



(51) International Patent Classification:

H04N 19/176 (2014.01) H04N 19/186 (2014.01)

H04N 19/107 (2014.01) H04N 19/159 (2014.01)

(21) International Application Number:

PCT/CN2022/101685

(22) International Filing Date:

27 June 2022 (27.06.2022)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

PCT/CN2021/102874

28 June 2021 (28.06.2021)

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

Published:

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

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

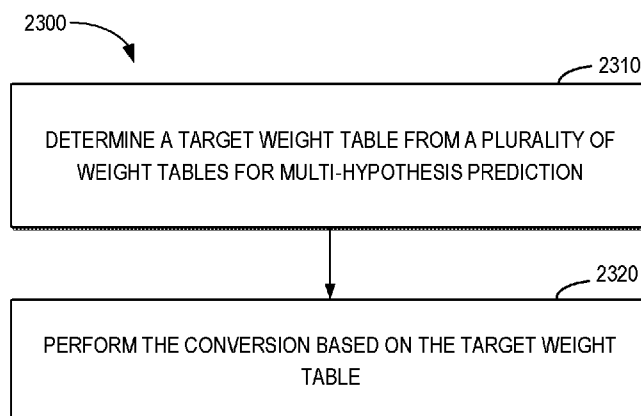


Fig. 23

(57) Abstract: Embodiments of the present disclosure provide a solution for video processing. A method for video processing is proposed. The method comprises: determining, during a conversion between a target block of a video and a bitstream of the target block, a target weight table from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; and performing the conversion based on the target weight table.

METHOD, DEVICE, AND MEDIUM FOR VIDEO PROCESSING

FIELD

[0001] Embodiments of the present disclosure relates generally to video coding techniques, and more particularly, to signaling of multiple hypothesis prediction in image/video coding.

BACKGROUND

[0002] In nowadays, digital video capabilities are being applied in various aspects of people's 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 proposed for video encoding/decoding. However, coding efficiency of conventional video coding techniques is generally 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: determining, during a conversion between a target block of a video and a bitstream of the target block, a target weight table from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; and performing the conversion based on the target weight table. Compared with the conventional solution, multiple weight tabled can be used for blending prediction blocks, thereby advantageously improving coding efficiency, coding performance, and flexibility.

[0005] In a second aspect, another method for video processing is proposed. The method comprises: determining, during a conversion between a target block of a video and a bitstream of the target block, whether a coding method is applied to a hypothesis of the target block based on coding information associated with the target block; and performing the conversion based on the determining. Compared with the conventional solution, the proposed method can advantageously improve the coding efficiency and performance.

[0006] In a third 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 determine, during a conversion between a target block of a video and a bitstream of the target block, a target weight table from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; and perform the conversion based on the target weight table.

[0007] In a fourth 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 determine, during a conversion between a target block of a video and a bitstream of the target block, whether a coding method is applied to a hypothesis of the target block based on coding information associated with the target block; and perform the conversion based on the determining.

[0008] In a fifth aspect, a non-transitory computer-readable storage medium is proposed. The non-transitory computer-readable storage medium stores instructions that cause a processor to determine, during a conversion between a target block of a video and a bitstream of the target block, a target weight table from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; and perform the conversion based on the target weight table.

[0009] In a sixth aspect, a non-transitory computer-readable storage medium is proposed. The non-transitory computer-readable storage medium stores instructions that cause a processor to determine, during a conversion between a target block of a video and a bitstream of the target block, whether a coding method is applied to a hypothesis of the target block based on coding information associated with the target block; and perform the conversion based on the determining.

[0010] In a seventh aspect, a non-transitory computer-readable recording medium is proposed. The 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: determining a target weight table applied to a target block of the video from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a

hypothesis of the target block, the target block being a multiple hypothesis prediction block; and generating a bitstream of the target block based on the target weight table.

[0011] In an eighth aspect, a non-transitory computer-readable recording medium is proposed. The 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: determining whether a coding method is applied to a hypothesis of the video based on coding information associated with the target block; and generating a bitstream of the target block based on the determining.

[0012] In a ninth aspect, a method for storing a bitstream of a video is proposed. The method comprises: determining a target weight table applied to a target block of the video from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; generating the bitstream based on the target weight table; and storing the bitstream in a non-transitory computer-readable recording medium.

[0013] In a tenth aspect, a method for storing a bitstream of a video is proposed. The method comprises: determining whether a coding method is applied to a hypothesis of a target block of the video based on coding information associated with the target block; generating the bitstream based on the determining; and storing the bitstream in a non-transitory computer-readable recording medium.

[0014] 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

[0015] 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.

[0016] Fig. 1 illustrates a block diagram that illustrates an example video coding system, in accordance with some embodiments of the present disclosure;

- [0017] Fig. 2 illustrates a block diagram that illustrates a first example video encoder, in accordance with some embodiments of the present disclosure;
- [0018] Fig. 3 illustrates a block diagram that illustrates an example video decoder, in accordance with some embodiments of the present disclosure;
- [0019] Fig. 4 is a schematic diagram of intra prediction modes;
- [0020] Fig. 5 illustrates a block diagram of reference samples for wide-angular intra prediction;
- [0021] Fig. 6 illustrates a schematic diagram of problem of discontinuity in case of directions beyond 45°;
- [0022] Fig. 7 illustrates a schematic diagram of definition of samples used by PDPC applied to diagonal and adjacent angular intra modes;
- [0023] Fig. 8 illustrates a schematic diagram of example of four reference lines neighboring to a prediction block;
- [0024] Fig. 9 illustrates a schematic diagram of sub-partition depending on the block size;
- [0025] Fig. 10 illustrates matrix weighted intra prediction process;
- [0026] Fig. 11 illustrates positions of spatial merge candidate;
- [0027] Fig. 12 illustrates candidate pairs considered for redundancy check of spatial merge candidates;
- [0028] Fig. 13 illustrates an illustration of motion vector scaling for temporal merge candidate;
- [0029] Fig. 14 illustrates candidate positions for temporal merge candidate, C0 and C1;
- [0030] Fig. 15 illustrates a schematic diagram of MMVD search point;
- [0031] Fig. 16 illustrates extended CU region used in BDOF;
- [0032] Fig. 17 illustrates an illustration for symmetrical MVD mode;
- [0033] Fig. 18 illustrates decoding side motion vector refinement;
- [0034] Fig. 19 illustrates top and left neighboring blocks used in CIIP weight derivation;
- [0035] Fig. 20 illustrates examples of the GPM splits grouped by identical angles;

[0036] Fig. 21 illustrates uni-prediction MV selection for geometric partitioning mode;

[0037] Fig. 22 illustrates exemplified generation of a bending weight w_0 using geometric partitioning mode;

[0038] Fig. 23 illustrates a flow chart of a method according to embodiments of the present disclosure;

[0039] Fig. 24 illustrates a flow chart of a method according to embodiments of the present disclosure; and

[0040] Fig. 25 illustrates a block diagram of a computing device in which various embodiments of the present disclosure can be implemented.

[0041] Throughout the drawings, the same or similar reference numerals usually refer to the same or similar elements.

DETAILED DESCRIPTION

[0042] 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.

[0043] 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.

[0044] 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.

[0045] 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.

[0046] 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

[0047] 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.

[0048] 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.

[0049] 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.

[0050] 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.

[0051] 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.

[0052] 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.

[0053] 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.

[0054] 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.

[0055] 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.

[0056] 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.

[0057] 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.

[0058] 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.

[0059] 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.

[0060] 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.

[0061] 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.

[0062] 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.

[0063] 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.

[0064] 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.

[0065] 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 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.

[0066] 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.

[0067] 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.

[0068] 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.

[0069] 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.

[0070] 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.

[0071] 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.

[0072] 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.

[0073] After the reconstruction unit 212 reconstructs the video block, loop filtering operation may be performed to reduce video blocking artifacts in the video block.

[0074] 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.

[0075] 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.

[0076] 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.

[0077] 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.

[0078] 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.

[0079] 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.

[0080] 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.

[0081] 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.

[0082] 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.

[0083] 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.

[0084] 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

The present disclosure is related to video coding technologies. Specifically, it is about the syntax signalling of a prediction mode. It may be applied to the 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 [1] 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

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. The new directional modes not in HEVC are depicted as red dotted arrows in Fig. 4, and the planar and DC modes remain the same. These denser directional intra 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 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 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

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.

- 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, $Max - 1$, $Max + 1$, $Max - 2$ }
- 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, $Max - 1$, $Max + 1$ }
 - Otherwise
 - MPM list $\rightarrow$ {Planar, Left, Above, DC, $Max - 2$, $Max + 2$ }
- If Left and Above are both angular and they are the same:
 - MPM list $\rightarrow$ {Planar, Left, $Left - 1$, $Left + 1$, DC, $Left - 2$ }

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

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 intra mode coding method is unchanged.

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

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

Fig. 6 illustrates a block diagram of discontinuity in case of directions beyond 45 degree. As shown in the diagram 600 of 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 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.

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.

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:

- 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;
- If the directional intra-prediction mode is classified as belonging to group A, 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 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 signaling: 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. 7 illustrates the definition of reference samples ($R_{x, -1}$, $R_{-1, y}$ and $R_{-1, -1}$) for PDPC applied over various prediction modes. Fig. 7 shows a diagonal top-right mode 710, a diagonal bottom-left mode 720, an adjacent diagonal top-right mode 730 and an adjacent diagonal bottom-left mode 740. 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	$16 \gg ((y' < 1) \gg \text{shift})$	$16 \gg ((x' < 1) \gg \text{shift})$	0
Diagonal bottom-left	$16 \gg ((y' < 1) \gg \text{shift})$	$16 \gg ((x' < 1) \gg \text{shift})$	0
Adjacent diagonal top-right	$32 \gg ((y' < 1) \gg \text{shift})$	0	0
Adjacent diagonal bottom-left	0	$32 \gg ((x' < 1) \gg \text{shift})$	0

2.1.1.6. Multiple reference line (MRL) intra prediction

Multiple reference line (MRL) intra prediction uses more reference lines for intra prediction. In Fig. 8, an example of 4 reference lines is depicted, where the samples of segments A and F are not fetched from reconstructed neighboring samples but padded with the closest samples from Segment B and E, respectively. HEVC intra-picture prediction uses the nearest reference line (i.e., 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 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 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 4×8 (or 8×4). If block size is greater than 4×8 (or 8×4) 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. 9 shows examples of the two possibilities. All sub-partitions fulfill the condition of having at least 16 samples. Fig. 9 shows an example 910 of sub-partitions for 4×8 and 8×4 CUs and an example 920 of sub-partitions for CUs other than 4×8 , 8×4 and 4×4 .

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. 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.

3.3.6.1 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 $bdry^{top}$ and $bdry^{left}$ 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:

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

3.3.6.2 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:

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

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 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 $b_2^i, i \in \{0, \dots, 5\}$ of size 64.

3.3.6.3 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 regardless of block shape or block size.

3.3.6.4 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, and MIP mode Id (*modelId*), which determines which matrix is to be used for the given MIP mode is derived as follows

$$isTransposed = predModelIntra \& 1$$

$$modelId = predModelIntra >> 1 \quad (2-6)$$

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)
- 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

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
- 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.

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

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 is a schematic diagram 1100 illustrating positions of a spatial merge candidate. A maximum of four merge candidates are selected among candidates located in the positions depicted in Fig. 11. The order of derivation is B_0 , A_0 , B_1 , A_1 and B_2 . Position B_2 is considered only when one or more than one CUs of position B_0 , A_0 , B_1 , A_1 are not available (e.g. because it belongs to another slice or tile) or is intra coded. After candidate at position A_1 is added, the addition of the remaining candidates is subject to a redundancy check 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 is a schematic diagram 1200 illustrating 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. The scaled motion vector for temporal merge candidate is obtained as illustrated by the dotted line in the diagram 1300 of 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 is a schematic diagram 1400 illustrating candidate positions for temporal merge candidate, 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. 15 is a schematic diagram 1500 illustrating a merge mode with motion vector differences (MMVD) search point. As shown in Fig. 15, 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)

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.

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

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 ($w \in \{3, 4, 5\}$) 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. When combined with AMVR, unequal weights are only conditionally checked for 1-pel and 4-pel motion 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
- 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

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.

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.,

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

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

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

$$S_5 = \sum_{(i,j) \in \Omega} Abs(\psi_y(i,j)), \quad S_6 = \sum_{(i,j) \in \Omega} \theta(i,j) \cdot Sign(\psi_y(i,j))$$

where

$$\begin{aligned} \psi_x(i,j) &= \left(\frac{\partial I^{(1)}}{\partial x}(i,j) + \frac{\partial I^{(0)}}{\partial x}(i,j) \right) \gg n_a \\ \psi_y(i,j) &= \left(\frac{\partial I^{(1)}}{\partial y}(i,j) + \frac{\partial I^{(0)}}{\partial y}(i,j) \right) \gg n_a \\ \theta(i,j) &= (I^{(1)}(i,j) \gg n_b) - (I^{(0)}(i,j) \gg n_b) \end{aligned} \quad (2-10)$$

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:

$$\begin{aligned} v_x &= S_1 > 0 ? \text{clip3} \left(-th'_{BIO}, th'_{BIO}, -((S_3 \cdot 2^{n_b - n_a}) \gg \lfloor \log_2 S_1 \rfloor) \right) : 0 \\ v_y &= S_5 > 0 ? \text{clip3} \left(-th'_{BIO}, th'_{BIO}, - \left((S_6 \cdot 2^{n_b - n_a} \right. \right. \\ &\quad \left. \left. - ((v_x S_{2,m}) \ll n_{S_2} + v_x S_{2,s}) / 2 \right) \gg \lfloor \log_2 S_5 \rfloor \right) : 0 \end{aligned} \quad (2-11)$$

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:

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

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

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

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.

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 of extended CU region used in BDOF. As depicted in the diagram 1600 of 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 (denoted as 1610 in Fig. 16) 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 (denoted as 1620 in Fig. 16). 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.

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.

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

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:

- 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.
- 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 is 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. 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 (BM) 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 is a schematic diagram illustrating the 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

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

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:

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

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

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

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 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 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 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. 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 a schematic diagram 1900 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 $m, n \in \{3 \cdots 6\}$ excluding 8x64 and 64x8.

Fig. 20 shows 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 (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. Fig. 21 is a schematic diagram illustrating the uni-prediction MV selection for geometric partitioning mode. Denote n as the index of the uni-prediction motion in the geometric uni-prediction candidate list 2110. 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. 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 . One example of weigh w_0 is illustrated in the schematic diagram 2200 of Fig. 22.

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 field 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. Multi-hypothesis prediction (MHP)

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.

3. Problems

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

- (1) A fixed weight table is used for blending two prediction/hypothetic blocks during the MHP coding, which may be not flexible enough.
- (2) The MHP allows a limited number of prediction methods as the base hypothesis, which may not achieve the highest efficiency.
- (3) The MHP generates additional hypothesis based on a set of rule, while regular merge and regular AMVP are not allowed to be used as an additional hypothesis, which may not achieve the highest efficiency.

4. Embodiments of the present disclosure

The detailed embodiments 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.

In this invention, 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, OBMC, and etc.).

A “multiple hypothesis prediction” in this invention 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, OBMC, 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, and generally, a “base hypothesis” may be a prediction unit generated from a certain prediction mode (such as MODE_INTER, or MODE_INTRA, and etc.).

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, and generally, syntax elements for an “additional hypothesis” are additionally signaled associated with syntax elements of a “base hypothesis”. There might be more than one additional hypothesis associated with a “base hypothesis”. A multiple hypothesis prediction video unit is generally a hybrid prediction unit, in which final prediction samples are blended from a “base hypothesis” and one or more “additional hypotheses”.

On signalling of multiple hypothesis prediction

1. In one example, more than one weight table (e.g., a set of weighting factors) may be defined for multiple hypothesis prediction.
 - 1) For example, which weighting table is used for a hypothesis of a multiple hypothesis predicted coding unit, may be signaled in the bitstream.
 - 2) For example, which weighting table is used (e.g., for a hypothesis) for a multiple hypothesis predicted coding unit, may be determined by one or more rules.
 - a) For example, the rule may be dependent on the prediction method of the base hypothesis.
 - b) For example, the rule may be dependent on the prediction method of an additional hypothesis.
 - c) For example, the rule may be dependent on the width and/or height of the coded sequence.
 - d) For example, the rule be dependent on the width and/or height of the coded unit.
 - 3) For example, the dimensions of a weighting table may depend on the dimensions of the block which it is applied on.
2. In one example, whether a hypothesis is allowed to be coded with a coding method **M** may be dependent on coded information.
 - 1) For example, only if a coding unit is allowed to be coded as base hypothesis, multiple hypothesis information of this coding unit may be signalled thereafter.
 - a) For example, a coding unit is coded with prediction method **M** but not satisfy certain conditions, then it is not allowed to be perceived as base hypothesis, such that it should not signal multiple hypothesis information associated with the coding unit.
 - 2) For example, whether a coding unit with prediction method **M** is allowed to be as a hypothesis is dependent on prediction methods of neighboring/adjacent coding units.

