	None			