- a) For example, if specified position/location of neighboring video units (e.g., from above, left, above-right, above-left, bottom-left of the current video unit) are coded with prediction method N, the current coding unit coded with prediction method M is allowed to be as a base hypothesis.
 - i. Alternatively, if specified position/location of neighboring video units are coded with prediction method N, an additional hypothesis associated with the base hypothesis is allowed to be coded with prediction method M.
 - b) For example, if a specified number of neighboring video units are coded with prediction method N, the current coding unit coded with prediction method M is allowed to be as a base hypothesis.
 - i. Alternatively, if a specified number of neighboring video units are coded with prediction method N, an additional hypothesis associated with the base hypothesis is allowed to be coded with prediction method M.
 - c) For example, if a specified number of neighboring video units at specified position/location are coded with prediction method N, the current coding unit coded with prediction method M is allowed to be as a base hypothesis.
 - i. Alternatively, if a specified number of neighboring video units at specified position/location are coded with prediction method N, an additional hypothesis associated with the base hypothesis is allowed to be coded with prediction method M.
 - d) For example, the coding method N may be INTRA, intra Planar, intra DC, PDPC, intra angular, BDPCM, MIP, MRL, ISP, LM, IBC, INTER, affine, affine merge, sbTMVP, SMVD, CIIP, CIIP PDPC, GPM, TM, DMVR, and etc.
- 3) For example, whether an additional hypothesis is allowed to be coded with prediction method M is dependent on prediction methods of its base hypothesis.
- a) For example, if the current coding unit as base hypothesis is coded with prediction method N, it is then allowed to have additional hypothesis coded with prediction method M.
- 4) For example, for those prediction methods which are not allowed to a certain additional hypothesis, there is no need to signal such prediction related syntax elements (e.g., the corresponding syntax elements may be inferred to default values instead of signalling) to the multiple hypothesis information.
- 5) For example, prediction method M may be INTRA, intra Planar, intra DC, PDPC, intra angular, BDPCM, MIP, MRL, ISP, LM, IBC, INTER, affine, affine merge, sbTMVP, SMVD, CIIP, CIIP PDPC, GPM, TM, DMVR, and etc.
3. In one example, multiple hypothesis information (e.g., coding info of additional hypotheses, weight index, and so on) may be signaled associated with the coding unit level syntax

elements, in case that the current coding unit (i.e., base hypothesis) is coded with one or more of the following methods.

- 1) For example, the current coding unit is coded with MODE_INTRA.
- 2) For example, the current coding unit is coded with single tree.
- 3) For example, the current coding unit is coded with MODE_IBC.
- 4) For example, the current coding unit is coded with BDPCM.
- 5) For example, the current coding unit is coded with MIP.
- 6) For example, the current coding unit is coded with MRL.
- 7) For example, the current coding unit is coded with ISP.
- 8) For example, the current coding unit is coded with Intra DC prediction.
- 9) For example, the current coding unit is coded with Intra Planar prediction.
- 10) For example, the current coding unit is coded with Intra angular prediction.
- 11) For example, the current coding unit is coded with PDPC.
- 12) For example, the current coding unit is coded with LM mode.
- 13) For example, the current coding unit is coded with DM mode.
- 14) For example, the current coding unit is coded with CIIP mode.
- 15) For example, the current coding unit is coded with CIIP mode but not CIIP PDPC mode.
 - a) For example, given a CIIP PDPC coded video unit, multiple hypothesis information (such as multiple hypothesis flag for each additional hypothesis) may be signalled before the CIIP PDPC flag.
 - b) For example, given a CIIP PDPC coded video unit, multiple hypothesis information (such as multiple hypothesis flag for each additional hypothesis) may be signalled before the merge index.
 - c) For example, given a CIIP PDPC coded video unit, multiple hypothesis information (such as multiple hypothesis flag for each additional hypothesis) may be signalled within the merge data syntax structure.
 - d) For example, given a coding unit, whether CIIP PDPC flag is signalled or not may be dependent on whether multiple hypothesis prediction is used to the coding unit.
 - i. For example, only if multiple hypothesis prediction is not used to the coding unit (e.g., multiple hypothesis flags for all additional hypotheses are equal to false), CIIP PDPC flag is signalled.
 - ii. For example, if multiple hypothesis prediction is used to the coding unit (e.g., multiple hypothesis flag for at least one additional hypothesis is equal to true), CIIP PDPC flag is not signalled but inferred to be equal to a value (e.g., equal to 0 indicating CIIP_PDPC is not used to the coding unit).
- 16) For example, the current coding unit is coded with CIIP PDPC mode.
- 17) For example, the current coding unit is coded with GPM mode.
- 18) For example, the current coding unit is coded with template matching based refinement method.

- 19) For example, the current coding unit is coded with bilateral matching based refinement method.
 - 20) Alternatively, whether multiple hypothesis information is signaled or not, may be dependent on whether one or more of the above conditions are satisfied.
 - 21) Alternatively, whether multiple hypothesis information is signaled or not, may be dependent on whether one or more of the above conditions are not satisfied.
 - 22) Additionally, in case that multiple hypothesis information is allowed to be signaled and the base hypothesis is coded with specific prediction modes (e.g., MODE_INTRA, MODE_INTER, and etc.) associated with specific prediction methods (e.g., intra planar, intra angular prediction, PDPC, regular merge, MMVD, affine merge, CIIP, GPM, TM, DMVR, and etc.), one or more of the following rules may be conformed.
 - a) For example, the multiple hypothesis information is present for specified prediction modes but not present for some other prediction modes (e.g., MODE_IBC, MODE_PLT, and etc.).
 - b) For example, the multiple hypothesis information is present for specified prediction modes but not present for some other prediction methods (e.g., intra LM, intra BDPCM, intra ISP, intra MIP, and etc.).
4. Multiple hypothesis prediction may be applied to a video unit wherein subblock-block based coding method is applied.
 - 1) In one example, for affine inter coded video units, multiple hypothesis prediction may be applied no matter the BCW index is equal to the default value or not.
 - 2) In one example, for subblock based DMVR coded video units, multiple hypothesis prediction may be applied.
 - 3) In one example, for subblock based BDOF coded video units, multiple hypothesis prediction may be applied.
 - 4) In one example, for subblock based PROF coded video unit,s multiple hypothesis prediction may be applied.
 5. Multiple hypothesis prediction may be applied to a video unit wherein AMVP based coding method is applied.
 - 1) In one example, for uni-directional (e.g, L0 or L1) predicted AMVP video units, multiple hypothesis prediction may be applied.
 - 2) In one example, for bi-directional (e.g, BI) predicted AMVP video units, multiple hypothesis prediction may be applied no matter the BCW index is equal to the default value or not.

- 3) In one example, for bi-directional (e.g., BI) predicted AMVP video units, multiple hypothesis prediction may be applied even though the BCW index is equal to the default value.
 - 4) For example, above mentioned AMVP coded video units may be non-affine AMVP video units.
 - 5) For example, above mentioned AMVP coded video units may be affine AMVP video units.
6. In one example, once a multiple hypothesis prediction is used on a video unit (e.g., a coding unit), at least one message corresponding to multiple hypothesis information (i.e., coding information of additional hypotheses other than the base hypothesis) is signalled associated with the video unit, wherein the multiple hypothesis information may comprise one or more of the following coded information.
- 1) Whether the additional hypothesis is coded with `MODE_INTER` or `MODE_INTRA`.
 - 2) In case that the additional hypothesis is coded with `MODE_INTRA`,
 - a) For example, it may further signal whether the additional hypothesis is coded with prediction method A.
 - i. Additionally, the additional hypothesis may be coded by method A, under the condition that the base hypothesis is not coded by method X (such as X may or may not be the same as A).
 - b) Alternatively, it may be mandatorily use method A for an INTRA coded additional hypothesis, such that there is no need to signal whether the additional hypothesis is coded with method A.
 - i. For example, a pre-defined INTRA prediction method is mandatorily used, without additional signalling.
 - ii. For example, whether a mandatory prediction method is used for an INTRA coded additional hypothesis, may be dependent on the coding unit size (such as width and/or height).
 - c) Alternatively, the additional hypothesis is not allowed to be coded by method A, such that there is no need to signal whether the additional hypothesis is coded with method A.
 - d) For example, method A may be MIP.
 - i. In such case, the MIP transpose flag and/or MIP mode index may be further signalled.
 - ii. Alternatively, default MIP transpose method and/or MIP mode index may be used in such case, then there is no need to signal additional MIP information.
 - e) For example, method A may be MRL.
 - i. In such case, the reference index of MRL may be further signalled.

- ii. Alternatively, a default MRL index other than the nearest reference line may be used in such case, then there is no need to signal additional MRL information.
- f) For example, method A may be ISP.
 - i. In such case, split direction of ISP may be further signalled.
 - ii. Alternatively, a default ISP split direction may be used in such case, then there is no need to signal additional ISP information.
- g) For example, method A may be Intra DC prediction.
- h) For example, method A may be Intra Planar prediction.
- i) For example, method A may be regular Intra prediction.
 - i. In such case, intra prediction mode index may be further signalled.
 - ii. Alternatively, a default intra mode (such as horizontal, vertical, or specific angular mode) may be used in such case, then there is no need to signal additional intra information.
- j) For example, method A may be Intra DM prediction.
- k) For example, method A may be Intra LM prediction.
 - i. In such case, LM mode index may be further signalled.
 - ii. Alternatively, a default LM mode may be used in such case, then there is no need to signal additional LM information.
- l) For example, method A may be PDPC.
 - i. For example, a PDPC predictor generated from neighboring samples only (rather than both neighboring samples and current prediction samples), may be used as an additional hypothesis.
 - ii. Alternatively, a PDPC predictor associated with an INTRA prediction mode may be used as an additional hypothesis, and it may follow the same rule of “PDPC on regular intra mode” to apply “PDPC on an INTRA coded additional hypothesis” – whether to apply PDPC is determined on the intra prediction mode index – such that there is no need to signal whether the additional INTRA code hypothesis uses PDPC.
- m) For example, method A may be BDPCM.
 - i. In such case, the BDPCM prediction direction may be further signalled.
 - ii. Alternatively, a default BDPCM prediction direction may be used, then there is no need to signal additional BDPCM information.
- 3) In case that the additional hypothesis is coded with `MODE_INTER`,
 - a) For example, it may further signal whether the additional hypothesis is coded with prediction method B.
 - i. Additionally, the additional hypothesis may be coded by method B, under the condition that the base hypothesis is not coded by method M (such as M may or may not be the same as B).

- b) Alternatively, it may be mandatorily use method B for an INTER coded additional hypothesis, such that there is no need to signal whether the additional hypothesis is coded with method B.
 - i. For example, a pre-defined INTER prediction method is mandatorily used, without additional signalling.
 - ii. For example, whether a mandatory prediction method is used for an INTER coded additional hypothesis, may be dependent on the coding unit size (such as width and/or height).
 - c) Alternatively, the additional hypothesis is not allowed to be coded by method B, such that there is no need to signal whether the additional hypothesis is coded with method B.
 - d) For example, method B may be CIIP.
 - e) For example, method B may be GPM.
 - f) For example, method B may be MMVD.
 - g) For example, method B may be SMVD.
 - h) For example, method B may be Affine.
 - i) For example, method B may be SbTMVP.
 - j) For example, method B may be PROF.
 - k) For example, method B may be BDOF.
 - l) For example, method B may be regular Merge mode.
 - i. For example, additional hypothesis may be coded by a regular merge mode, but its merge index is different from the merge index of the base hypothesis.
 - m) For example, method B may be regular AMVP mode.
 - i. For example, additional hypothesis may be coded by a regular AMVP mode, but its AMVP index is different from the AMVP index of the base hypothesis.
 - n) For example, method B may be a template matching based refinement.
 - o) For example, method B may be a bilateral matching based refinement.
 - p) For example, method B may be PDPC.
 - i. For example, a PDPC predictor generated from neighboring samples only (rather than both neighboring samples and current prediction samples), may be used as an additional hypothesis.
7. For a block coded with CIIP mode, at least one message related to motion vector difference may be signaled to refine the motion information of the inter-prediction used in CIIP.
- 1) In one example, the message of motion vector difference may be coded in the same way as the motion vector difference coded for MMVD mode.
 - 2) In one example, it is possible that CIIP mode and MMVD mode are used together for a single block, wherein the inter-prediction in CIIP mode is generated based on the MMVD mode.

General claims

8. 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.
9. 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.
10. 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.

[0085] Embodiments of the present disclosure are related to determining whether to and/or how to apply an optical flow based coding method to a video unit based on illuminance information.

[0086] As used herein, the terms “video unit” or “coding unit” or “block” used herein may refer to one or more of: a color component, a sub-picture, a slice, a tile, a coding tree unit (CTU), a CTU row, a group of CTUs, a coding unit (CU), a prediction unit (PU), a transform unit (TU), a coding tree block (CTB), a coding block (CB), a prediction block (PB), a transform block (TB), a block, a sub-block of a block, a sub-region within the block, or a region that comprises more than one sample or pixel.

[0087] In this present disclosure, regarding “a block coded with mode N”, the term “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.).

[0088] A “multiple hypothesis prediction” in this present 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 the like.

[0089] 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. 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.

[0090] Fig. 23 illustrates a flowchart of a method 2300 for video processing in accordance with some embodiments of the present disclosure. The method 2300 may be implemented during a conversion between a video unit and a bitstream of the video unit.

[0091] At block 2310, a target weight table is determined from a plurality of weight tables for multi-hypothesis prediction. The target weight table is used for a hypothesis of the target block. The target block is a multiple hypothesis prediction block. For example, more than one weight table may be defined for multiple hypothesis prediction. In some embodiments, a set of weighting factors may be defined for multiple hypothesis prediction.

[0092] At block 2320, the conversion is performed based on the target weight table. In some embodiments, the conversion may comprise encoding the target block into the bitstream. In some embodiments, the conversion may comprise decoding the target block from the bitstream.

[0093] According to embodiments of the present disclosure, multiple weight tabled can be used for blending prediction blocks. Compared with the conventional solution, some embodiments of the present disclosure can advantageously improve improving the coding efficiency, coding performance, and flexibility.

[0094] In some embodiments, the target weight table used for the hypothesis of the target block may be indicated in the bitstream. For example, which weighting table is used for the hypothesis of the multiple hypothesis predicted coding unit may be indicated/signaled in the bitstream.

[0095] In some embodiments, the target weight tabled may be determined based on one or more of: a first prediction method of a base hypothesis of the target block, a second prediction method of an additional hypothesis of the target block, a width of a coded sequence associated with the target block, a height of the coded sequence associated with the target block, a width of the target block, or a height of the target block. For example, which weighting table is used (e.g., for a hypothesis) for a multiple hypothesis predicted coding unit, may be determined by one or more rules. In some embodiments, the rule may be dependent on the prediction method of the base hypothesis. For example, the rule may be dependent on the prediction method of an additional hypothesis. For example, the rule may be dependent on the width and/or height of the coded sequence. For example, the rule be dependent on the width and/or height of the coded unit. The one or more rules may also comprise other rules.

[0096] In some embodiments, a dimension of the target weight table may be determined based on a dimension of the target block. For example, the dimension of a weight table may depend on the dimension of the block on which the weight table is applied.

[0097] In some embodiments, an indication of whether to and/or how to determine the target weight table may be indicated at one of the followings: sequence level, group of pictures level, picture level, slice level, or tile group level. In some embodiments, an indication of whether to and/or how to determine the target weight table may be indicated in one of the following: a sequence header, a picture header, a sequence parameter set (SPS), a video parameter set (VPS), a dependency parameter set (DPS), a decoding capability information (DCI), a picture parameter set (PPS), an adaptation parameter sets (APS), a slice header, or a tile group header.

[0098] In some embodiments, an indication of whether to and/or how to determine the target weight table may be included in one of the following: a prediction block (PB), a transform block (TB), a coding block (CB), a prediction unit (PU), a transform unit (TU), a coding unit (CU), a virtual pipeline data unit (VPDU), a coding tree unit (CTU), a CTU row, a slice, a tile, a sub-picture, or a region containing more than one sample or pixel.

[0099] In some embodiments, whether and/or how the target weight table is applied may be determined based on coded information of the target block. For example, the coded information may comprise at least one of: a block size, a colour format, a single and/or dual tree partitioning, a colour component, a slice type, or a picture type.

[00100] 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, a target weight table is determined from a plurality of weight tables for multi-hypothesis prediction. The target weight table may be used for a hypothesis of the target block and the target block may be a multiple hypothesis prediction block. A bitstream of the target block may be generated based on the target weight table.

[00101] In some embodiments, a target weight table may be determined from a plurality of weight tables for multi-hypothesis prediction. The target weight table may be used for a hypothesis of the target block and the target block may be a multiple hypothesis prediction block. A bitstream of the target block may be generated based on the target weight table and stored in a non-transitory computer-readable recording medium.

[00102] Fig. 24 illustrates a flowchart of a method 2400 for video processing in accordance with some embodiments of the present disclosure. The method 2400 may be implemented during a conversion between a video unit and a bitstream of the video unit.

[00103] At block 2410, during a conversion between a video unit and a bitstream of the video unit, whether a coding method is applied to a hypothesis of the target block based on coding information associated with the target block is determined. In other words, whether a hypothesis is allowed to be coded with a coding method may be dependent on coded information.

[00104] At block 2420, the conversion is performed based on the determining. In some embodiments, the conversion may comprise encoding the target block the bitstream. In some embodiments, the conversion may comprise decoding the target block from the bitstream.

[00105] According to embodiments of the present disclosure, it proposes improved the coding method for a multiple hypothesis prediction block. Compared with the conventional solution, some embodiments of the present disclosure can advantageously improve the coding efficiency and coding performance.

[00106] In some embodiments, if the target block is allowed to be coded as a base hypothesis, multiple hypothesis information associated the target block may be indicated. For example, only if a coding unit is allowed to be coded as base hypothesis, multiple hypothesis information of this coding unit may be signaled thereafter. In some embodiments, if the target block is coded with the coding method but does not satisfy a condition, multiple hypothesis information associated the target block may not be indicated. For example, a coding unit (i.e., the target block) is coded with prediction method *M* (i.e., the coding method) but does not satisfy certain conditions, then it is not allowed to be perceived as base hypothesis, such that it may not signal multiple hypothesis information associated with the coding unit.

[00107] In some embodiments, whether the target block applied with the coding method is allowed to be as a hypothesis may depend on a further coding method of a neighbor/adjacent block. For example, in some embodiments, if the neighbor block at a specified position/location is applied with the other coding method, the target block applied with the coding method may be allowed to be as a base hypothesis. The specified position/location may refer to one or more of: from above, left, above-right, above-left, bottom-left of the current video unit. For example, if specified position/location of neighboring video units (e.g., from above, left, above-right, above-left, bottom-left of the current video unit) are coded with prediction method *N*, the current coding unit coded with prediction method *M* may not be allowed to be as a base hypothesis.

In some embodiments, if the neighbor block at a specified position/location is applied with the other coding method, an additional hypothesis associated with a base hypothesis may be allowed to be applied with the coding method.

[00108] In some embodiments, if a specified number of neighbor blocks are applied with the other coding method, the target block applied with the coding method may be allowed to be as a base hypothesis. In some embodiments, if the specified number of neighbor blocks are applied with the other coding method, an additional hypothesis associated with a base hypothesis may be allowed to be applied with the coding method.

[00109] In some embodiments, if a specified number of neighbor blocks at a specified position are applied with the other coding method, the target block applied with the coding method may be allowed to be as a base hypothesis. In some embodiments, if the specified number of neighbor blocks at the specified position are applied with the other coding method, an additional hypothesis associated with a base hypothesis may be allowed to be applied with the coding method.

[00110] In some embodiments, an additional hypothesis is allowed to be coded with the coding method may depend on a further coding method of a base hypothesis of the additional hypothesis. For example, if the base hypothesis is applied with the coding method, the additional hypothesis may be applied with the further coding method. In other words, if the current coding unit as base hypothesis is coded with prediction method N, it is then allowed to have additional hypothesis coded with prediction method M.

[00111] In some embodiments, for a prediction method is not allowed to an additional hypothesis, a syntax element related to the prediction method may not be indicated. For example, for those prediction methods which are not allowed to a certain additional hypothesis, there is no need to signal such prediction related syntax elements (e.g., the corresponding syntax elements may be inferred to default values instead of signaling) to the multiple hypothesis information.

[00112] In some embodiments, the coding method comprises one or more of: an intra prediction method, an intra planar prediction method, an intra direct current (DC) prediction method, a position dependent intra prediction combination (PDPC) method, an intra angular prediction method, a block-based differential pulse-code modulation (BDPCM) method, a matrix weighted intra prediction (MIP) method, a multiple reference line (MRL) method, an intra sub-partitions (ISP) method, a linear model (LM) method, an intra block copy (IBC) method, an

inter prediction method, an affine method, an affine merge method, a subblock-based temporal motion vector prediction (sbTMVP) method, a symmetric motion vector difference (SMVD) method, a combined inter and intra prediction (CIIP) method, a CIIP PDPC method, a geometric portioning mode (GPM) method, a temporal motion (TM) method, or a decoder side motion vector refinement (DMVR) method.

[00113] In some embodiments, multiple hypothesis information may be indicated, if the target block is coded with at least one of the followings: an intra prediction method, a single tree, an IBC mode, a BDPCM method, a MIP method, a MRL method, an ISP method, an intra DC prediction method, an intra planar prediction method, an intra angular prediction method, a PDPC method, a LM method, a derived mode (DM) method, a CIIP method, a CIIP method but not CIIP PDPC method, a CIIP PDPC method, a GPM method, a template matching based refinement method, or a bilateral matching based refinement method.

[00114] In some embodiments, for the target block is coded with the CIIP PDPC method, multiple hypothesis information may be indicated before a CIIP PDPC flag. In some embodiments, for the target block coded with the CIIP PDPC method, multiple hypothesis information may be indicated before a merge index. In some embodiments, for the target block coded with the CIIP PDPC method, multiple hypothesis information may be indicated a merge data syntax structure. In some embodiments, the multiple hypothesis information may comprise multiple hypothesis flag for each additional hypothesis.

[00115] In some embodiments, whether a CIIP PDPC flag is indicated may depend on whether a multiple hypothesis prediction is used to the target block. In some embodiments, if the multiple hypothesis prediction is not applied to the target block, a CIIP PDPC flag may be indicated. For example, only if multiple hypothesis prediction is not used to the coding unit (e.g., multiple hypothesis flags for all additional hypotheses are equal to false), CIIP PDPC flag may be signaled. In some embodiments, if the multiple hypothesis prediction is applied to the target block, a CIIP PDPC flag may not be indicated and may be set to a predefined value. For example, if multiple hypothesis prediction is used to the coding unit (e.g., multiple hypothesis flag for at least one additional hypothesis is equal to true), CIIP PDPC flag is not signaled but inferred to be equal to a value (e.g., equal to 0 indicating CIIP_PDPC is not used to the coding unit).

[00116] In some embodiments, whether multiple hypothesis information is indicated may depend on whether at least one condition is satisfied. In some embodiments, whether multiple

hypothesis information is indicated may depend on whether the at least one condition is not satisfied.

[00117] In some embodiments, if multiple hypothesis information is indicated and a base hypothesis is coded with a prediction mode associated with a prediction method, at least one of the followings may be conformed: the multiple hypothesis information is present for the specified prediction modes but not present for other prediction modes, or the multiple hypothesis information is present for the specified prediction mode but not present for other prediction methods. For example, the multiple hypothesis information is present for specified prediction modes but not present for some other prediction modes (e.g., `MODE_IBC`, `MODE_PLT`, and the like). For example, the multiple hypothesis information is present for specified prediction modes but not present for some other prediction methods (e.g., intra LM, intra BDPCM, intra ISP, intra MIP, and the like). In some embodiments, the prediction mode which is used for coding the base hypothesis may comprise at least one of: `MODE_INTRA`, `MODE_INTER` and the like. The prediction method which is associated with the prediction mode may comprise one or more of: intra planar, intra angular prediction, PDPC, regular merge, MMVD, affine merge, CIIP, GPM, TM, DMVR, and the like.

[00118] In some embodiments, if a subblock-block based coding method is applied, multiple hypothesis information may be applied to the target block. In some embodiments, for the target block coded with an affine inter method, the multiple hypothesis prediction may be applied regardless of a bi-prediction with coding unit level weight (BCW) index being equal to a default value or not. In some embodiments, for the target block coded with a subblock based DMVR method, the multiple hypothesis prediction may be applied. In some embodiments, for the target block coded with a subblock based bi-directional optical flow (BDOF) method, the multiple hypothesis prediction may be applied. In some embodiments, for the target block coded with a subblock based prediction refinement with optical flow (PROF) method, the multiple hypothesis prediction may be applied.

[00119] In some embodiments, if an advanced motion vector prediction (AMVP) based coding method is applied, multiple hypothesis information may be applied to the target block. In some embodiments, for the target block with a uni-directional (for example, L0 and L1) predicted AMVP, the multiple hypothesis information may be applied to the target block. In some embodiments, for the target block with a bi-directional (for example, BI) predicted AMVP, the multiple hypothesis information may be applied to the target block regardless of a bi-prediction

with coding unit level weight (BCW) index being equal to a default value or not. In some embodiments, for the target block with a bi-directional predicted AMVP, the multiple hypothesis information may be applied to the target block even though a BCW index is equal to the default value. In some embodiments, the target block may be a non-affine AMVP block. In some embodiments, the target block may be an affine AMVP block.

[00120] In some embodiments, if a multiple hypothesis prediction is applied to the target block, at least one message corresponding to multiple hypothesis information may be indicated associated with the target block. The multiple hypothesis information may comprise coding information of additional hypotheses other than the base hypothesis. The multiple hypothesis information may comprise coding information of the target block. In some embodiments, the coding information may comprise whether an additional hypothesis is coded with an inter prediction method or an intra prediction method. In some embodiments, if the additional hypothesis is coded with the intra prediction mode, whether the additional hypothesis may be coded with a first prediction method is indicated. In some embodiments, the additional hypothesis is coded with the first prediction method under a condition that a base hypothesis may not be coded by a second prediction method.

[00121] In some embodiments, the first prediction method may be mandatorily used for an intra coded additional hypothesis and the additional hypothesis may be coded with the first prediction method is not indicated. In some embodiments, a pre-defined intra prediction method may be mandatorily used. In some embodiments, whether a mandatory prediction method is used for an intra coded additional hypothesis may depend on a size of the target block. In some embodiments, if an additional hypothesis is not allowed to be coded by the first prediction method, whether the additional hypothesis is coded with the first prediction method may not be indicated.

[00122] In some embodiments, the first prediction method may be a MIP method. In some embodiments, a MIP transpose flag and/or a MIP mode index may be indicated. In some embodiments, a default MIP transpose flag and/or a default MIP mode index may be used, and additional MIP information may not be indicated.

[00123] In some embodiments, the first prediction method is a MRL method. In some embodiments, a reference index of MRL may be indicated. In some embodiments, a default MRL index may be used, and additional MIP information may not be indicated.

[00124] In some embodiments, the first prediction method may be an ISP method. In some embodiments, a split direction of ISP may be indicated. In some embodiments, a default ISP split direction may be used, and additional MIP information may not be indicated.

[00125] In some embodiments, the first prediction method may be an intra DC prediction method. In some embodiments, the first prediction method is an intra planar prediction method.

[00126] In some embodiments, the first prediction method is a regular intra prediction method. In some embodiments, an intra prediction mode may be indicated. In some embodiments, a default intra mode is used, and additional MIP information may not be indicated.

[00127] In some embodiments, the first prediction method may be an intra DM prediction method. In some embodiments, the first prediction method may be an intra LM prediction method. In some embodiments, a LM prediction method may be indicated. In some embodiments, a default LM method may be used, and additional LM information may be indicated.

[00128] In some embodiments, the first prediction method may be a PDPC method. In some embodiments, a PDPC predictor generated from a neighbor sample may be used as an additional hypothesis. In some embodiments, a PDPC predictor associated with an intra prediction mode may be as an additional hypothesis, whether to apply PDPC may be determined on an intra prediction mode index, and whether an additional intra code hypothesis uses PDPC may not be indicated.

[00129] In some embodiments, the first prediction method may be a BDPCM method. In some embodiments, a BDPCM prediction direction may be indicated. In some embodiments, a default BDPCM prediction direction may be used, and additional BDPCM information may not be indicated.

[00130] In some embodiments, an additional hypothesis may be coded with an inter prediction method. In some embodiments, whether the additional hypothesis may be coded with a third prediction method is indicated. In some embodiments, the additional hypothesis may be coded with the third prediction method under a condition that a base hypothesis is not coded by a fourth prediction method.

[00131] In some embodiments, the third prediction method may be mandatorily used for an inter coded additional hypothesis, whether the additional hypothesis is coded with the third prediction method may not be indicated. In some embodiments, a pre-defined inter prediction method may be mandatorily used. In some embodiments, whether a mandatory prediction

method is used for an inter coded additional hypothesis may depend on a size of the target block. In some embodiments, if an additional hypothesis is not allowed to be coded by the third prediction method, whether the additional hypothesis is coded with the third prediction method may not be indicated.

[00132] In some embodiments, the third prediction method may be a CIIP method. In some embodiments, the third prediction method may be a GPM method. In some embodiments, the third prediction method may be a MMVD method. In some embodiments, the third prediction method may be a SMVD method. In some embodiments, the third prediction method may be an affine method. In some embodiments, the third prediction method may be a sbTMVP method. In some embodiments, the third prediction method may be a PROF method.

[00133] In some embodiments, the third prediction method may be a BDOF method. In some embodiments, the third prediction method may be a merge method. In some embodiments, an additional hypothesis may be coded by the merge method, and a merge index associated with the additional hypothesis may be different from a merge index associated with a base hypothesis.

[00134] In some embodiments, the third prediction method may be an AMVP method. In some embodiments, an additional hypothesis may be coded by the AMVP method, and an AMVP index associated with the additional hypothesis may be different from an AMVP index associated with a base hypothesis.

[00135] In some embodiments, the third prediction method may be a template matching based refinement. In some embodiments, the third prediction method may be a bilateral matching based refinement. In some embodiments, the third prediction method may be a PDPC. In some embodiments, a PDPC predictor generated from a neighbor sample may be used as an additional hypothesis.

[00136] In some embodiments, for the target block coded with a CIIP method, at least one message related to motion vector difference may be indicated to refine motion information of an inter-prediction used in the CIIP method. For example, for a block coded with CIIP mode, at least one message related to motion vector difference may be signaled to refine the motion information of the inter-prediction used in CIIP.

[00137] In some embodiments, the at least one message related to motion vector difference may be coded in a same way as a motion vector difference coded for MMVD method. In some embodiments, the CIIP method and a MMVD method may be used together for the target block.

In some embodiments, an inter-prediction in the CIIP method may be generated based on the MMVD method.

[00138] In some embodiments, an indication of whether to and/or how to apply the coding tool may be indicated at one of the followings: sequence level, group of pictures level, picture level, slice level, or tile group level. In some embodiments, an indication of whether to and/or how to apply the coding tool may be indicated in one of the following: a sequence header, a picture header, a sequence parameter set (SPS), a video parameter set (VPS), a dependency parameter set (DPS), a decoding capability information (DCI), a picture parameter set (PPS), an adaptation parameter sets (APS), a slice header, or a tile group header.

[00139] In some embodiments, an indication of whether to and/or how to apply the coding tool may be included in one of the following: a prediction block (PB), a transform block (TB), a coding block (CB), a prediction unit (PU), a transform unit (TU), a coding unit (CU), a virtual pipeline data unit (VPDU), a coding tree unit (CTU), a CTU row, a slice, a tile, a sub-picture, or a region containing more than one sample or pixel.

[00140] In some embodiments, whether to and/or how to apply the coding tool may be determined based on coded information of the target block. The coded information may include at least one of: a block size, a colour format, a single and/or dual tree partitioning, a colour component, a slice type, or a picture type.

[00141] 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, whether a coding method is applied to a hypothesis of the video is determined based on coding information associated with the target block, and a bitstream of the target block is generated based on the determining.

[00142] In some embodiments, whether a coding method is applied to a hypothesis of a target block of the video is determined based on coding information associated with the target block. A bitstream of the target block is generated based on the determining, and the bitstream is stored in a non-transitory computer-readable recording medium.

[00143] Embodiments of the present disclosure can be implemented separately. Alternatively, embodiments of the present disclosure can be implemented in any proper combinations. 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.

[00144] Clause 1. A method of video processing, comprising: determining, during a conversion between a target block of a video and a bitstream of the target block, a target weight table from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; and performing the conversion based on the target weight table.

[00145] Clause 2. The method of clause 1, wherein the target weight table used for the hypothesis of the target block is indicated in the bitstream.

[00146] Clause 3. The method of clause 1, wherein determining the target weight table comprises: determining the target weight table based on at least one: a first prediction method of a base hypothesis of the target block, a second prediction method of an additional hypothesis of the target block, a width of a coded sequence associated with the target block, a height of the coded sequence associated with the target block, a width of the target block, or a height of the target block.

[00147] Clause 4. The method of clause 1, wherein a dimension of the target weight table is based on a dimension of the target block.

[00148] Clause 5. The method of clause 1, wherein the conversion includes encoding the target block into the bitstream.

[00149] Clause 6. The method of clause 1, wherein the conversion includes decoding the target block from the bitstream.

[00150] Clause 7. The method of any of clauses 1-6, wherein an indication of whether to and/or how to determine the target weight table is indicated at one of the followings: sequence level, group of pictures level, picture level, slice level, or tile group level.

[00151] Clause 8. The method of any of clauses 1-6, wherein an indication of whether to and/or how to determine the target weight table is indicated in one of the following: a sequence header, a picture header, a sequence parameter set (SPS), a video parameter set (VPS), a dependency parameter set (DPS), a decoding capability information (DCI), a picture parameter set (PPS), an adaptation parameter sets (APS), a slice header, or a tile group header.

[00152] Clause 9. The method of any of clauses 1-6, wherein an indication of whether to and/or how to determine the target weight table is included in one of the following: a prediction block (PB), a transform block (TB), a coding block (CB), a prediction unit (PU), a transform

unit (TU), a coding unit (CU), a virtual pipeline data unit (VPDU), a coding tree unit (CTU), a CTU row, a slice, a tile, a sub-picture, or a region containing more than one sample or pixel.

[00153] Clause 10. The method of any of clauses 1-6, further comprising: determining, based on coded information of the target block, whether to and/or how to determine the target weight table, the coded information including at least one of: a block size, a colour format, a single and/or dual tree partitioning, a colour component, a slice type, or a picture type.

[00154] Clause 11. A method of video processing, comprising: determining, during a conversion between a target block of a video and a bitstream of the target block, whether a coding method is applied to a hypothesis of the target block based on coding information associated with the target block; and performing the conversion based on the determining.

[00155] Clause 12. The method of clause 11, wherein if the target block is allowed to be coded as a base hypothesis, multiple hypothesis information associated the target block is indicated.

[00156] Clause 13. The method of clause 12, wherein if the target block is coded with the coding method but does not satisfy a condition, multiple hypothesis information associated the target block is not indicated.

[00157] Clause 14. The method of clause 11, wherein whether the target block applied with the coding method is allowed to be as a hypothesis depends on a further coding method of a neighbor block.

[00158] Clause 15. The method of clause 14, wherein if the neighbor block at a specified position is applied with the other coding method, the target block applied with the coding method is allowed to be as a base hypothesis.

[00159] Clause 16. The method of clause 15, wherein if the neighbor block at a specified position is applied with the other coding method, an additional hypothesis associated with a base hypothesis is allowed to be applied with the coding method.

[00160] Clause 17. The method of clause 14, wherein if a specified number of neighbor blocks are applied with the other coding method, the target block applied with the coding method is allowed to be as a base hypothesis.

[00161] Clause 18. The method of clause 17, wherein if the specified number of neighbor blocks are applied with the other coding method, an additional hypothesis associated with a base hypothesis is allowed to be applied with the coding method.

[00162] Clause 19. The method of clause 14, wherein if a specified number of neighbor blocks at a specified position are applied with the other coding method, the target block applied with the coding method is allowed to be as a base hypothesis.

[00163] Clause 20. The method of clause 17, wherein if the specified number of neighbor blocks at the specified position are applied with the other coding method, an additional hypothesis associated with a base hypothesis is allowed to be applied with the coding method.

[00164] Clause 21. The method of clause 11, wherein an additional hypothesis is allowed to be coded with the coding method depends on a further coding method of a base hypothesis of the additional hypothesis.

[00165] Clause 22. The method of clause 21, wherein if the base hypothesis is applied with the coding method, the additional hypothesis is applied with the further coding method.

[00166] Clause 23. The method of clause 11, wherein for a prediction method is not allowed to an additional hypothesis, a syntax element related to the prediction method is not indicated.

[00167] Clause 24. The method of clause 11, wherein the coding method comprises one of: an intra prediction method, an intra planar prediction method, an intra direct current (DC) prediction method, a position dependent intra prediction combination (PDPC) method, an intra angular prediction method, a block-based differential pulse-code modulation (BDPCM) method, a matrix weighted intra prediction (MIP) method, a multiple reference line (MRL) method, an intra sub-partitions (ISP) method, a linear model (LM) method, an intra block copy (IBC) method, an inter prediction method, an affine method, an affine merge method, a subblock-based temporal motion vector prediction (sbTMVP) method, a symmetric motion vector difference (SMVD) method, a combined inter and intra prediction (CIIP) method, a CIIP PDPC method, a geometric partitioning mode (GPM) method, a temporal motion (TM) method, or a decoder side motion vector refinement (DMVR) method.

[00168] Clause 25. The method of clause 11, wherein multiple hypothesis information is indicated, if the target block is coded with at least one of the followings: an intra prediction method, a single tree, an IBC mode, a BDPCM method, a MIP method, a MRL method, an ISP method, an intra DC prediction method, an intra planar prediction method, an intra angular

prediction method, a PDPC method, a LM method, a derived mode (DM) method, a CIIP method, a CIIP method but not CIIP PDPC method, a CIIP PDPC method, a GPM method, a template matching based refinement method, or a bilateral matching based refinement method.

[00169] Clause 26. The method of clause 25, wherein the target block is coded with the CIIP PDPC method, multiple hypothesis information is indicated before a CIIP PDPC flag.

[00170] Clause 27. The method of clause 25, wherein the target block is coded with the CIIP PDPC method, multiple hypothesis information is indicated before a merge index.

[00171] Clause 28. The method of clause 25, wherein the target block is coded with the CIIP PDPC method, multiple hypothesis information is indicated a merge data syntax structure.

[00172] Clause 29. The method of clause 25, wherein whether a CIIP PDPC flag is indicated depends on whether a multiple hypothesis prediction is used to the target block.

[00173] Clause 30. The method of clause 29, if the multiple hypothesis prediction is not applied to the target block, a CIIP PDPC flag is indicated.

[00174] Clause 31. The method of clause 29, if the multiple hypothesis prediction is applied to the target block, a CIIP PDPC flag is not indicated and set to a predefined value.

[00175] Clause 32. The method of clause 11, wherein whether multiple hypothesis information is indicated depends on whether at least one condition is satisfied, or wherein whether multiple hypothesis information is indicated depends on whether the at least one condition is not satisfied.

[00176] Clause 33. The method of clause 11, wherein if multiple hypothesis information is indicated and a base hypothesis is coded with a prediction mode associated with a prediction method, at least one of the followings is conformed: the multiple hypothesis information is present for the specified prediction modes but not present for other prediction modes, or the multiple hypothesis information is present for the specified prediction mode but not present for other prediction methods.

[00177] Clause 34. The method of clause 11, wherein if a subblock-block based coding method is applied, multiple hypothesis information is applied to the target block.

[00178] Clause 35. The method of clause 34, wherein for the target block coded with an affine inter method, the multiple hypothesis prediction is applied regardless of a bi-prediction with coding unit level weight (BCW) index being equal to a default value or not.

[00179] Clause 36. The method of clause 34, wherein for the target block coded with a sub-block based DMVR method, the multiple hypothesis prediction is applied.

[00180] Clause 37. The method of clause 34, wherein for the target block coded with a sub-block based bi-directional optical flow (BDOF) method, the multiple hypothesis prediction is applied.

[00181] Clause 38. The method of clause 34, wherein for the target block coded with a sub-block based prediction refinement with optical flow (PROF) method, the multiple hypothesis prediction is applied.

[00182] Clause 39. The method of clause 11, wherein if an advanced motion vector prediction (AMVP) based coding method is applied, multiple hypothesis information is applied to the target block.

[00183] Clause 40. The method of clause 39, wherein for the target block with a uni-directional predicted AMVP, the multiple hypothesis information is applied to the target block.

[00184] Clause 41. The method of clause 39, wherein for the target block with a bi-directional predicted AMVP, the multiple hypothesis information is applied to the target block regardless of a bi-prediction with coding unit level weight (BCW) index being equal to a default value or not.

[00185] Clause 42. The method of clause 39, wherein for the target block with a bi-directional predicted AMVP, the multiple hypothesis information is applied to the target block even though a BCW index is equal to the default value.

[00186] Clause 43. The method of clause 39, wherein the target block is a non-affine AMVP block, or Wherein the target block is an affine AMVP block.

[00187] Clause 44. The method of clause 11, wherein if a multiple hypothesis prediction is applied to the target block, at least one message corresponding to multiple hypothesis information is indicated associated with the target block, wherein the multiple hypothesis information comprises coding information of the target block.

[00188] Clause 45. The method of clause 44, wherein the coding information comprises whether an additional hypothesis is coded with an inter prediction method or an intra prediction method.

[00189] Clause 46. The method of clause 45, wherein if the additional hypothesis is coded with the intra prediction mode, whether the additional hypothesis is coded with a first prediction method is indicated.

[00190] Clause 47. The method of clause 45, wherein the additional hypothesis is coded with the first prediction method under a condition that a base hypothesis is not coded by a second prediction method.

[00191] Clause 48. The method of clause 45, wherein the first prediction method is mandatorily used for an intra coded additional hypothesis, whether the additional hypothesis is coded with the first prediction method is not indicated.

[00192] Clause 49. The method of clause 48, wherein a pre-defined intra prediction method is mandatorily used.

[00193] Clause 50. The method of clause 48, wherein whether a mandatory prediction method is used for an intra coded additional hypothesis depends on a size of the target block.

[00194] Clause 51. The method of clause 46, wherein if an additional hypothesis is not allowed to be coded by the first prediction method, whether the additional hypothesis is coded with the first prediction method is not indicated.

[00195] Clause 52. The method of any of clauses 46-51, wherein the first prediction method is a MIP method.

[00196] Clause 53. The method of clause 52, wherein a MIP transpose flag and/or a MIP mode index is indicated.

[00197] Clause 54. The method of clause 52, wherein a default MIP transpose flag and/or a default MIP mode index is used, and additional MIP information is not indicated.

[00198] Clause 55. The method of any of clauses 46-51, wherein the first prediction method is a MRL method.

[00199] Clause 56. The method of clause 55, wherein a reference index of MRL is indicated.

[00200] Clause 57. The method of clause 55, wherein a default MRL index is used, and additional MIP information is not indicated.

[00201] Clause 58. The method of any of clauses 46-51, wherein the first prediction method is an ISP method.

[00202] Clause 59. The method of clause 58, wherein a split direction of ISP is indicated.

[00203] Clause 60. The method of clause 58, wherein a default ISP split direction is used, and additional MIP information is not indicated.

[00204] Clause 61. The method of any of clauses 46-51, wherein the first prediction method is an intra DC prediction method.

[00205] Clause 62. The method of any of clauses 46-51, wherein the first prediction method is an intra planar prediction method.

[00206] Clause 63. The method of any of clauses 46-51, wherein the first prediction method is a regular intra prediction method.

[00207] Clause 64. The method of clause 63, wherein an intra prediction mode index is indicated.

[00208] Clause 65. The method of clause 63, wherein a default intra mode is used, and additional MIP information is not indicated.

[00209] Clause 66. The method of any of clauses 46-51, wherein the first prediction method is an intra DM prediction method.

[00210] Clause 67. The method of any of clauses 46-51, wherein the first prediction method is an intra LM prediction method.

[00211] Clause 68. The method of clause 67, wherein a LM prediction method is indicated.

[00212] Clause 69. The method of clause 67, wherein a default LM method is used, and additional LM information is indicated.

[00213] Clause 70. The method of any of clauses 46-51, wherein the first prediction method is a PDPC method.

[00214] Clause 71. The method of clause 70, wherein a PDPC predictor generated from a neighbor sample is used as an additional hypothesis.

[00215] Clause 72. The method of clause 70, wherein a PDPC predictor associated with an intra prediction mode is as an additional hypothesis, whether to apply PDPC is determined on an intra prediction mode index, and whether an additional intra code hypothesis uses PDPC is not indicated.

[00216] Clause 73. The method of any of clauses 46-51, wherein the first prediction method is a BDPCM method.

[00217] Clause 74. The method of clause 73, wherein a BDPCM prediction direction is indicated.

[00218] Clause 75. The method of clause 73, wherein a default BDPCM prediction direction is used, and additional BDPCM information is not indicated.

[00219] Clause 76. The method of clause 11, wherein an additional hypothesis is coded with an inter prediction method.

[00220] Clause 77. The method of clause 76, wherein whether the additional hypothesis is coded with a third prediction method is indicated.

[00221] Clause 78. The method of clause 77, wherein the additional hypothesis is coded with the third prediction method under a condition that a base hypothesis is not coded by a fourth prediction method.

[00222] Clause 79. The method of clause 77, wherein the third prediction method is mandatorily used for an inter coded additional hypothesis, whether the additional hypothesis is coded with the third prediction method is not indicated.

[00223] Clause 80. The method of clause 79, wherein a pre-defined inter prediction method is mandatorily used.

[00224] Clause 81. The method of clause 80, wherein whether a mandatory prediction method is used for an inter coded additional hypothesis depends on a size of the target block.

[00225] Clause 82. The method of clause 77, wherein if an additional hypothesis is not allowed to be coded by the third prediction method, whether the additional hypothesis is coded with the third prediction method is not indicated.

[00226] Clause 83. The method of any of clauses 77-82, wherein the third prediction method is a CIIP method.

[00227] Clause 84. The method of any of clauses 77-82, wherein the third prediction method is a GPM method.

[00228] Clause 85. The method of any of clauses 77-82, wherein the third prediction method is a MMVD method.

[00229] Clause 86. The method of any of clauses 77-82, wherein the third prediction method is a SMVD method.

[00230] Clause 87. The method of any of clauses 77-82, wherein the third prediction method is an affine method.

[00231] Clause 88. The method of any of clauses 77-82, wherein the third prediction method is a sbTMVP method.

[00232] Clause 89. The method of any of clauses 77-82, wherein the third prediction method is a PROF method.

[00233] Clause 90. The method of any of clauses 77-82, wherein the third prediction method is a BDOF method.

[00234] Clause 91. The method of any of clauses 77-82, wherein the third prediction method is an overlapped block based motion compensation (OBMC) method.

[00235] Clause 92. The method of any of clauses 77-82, wherein the third prediction method is a merge method.

[00236] Clause 93. The method of clause 92, wherein an additional hypothesis is coded by the merge method, and a merge index associated with the additional hypothesis is different from a merge index associated with a base hypothesis.

[00237] Clause 94. The method of any of clauses 77-82, wherein the third prediction method is an AMVP method.

[00238] Clause 95. The method of clause 94, wherein an additional hypothesis is coded by the AMVP method, and an AMVP index associated with the additional hypothesis is different from an AMVP index associated with a base hypothesis.

[00239] Clause 96. The method of any of clauses 77-82, wherein the third prediction method is a template matching based refinement.

[00240] Clause 97. The method of any of clauses 77-82, wherein the third prediction method is a bilateral matching based refinement.

[00241] Clause 98. The method of any of clauses 77-82, wherein the third prediction method is a PDPC.

[00242] Clause 99. The method of clause 98, wherein a PDPC predictor generated from a neighbor sample is used as an additional hypothesis.

[00243] Clause 100. The method of clause 11, wherein for the target block coded with a CIIP method, at least one message related to motion vector difference is indicated to refine motion information of an inter-prediction used in the CIIP method.

[00244] Clause 101. The method of clause 100, wherein the at least one message related to motion vector difference is coded in a same way as a motion vector difference coded for MMVD method.

[00245] Clause 102. The method of clause 100, wherein the CIIP method and a MMVD method are used together for the target block, wherein an inter-prediction in the CIIP method is generated based on the MMVD method.

[00246] Clause 103. The method of clause 11, wherein the conversion includes encoding the target block into the bitstream.

[00247] Clause 104. The method of clause 11, wherein the conversion includes decoding the target block from the bitstream.

[00248] Clause 105. The method of any of clauses 11-104, wherein an indication of whether to and/or how to apply the coding tool is indicated at one of the followings: sequence level, group of pictures level, picture level, slice level, or tile group level.

[00249] Clause 106. The method of any of clauses 11-104, wherein an indication of whether to and/or how to apply the coding tool is indicated in one of the following: a sequence header, a picture header, a sequence parameter set (SPS), a video parameter set (VPS), a dependency parameter set (DPS), a decoding capability information (DCI), a picture parameter set (PPS), an adaptation parameter sets (APS), a slice header, or a tile group header.

[00250] Clause 107. The method of any of clauses 11-104, wherein an indication of whether to and/or how to apply the coding tool is included in one of the following: a prediction block (PB), a transform block (TB), a coding block (CB), a prediction unit (PU), a transform unit (TU), a coding unit (CU), a virtual pipeline data unit (VPDU), a coding tree unit (CTU), a CTU row, a slice, a tile, a sub-picture, or a region containing more than one sample or pixel.

[00251] Clause 108. The method of any of clauses 11-104, further comprising: determining, based on coded information of the target block, whether and/or how the coding tool is applied,

the coded information including at least one of: a block size, a colour format, a single and/or dual tree partitioning, a colour component, a slice type, or a picture type.

[00252] Clause 109. 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 1-10 or any of clauses 11-104.

[00253] Clause 110. A non-transitory computer-readable storage medium storing instructions that cause a processor to perform a method in accordance with any of clauses 1-10 or any of clauses 11-104.

[00254] Clause 111. 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: determining a target weight table applied to a target block of the video from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; and generating a bitstream of the target block based on the determining.

[00255] Clause 112. A method for storing bitstream of a video, comprising: determining a target weight table applied to a target block of the video from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; generating a bitstream of the target block based on the determining; and storing the bitstream in a non-transitory computer-readable recording medium.

[00256] Clause 113. 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: determining whether a coding method is applied to a hypothesis of the video based on coding information associated with the target block; and generating a bitstream of the target block based on the determining.

[00257] Clause 114. A method for storing bitstream of a video, comprising: determining whether a coding method is applied to a hypothesis of a target block of the video based on coding information associated with the target block; generating a bitstream of the target block based on the determining; and storing the bitstream in a non-transitory computer-readable recording medium.

Example Device

[00258] Fig. 25 illustrates a block diagram of a computing device 2500 in which various embodiments of the present disclosure can be implemented. The computing device 2500 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).

[00259] It would be appreciated that the computing device 2500 shown in Fig. 25 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.

[00260] As shown in Fig. 25, the computing device 2500 includes a general-purpose computing device 2500. The computing device 2500 may at least comprise one or more processors or processing units 2510, a memory 2520, a storage unit 2530, one or more communication units 2540, one or more input devices 2550, and one or more output devices 2560.

[00261] In some embodiments, the computing device 2500 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 2500 can support any type of interface to a user (such as “wearable” circuitry and the like).

[00262] The processing unit 2510 may be a physical or virtual processor and can implement various processes based on programs stored in the memory 2520. 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 2500. The processing unit 2510 may also be referred to as a central processing unit (CPU), a microprocessor, a controller or a microcontroller.

[00263] The computing device 2500 typically includes various computer storage medium. Such medium can be any medium accessible by the computing device 2500, including, but not limited to, volatile and non-volatile medium, or detachable and non-detachable medium. The memory 2520 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 Programmable Read-Only Memory (EEPROM), or a flash memory), or any combination thereof. The storage unit 2530 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 2500.

[00264] The computing device 2500 may further include additional detachable/non-detachable, volatile/non-volatile memory medium. Although not shown in Fig. 25, 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 medium interfaces.

[00265] The communication unit 2540 communicates with a further computing device via the communication medium. In addition, the functions of the components in the computing device 2500 can be implemented by a single computing cluster or multiple computing machines that can communicate via communication connections. Therefore, the computing device 2500 can operate in a networked environment using a logical connection with one or more other servers, networked personal computers (PCs) or further general network nodes.

[00266] The input device 2550 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 2560 may be one or more of a variety of output devices, such as a display, loudspeaker, printer, and the like. By means of the communication unit 2540, the computing device 2500 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 2500, or any devices (such as a network card, a modem and the like) enabling the computing device 2500 to communicate with one or more other computing devices, if required. Such communication can be performed via input/output (I/O) interfaces (not shown).

[00267] In some embodiments, instead of being integrated in a single device, some or all components of the computing device 2500 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.

[00268] The computing device 2500 may be used to implement video encoding/decoding in embodiments of the present disclosure. The memory 2520 may include one or more video coding modules 2525 having one or more program instructions. These modules are accessible and executable by the processing unit 2510 to perform the functionalities of the various embodiments described herein.

[00269] In the example embodiments of performing video encoding, the input device 2550 may receive video data as an input 2570 to be encoded. The video data may be processed, for example, by the video coding module 2525, to generate an encoded bitstream. The encoded bitstream may be provided via the output device 2560 as an output 2580.

[00270] In the example embodiments of performing video decoding, the input device 2550 may receive an encoded bitstream as the input 2570. The encoded bitstream may be processed, for example, by the video coding module 2525, to generate decoded video data. The decoded video data may be provided via the output device 2560 as the output 2580.

[00271] 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 of video processing, comprising:
determining, during a conversion between a target block of a video and a bitstream of the target block, a target weight table from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; and
performing the conversion based on the target weight table.
2. The method of claim 1, wherein the target weight table used for the hypothesis of the target block is indicated in the bitstream.
3. The method of claim 1, wherein determining the target weight table comprises:
determining the target weight table based on at least one:
a first prediction method of a base hypothesis of the target block,
a second prediction method of an additional hypothesis of the target block,
a width of a coded sequence associated with the target block,
a height of the coded sequence associated with the target block,
a width of the target block, or
a height of the target block.
4. The method of claim 1, wherein a dimension of the target weight table is based on a dimension of the target block.
5. The method of claim 1, wherein the conversion includes encoding the target block into the bitstream.

6. The method of claim 1, wherein the conversion includes decoding the target block from the bitstream.

7. The method of any of claims 1-6, wherein an indication of whether to and/or how to determine the target weight table is indicated at one of the followings:

- sequence level,
- group of pictures level,
- picture level,
- slice level, or
- tile group level.

8. The method of any of claims 1-6, wherein an indication of whether to and/or how to determine the target weight table is indicated in one of the following:

- a sequence header,
- a picture header,
- a sequence parameter set (SPS),
- a video parameter set (VPS),
- a dependency parameter set (DPS),
- a decoding capability information (DCI),
- a picture parameter set (PPS),
- an adaptation parameter sets (APS),
- a slice header, or
- a tile group header.

9. The method of any of claims 1-6, wherein an indication of whether to and/or how to determine the target weight table is included in one of the following:

a prediction block (PB),
a transform block (TB),
a coding block (CB),
a prediction unit (PU),
a transform unit (TU),
a coding unit (CU),
a virtual pipeline data unit (VPDU),
a coding tree unit (CTU),
a CTU row,
a slice,
a tile,
a sub-picture, or
a region containing more than one sample or pixel.

10. The method of any of claims 1-6, further comprising:

determining, based on coded information of the target block, whether to and/or how to determine the target weight table, the coded information including at least one of:

a block size,
a colour format,
a single and/or dual tree partitioning,
a colour component,
a slice type, or
a picture type.

11. A method of video processing, comprising:

determining, during a conversion between a target block of a video and a bitstream of the target block, whether a coding method is applied to a hypothesis of the target block based on coding information associated with the target block; and

performing the conversion based on the determining.

12. The method of claim 11, wherein if the target block is allowed to be coded as a base hypothesis, multiple hypothesis information associated the target block is indicated.

13. The method of claim 12, wherein if the target block is coded with the coding method but does not satisfy a condition, multiple hypothesis information associated the target block is not indicated.

14. The method of claim 11, wherein whether the target block applied with the coding method is allowed to be as a hypothesis depends on a further coding method of a neighbor block.

15. The method of claim 14, wherein if the neighbor block at a specified position is applied with the other coding method, the target block applied with the coding method is allowed to be as a base hypothesis.

16. The method of claim 15, wherein if the neighbor block at a specified position is applied with the other coding method, an additional hypothesis associated with a base hypothesis is allowed to be applied with the coding method.

17. The method of claim 14, wherein if a specified number of neighbor blocks are applied with the other coding method, the target block applied with the coding method is allowed to be as a base hypothesis.

18. The method of claim 17, wherein if the specified number of neighbor blocks are applied with the other coding method, an additional hypothesis associated with a base hypothesis is allowed to be applied with the coding method.

19. The method of claim 14, wherein if a specified number of neighbor blocks at a specified position are applied with the other coding method, the target block applied with the coding method is allowed to be as a base hypothesis.

20. The method of claim 17, wherein if the specified number of neighbor blocks at the specified position are applied with the other coding method, an additional hypothesis associated with a base hypothesis is allowed to be applied with the coding method.

21. The method of claim 11, wherein an additional hypothesis is allowed to be coded with the coding method depends on a further coding method of a base hypothesis of the additional hypothesis.

22. The method of claim 21, wherein if the base hypothesis is applied with the coding method, the additional hypothesis is applied with the further coding method.

23. The method of claim 11, wherein for a prediction method is not allowed to an additional hypothesis, a syntax element related to the prediction method is not indicated.

24. The method of claim 11, wherein the coding method comprises one of:

an intra prediction method,

an intra planar prediction method,

an intra direct current (DC) prediction method,

a position dependent intra prediction combination (PDPC) method,

an intra angular prediction method,
a block-based differential pulse-code modulation (BDPCM) method,
a matrix weighted intra prediction (MIP) method,
a multiple reference line (MRL) method,
an intra sub-partitions (ISP) method,
a linear model (LM) method,
an intra block copy (IBC) method,
an inter prediction method,
an affine method,
an affine merge method,
a subblock-based temporal motion vector prediction (sbTMVP) method,
a symmetric motion vector difference (SMVD) method,
a combined inter and intra prediction (CIIP) method,
a CIIP PDPC method,
a geometric partitioning mode (GPM) method,
an overlapped block based motion compensation (OBMC) method,
a temporal motion (TM) method , or
a decoder side motion vector refinement (DMVR) method.

25. The method of claim 11, wherein multiple hypothesis information is indicated, if the target block is coded with at least one of the followings:

an intra prediction method,
a single tree,
an IBC mode,
a BDPCM method,
a MIP method,

a MRL method,
an ISP method,
an intra DC prediction method,
an intra planar prediction method,
an intra angular prediction method,
a PDPC method,
a LM method,
a derived mode (DM) method,
a CIIP method,
a CIIP method but not CIIP PDPC method,
a CIIP PDPC method,
a GPM method,
an overlapped block based motion compensation (OBMC) method,
a template matching based refinement method, or
a bilateral matching based refinement method.

26. The method of claim 25, wherein the target block is coded with the CIIP PDPC method, multiple hypothesis information is indicated before a CIIP PDPC flag.

27. The method of claim 25, wherein the target block is coded with the CIIP PDPC method, multiple hypothesis information is indicated before a merge index.

28. The method of claim 25, wherein the target block is coded with the CIIP PDPC method, multiple hypothesis information is indicated a merge data syntax structure.

29. The method of claim 25, wherein whether a CIIP PDPC flag is indicated depends on whether a multiple hypothesis prediction is used to the target block.

30. The method of claim 29, if the multiple hypothesis prediction is not applied to the target block, a CIIP PDPC flag is indicated.

31. The method of claim 29, if the multiple hypothesis prediction is applied to the target block, a CIIP PDPC flag is not indicated and set to a predefined value.

32. The method of claim 11, wherein whether multiple hypothesis information is indicated depends on whether at least one condition is satisfied, or

wherein whether multiple hypothesis information is indicated depends on whether the at least one condition is not satisfied.

33. The method of claim 11, wherein if multiple hypothesis information is indicated and a base hypothesis is coded with a prediction mode associated with a prediction method, at least one of the followings is conformed:

the multiple hypothesis information is present for the specified prediction modes but not present for other prediction modes, or

the multiple hypothesis information is present for the specified prediction mode but not present for other prediction methods.

34. The method of claim 11, wherein if a subblock-block based coding method is applied, multiple hypothesis information is applied to the target block.

35. The method of claim 34, wherein for the target block coded with an affine inter method, the multiple hypothesis prediction is applied regardless of a bi-prediction with coding unit level weight (BCW) index being equal to a default value or not.

36. The method of claim 34, wherein for the target block coded with a subblock based DMVR method, the multiple hypothesis prediction is applied.

37. The method of claim 34, wherein for the target block coded with a subblock based bi-directional optical flow (BDOF) method, the multiple hypothesis prediction is applied.

38. The method of claim 34, wherein for the target block coded with a subblock based prediction refinement with optical flow (PROF) method, the multiple hypothesis prediction is applied.

39. The method of claim 11, wherein if an advanced motion vector prediction (AMVP) based coding method is applied, multiple hypothesis information is applied to the target block.

40. The method of claim 39, wherein for the target block with a uni-directional predicted AMVP, the multiple hypothesis information is applied to the target block.

41. The method of claim 39, wherein for the target block with a bi-directional predicted AMVP, the multiple hypothesis information is applied to the target block regardless of a bi-prediction with coding unit level weight (BCW) index being equal to a default value or not.

42. The method of claim 39, wherein for the target block with a bi-directional predicted AMVP, the multiple hypothesis information is applied to the target block even though a BCW index is equal to the default value.

43. The method of claim 39, wherein the target block is a non-affine AMVP block, or wherein the target block is an affine AMVP block.

44. The method of claim 11, wherein if a multiple hypothesis prediction is applied to the target block, at least one message corresponding to multiple hypothesis information is indicated associated with the target block, wherein the multiple hypothesis information comprises coding information of the target block.

45. The method of claim 44, wherein the coding information comprises whether an additional hypothesis is coded with an inter prediction method or an intra prediction method.

46. The method of claim 45, wherein if the additional hypothesis is coded with the intra prediction mode, whether the additional hypothesis is coded with a first prediction method is indicated.

47. The method of claim 45, wherein the additional hypothesis is coded with the first prediction method under a condition that a base hypothesis is not coded by a second prediction method.

48. The method of claim 45, wherein the first prediction method is mandatorily used for an intra coded additional hypothesis, whether the additional hypothesis is coded with the first prediction method is not indicated.

49. The method of claim 48, wherein a pre-defined intra prediction method is mandatorily used.

50. The method of claim 48, wherein whether a mandatory prediction method is used for an intra coded additional hypothesis depends on a size of the target block.

51. The method of claim 46, wherein if an additional hypothesis is not allowed to be coded by the first prediction method, whether the additional hypothesis is coded with the first prediction method is not indicated.

52. The method of any of claims 46-51, wherein the first prediction method is a MIP method.

53. The method of claim 52, wherein a MIP transpose flag and/or a MIP mode index is indicated.

54. The method of claim 52, wherein a default MIP transpose flag and/or a default MIP mode index is used, and additional MIP information is not indicated.

55. The method of any of claims 46-51, wherein the first prediction method is a MRL method.

56. The method of claim 55, wherein a reference index of MRL is indicated.

57. The method of claim 55, wherein a default MRL index is used, and additional MIP information is not indicated.

58. The method of any of claims 46-51, wherein the first prediction method is an ISP method.

59. The method of claim 58, wherein a split direction of ISP is indicated.

60. The method of claim 58, wherein a default ISP split direction is used, and additional MIP information is not indicated.

61. The method of any of claims 46-51, wherein the first prediction method is an intra DC prediction method.

62. The method of any of claims 46-51, wherein the first prediction method is an intra planar prediction method.

63. The method of any of claims 46-51, wherein the first prediction method is a regular intra prediction method.

64. The method of claim 63, wherein an intra prediction mode index is indicated.

65. The method of claim 63, wherein a default intra mode is used, and additional MIP information is not indicated.

66. The method of any of claims 46-51, wherein the first prediction method is an intra DM prediction method.

67. The method of any of claims 46-51, wherein the first prediction method is an intra LM prediction method.

68. The method of claim 67, wherein a LM prediction method is indicated.

69. The method of claim 67, wherein a default LM method is used, and additional LM information is indicated.

70. The method of any of claims 46-51, wherein the first prediction method is a PDPC method.

71. The method of claim 70, wherein a PDPC predictor generated from a neighbor sample is used as an additional hypothesis.

72. The method of claim 70, wherein a PDPC predictor associated with an intra prediction mode is as an additional hypothesis, whether to apply PDPC is determined on an intra prediction mode index, and whether an additional intra code hypothesis uses PDPC is not indicated.

73. The method of any of claims 46-51, wherein the first prediction method is a BDPCM method.

74. The method of claim 73, wherein a BDPCM prediction direction is indicated.

75. The method of claim 73, wherein a default BDPCM prediction direction is used, and additional BDPCM information is not indicated.

76. The method of claim 11, wherein an additional hypothesis is coded with an inter prediction method.

77. The method of claim 76, wherein whether the additional hypothesis is coded with a third prediction method is indicated.

78. The method of claim 77, wherein the additional hypothesis is coded with the third prediction method under a condition that a base hypothesis is not coded by a fourth prediction method.

79. The method of claim 77, wherein the third prediction method is mandatorily used for an inter coded additional hypothesis, whether the additional hypothesis is coded with the third prediction method is not indicated.

80. The method of claim 79, wherein a pre-defined inter prediction method is mandatorily used.

81. The method of claim 80, wherein whether a mandatory prediction method is used for an inter coded additional hypothesis depends on a size of the target block.

82. The method of claim 77, wherein if an additional hypothesis is not allowed to be coded by the third prediction method, whether the additional hypothesis is coded with the third prediction method is not indicated.

83. The method of any of claims 77-82, wherein the third prediction method is a CIIP method.

84. The method of any of claims 77-82, wherein the third prediction method is a GPM method.

85. The method of any of claims 77-82, wherein the third prediction method is a MMVD method.

86. The method of any of claims 77-82, wherein the third prediction method is a SMVD method.

87. The method of any of claims 77-82, wherein the third prediction method is an affine method.

88. The method of any of claims 77-82, wherein the third prediction method is a sbTMVP method.

89. The method of any of claims 77-82, wherein the third prediction method is a PROF method.

90. The method of any of claims 77-82, wherein the third prediction method is a BDOF method.

91. The method of any of claims 77-82, wherein the third prediction method is an overlapped block based motion compensation (OBMC) method.

92. The method of any of claims 77-82, wherein the third prediction method is a merge method.

93. The method of claim 92, wherein an additional hypothesis is coded by the merge method, and a merge index associated with the additional hypothesis is different from a merge index associated with a base hypothesis.

94. The method of any of claims 77-82, wherein the third prediction method is an AMVP method.

95. The method of claim 94, wherein an additional hypothesis is coded by the AMVP method, and an AMVP index associated with the additional hypothesis is different from an AMVP index associated with a base hypothesis.

96. The method of any of claims 77-82, wherein the third prediction method is a template matching based refinement.

97. The method of any of claims 77-82, wherein the third prediction method is a bilateral matching based refinement.

98. The method of any of claims 77-82, wherein the third prediction method is a PDPC.

99. The method of claim 98, wherein a PDPC predictor generated from a neighbor sample is used as an additional hypothesis.

100. The method of claim 11, wherein for the target block coded with a CIIP method, at least one message related to motion vector difference is indicated to refine motion information of an inter-prediction used in the CIIP method.

101. The method of claim 100, wherein the at least one message related to motion vector difference is coded in a same way as a motion vector difference coded for MMVD method.

102. The method of claim 100, wherein the CIIP method and a MMVD method are used together for the target block, wherein an inter-prediction in the CIIP method is generated based on the MMVD method.

103. The method of claim 11, wherein the conversion includes encoding the target block into the bitstream.

104. The method of claim 11, wherein the conversion includes decoding the target block from the bitstream.

105. The method of any of claims 11-104, wherein an indication of whether to and/or how to apply the coding tool is indicated at one of the followings:

sequence level,
group of pictures level,
picture level,
slice level, or
tile group level.

106. The method of any of claims 11-104, wherein an indication of whether to and/or how to apply the coding tool is indicated in one of the following:

a sequence header,
a picture header,
a sequence parameter set (SPS),
a video parameter set (VPS),
a dependency parameter set (DPS),
a decoding capability information (DCI),

a picture parameter set (PPS),
an adaptation parameter sets (APS),
a slice header, or
a tile group header.

107. The method of any of claims 11-104, wherein an indication of whether to and/or how to apply the coding tool is included in one of the following:

a prediction block (PB),
a transform block (TB),
a coding block (CB),
a prediction unit (PU),
a transform unit (TU),
a coding unit (CU),
a virtual pipeline data unit (VPDU),
a coding tree unit (CTU),
a CTU row,
a slice,
a tile,
a sub-picture, or
a region containing more than one sample or pixel.

108. The method of any of claims 11-104, further comprising:
determining, based on coded information of the target block, whether and/or how the coding tool is applied, the coded information including at least one of:

a block size,
a colour format,

a single and/or dual tree partitioning,
a colour component,
a slice type, or
a picture type.

109. 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 claims 1-10 or any of claims 11-104.

110. A non-transitory computer-readable storage medium storing instructions that cause a processor to perform a method in accordance with any of claims 1-10 or any of claims 11-104.

111. 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:

determining a target weight table applied to a target block of the video from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block; and
generating a bitstream of the target block based on the determining.

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

determining a target weight table applied to a target block of the video from a plurality of weight tables for multi-hypothesis prediction, the target weight table being used for a hypothesis of the target block, the target block being a multiple hypothesis prediction block;
generating a bitstream of the target block based on the determining; and

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

113. 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:

determining whether a coding method is applied to a hypothesis of the video based on coding information associated with the target block; and
generating a bitstream of the target block based on the determining.

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

determining whether a coding method is applied to a hypothesis of a target block of the video based on coding information associated with the target block;
generating a bitstream of the target block based on the determining; and
storing the bitstream in a non-transitory computer-readable recording medium.

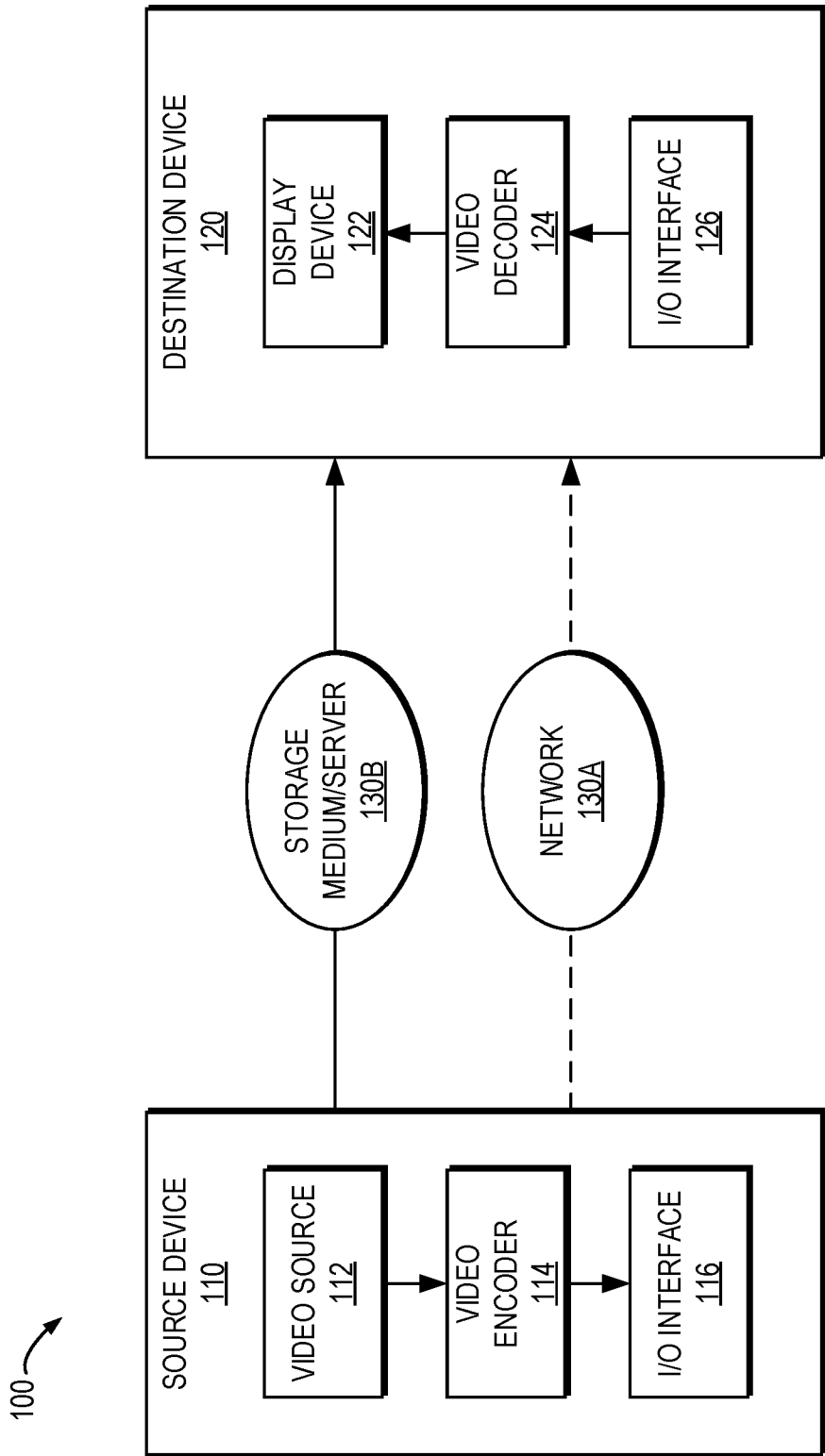


Fig. 1

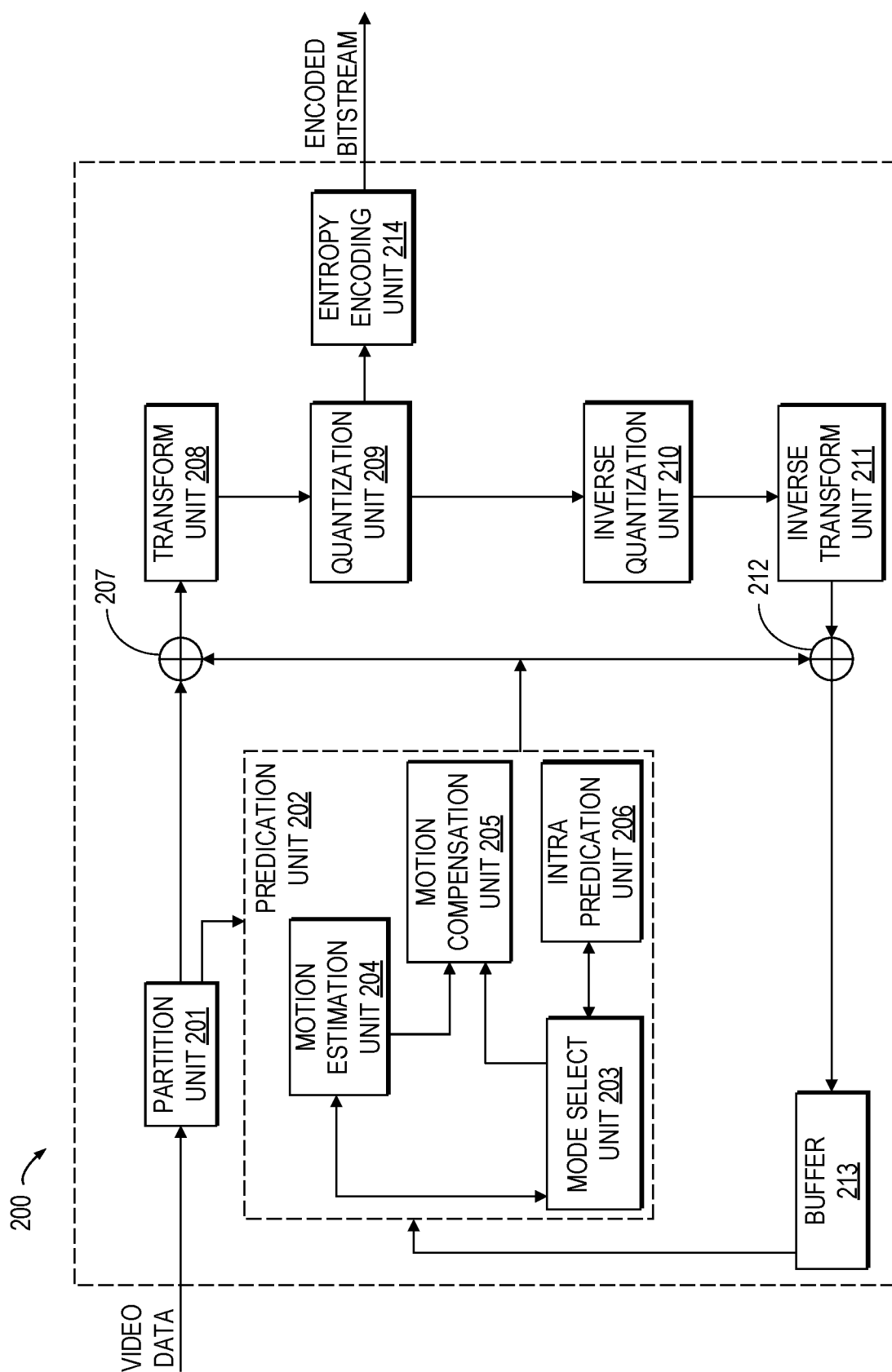


Fig. 2

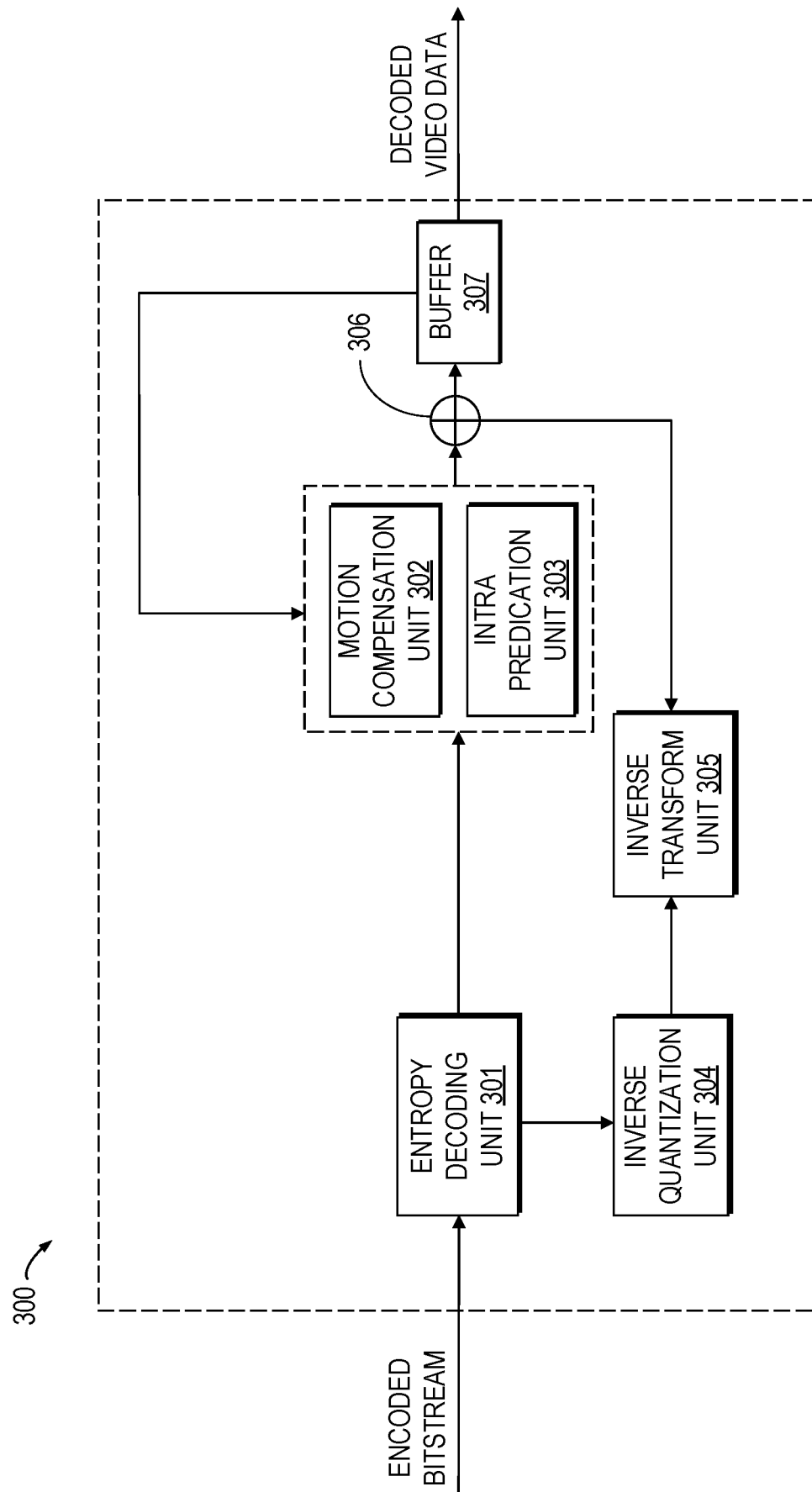


Fig. 3

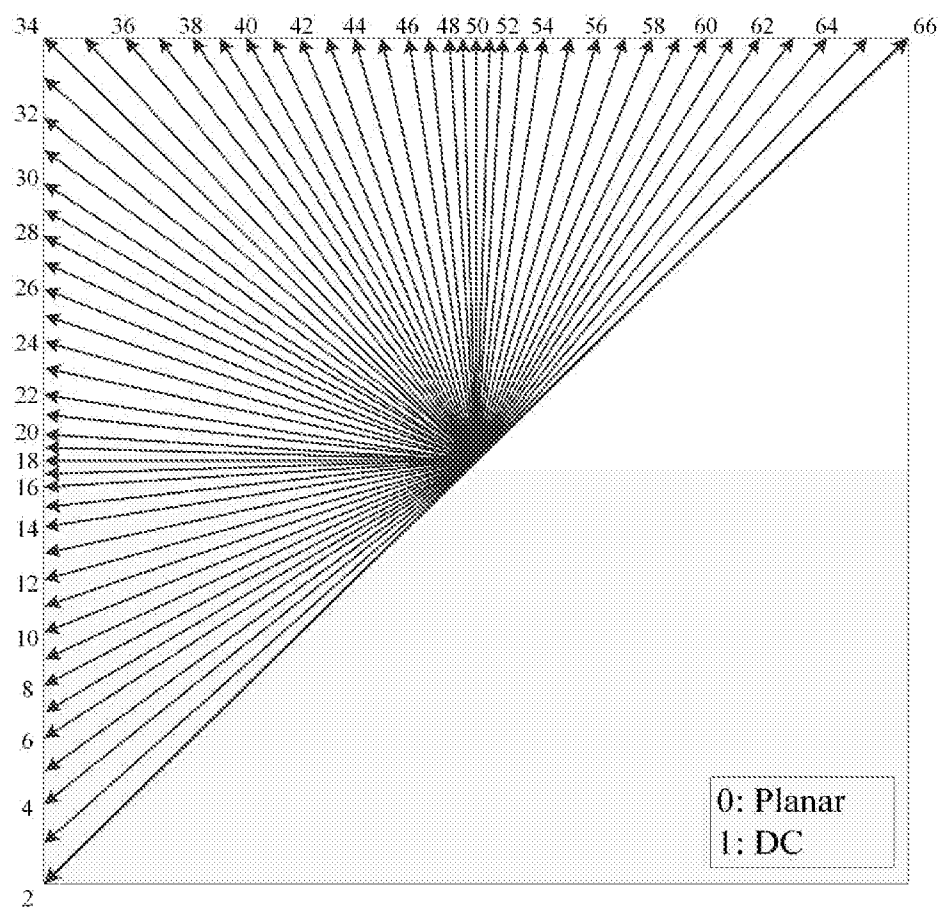


Fig. 4

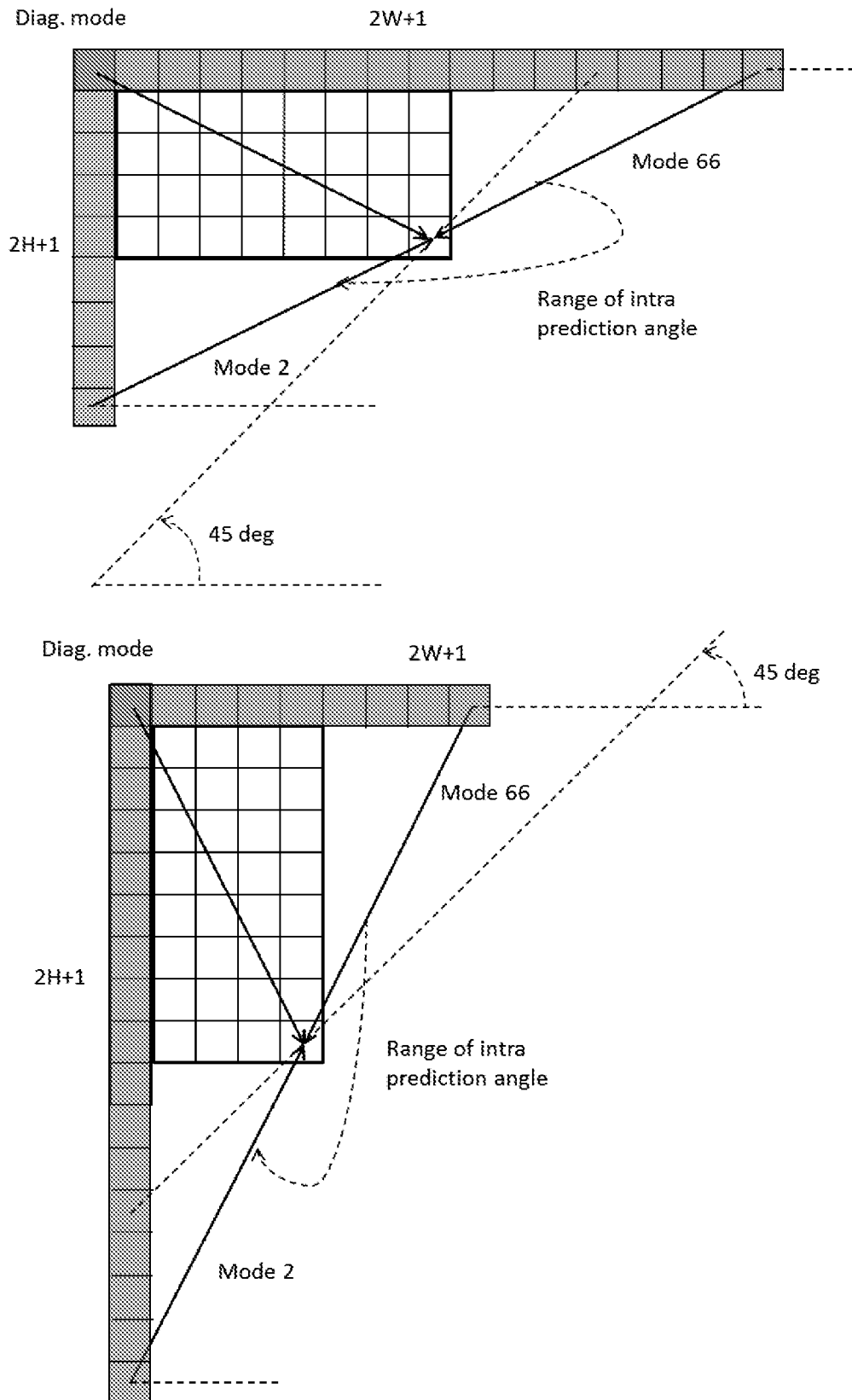


Fig. 5

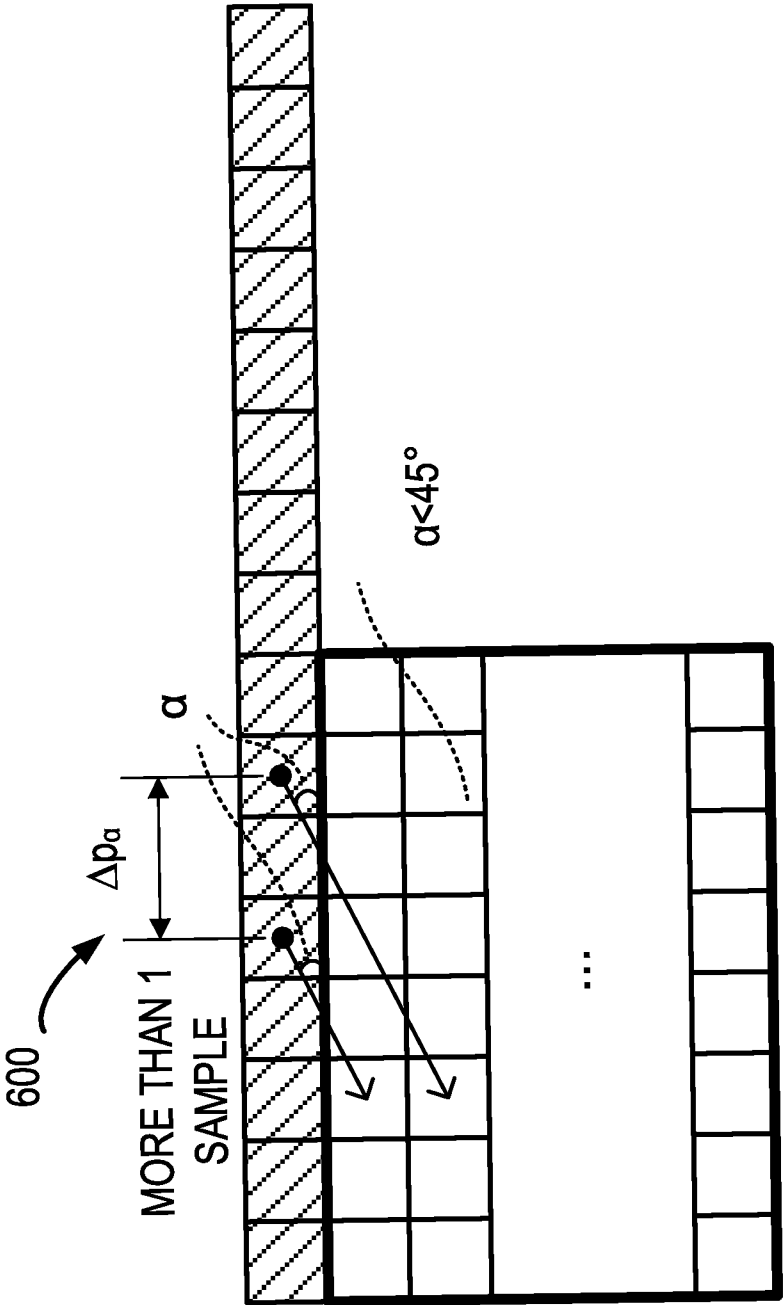


Fig. 6

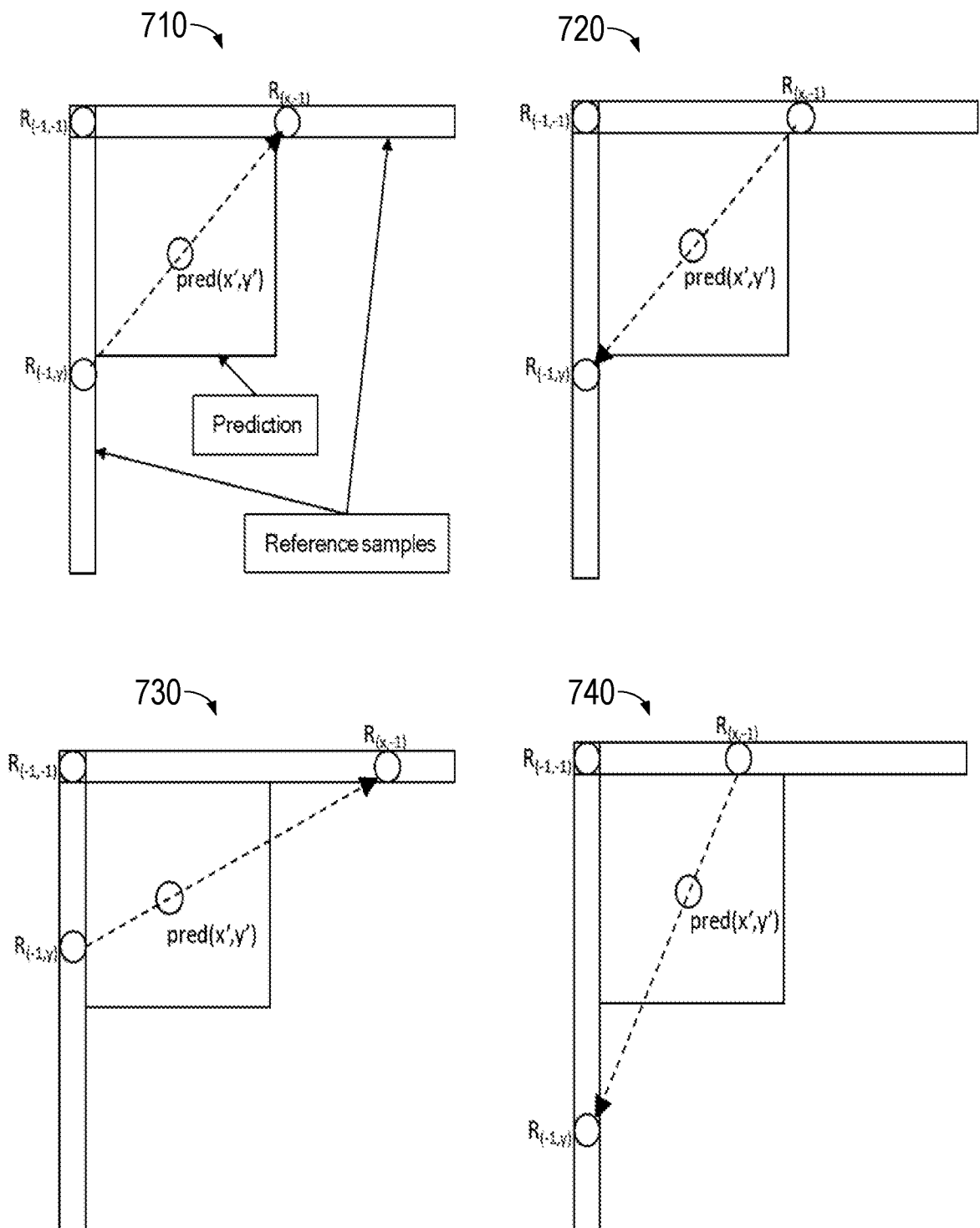


Fig. 7

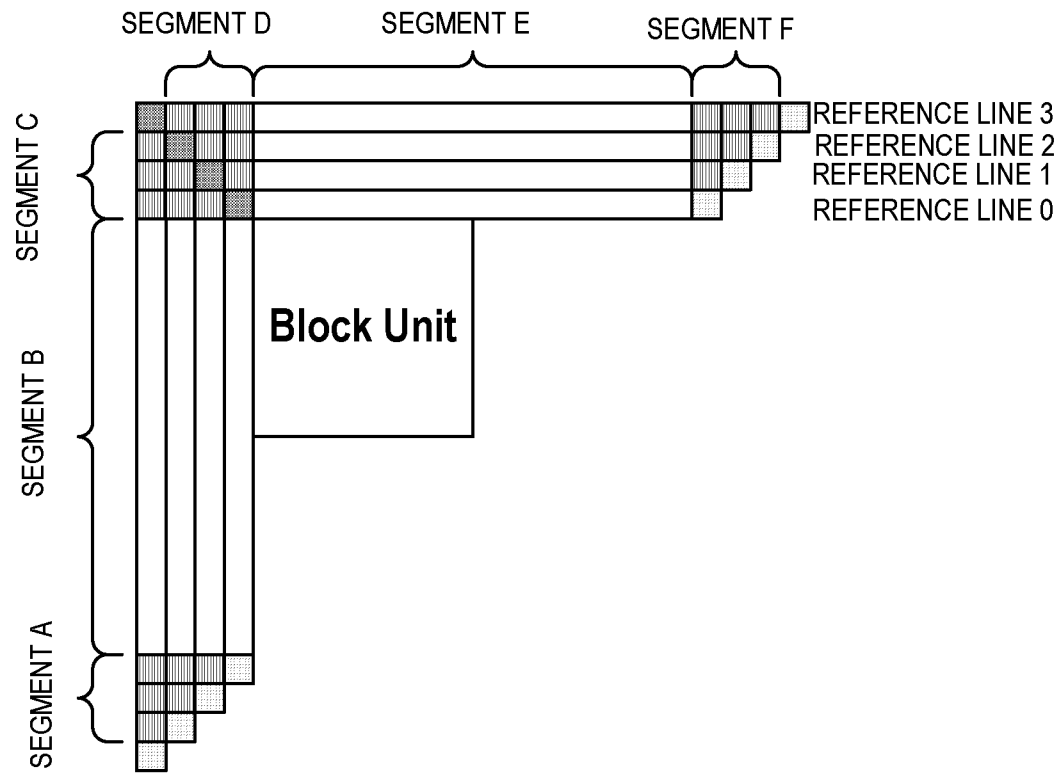


Fig. 8

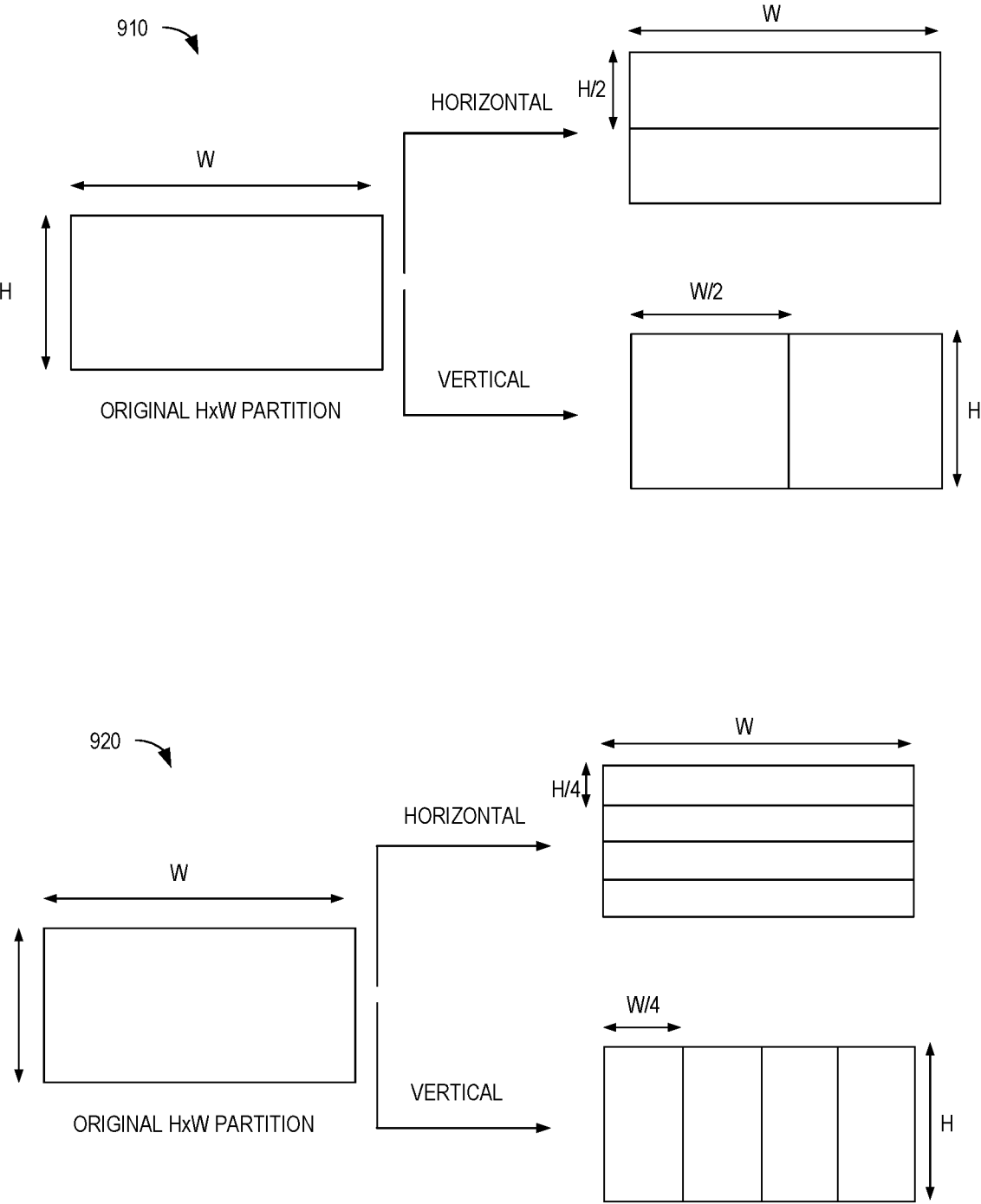


Fig. 9

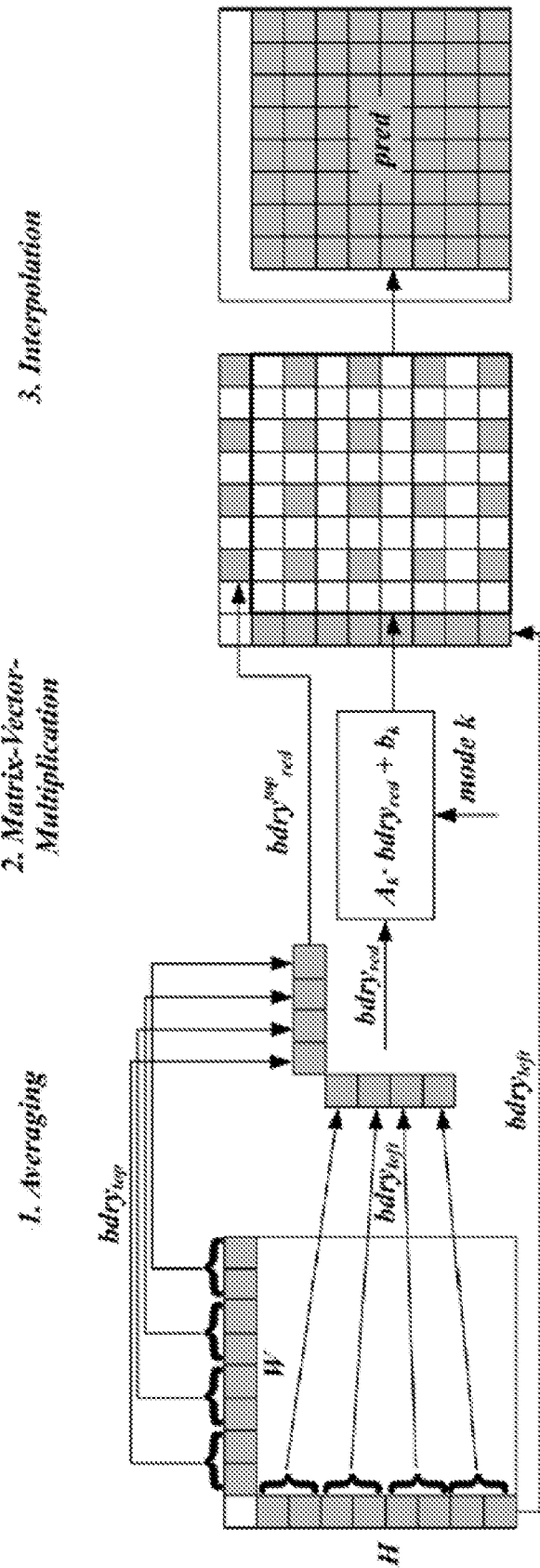


Fig. 10

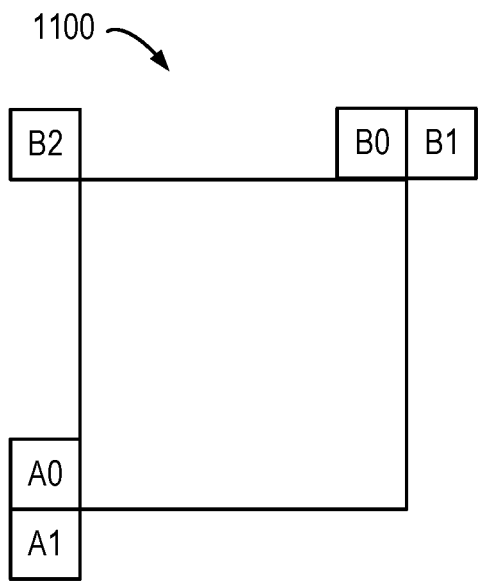


Fig. 11

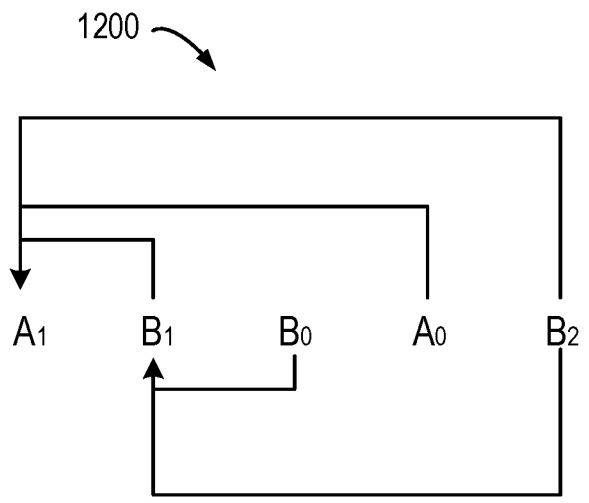


Fig. 12

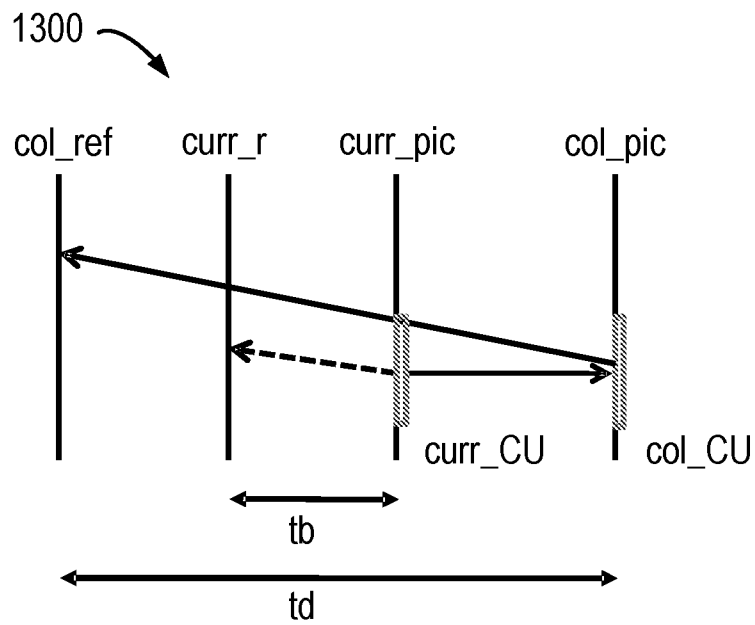


Fig. 13

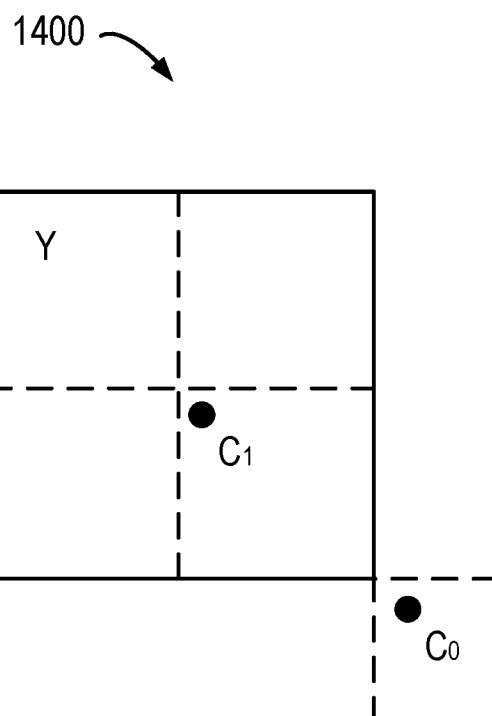


Fig. 14

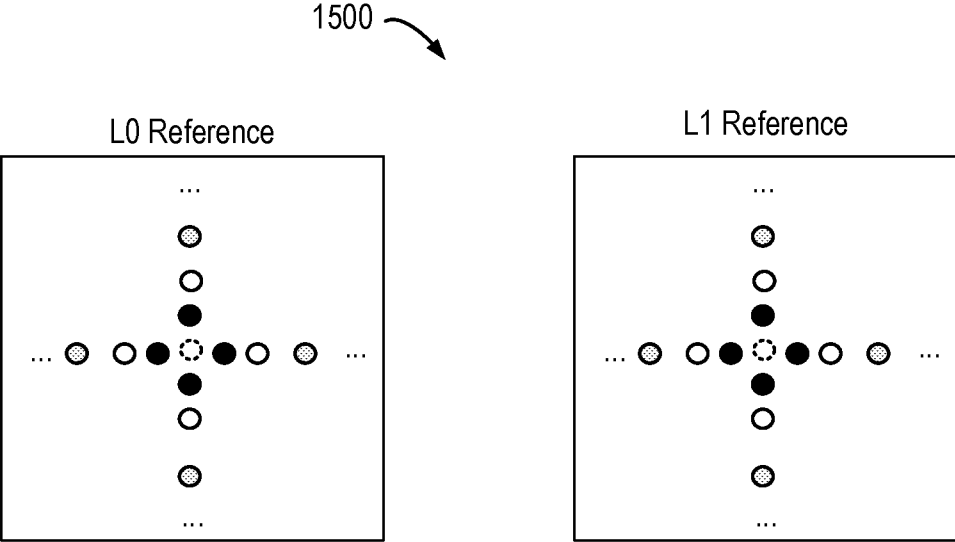


Fig. 15

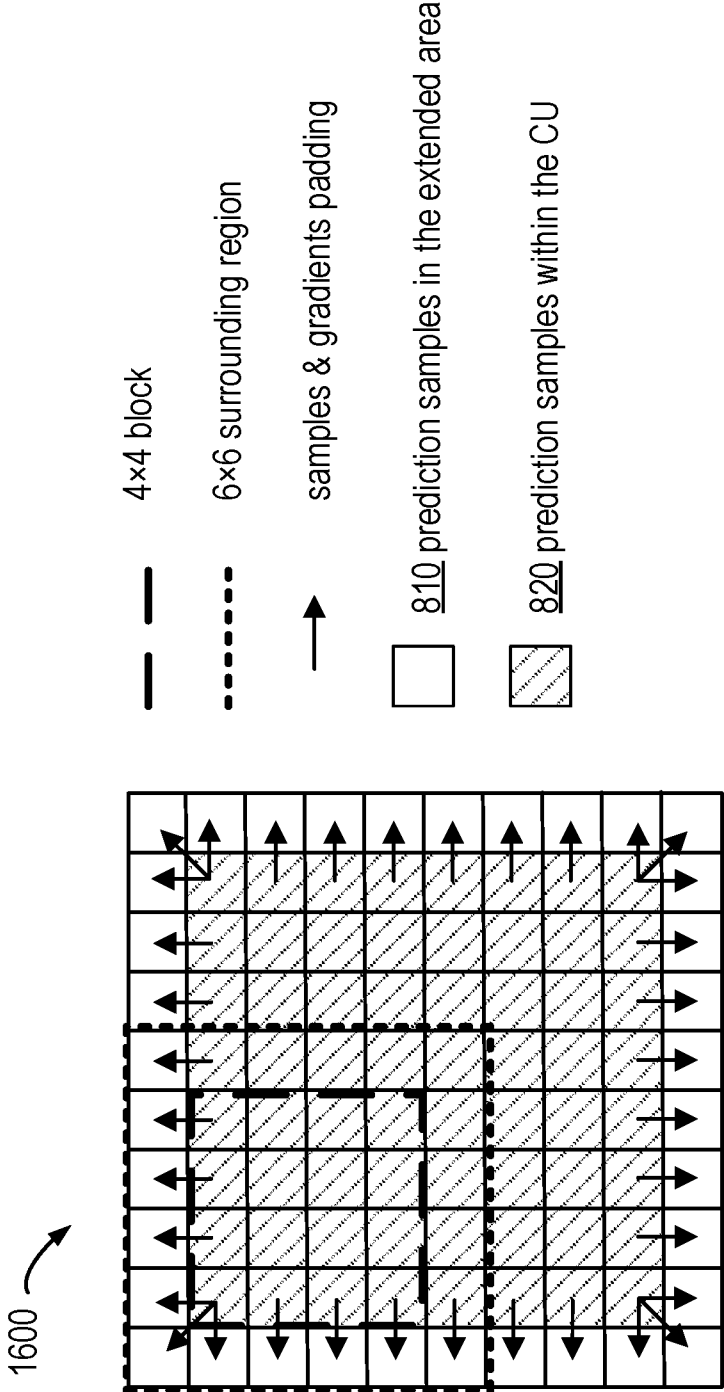


Fig. 16

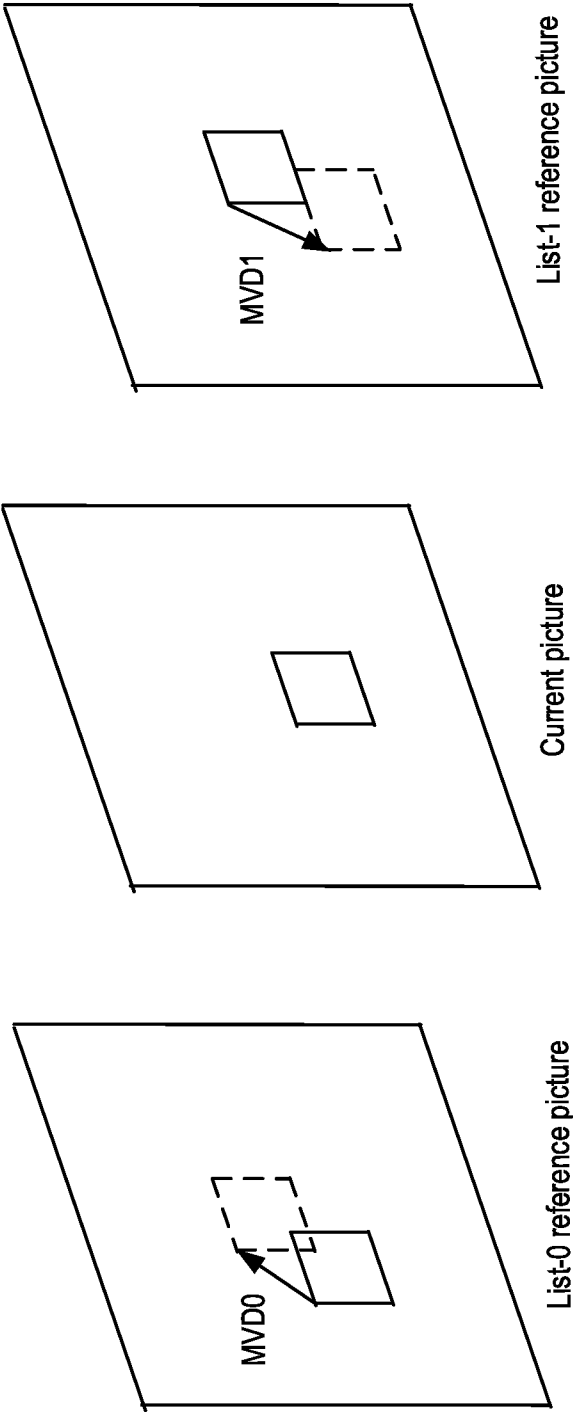


Fig. 17

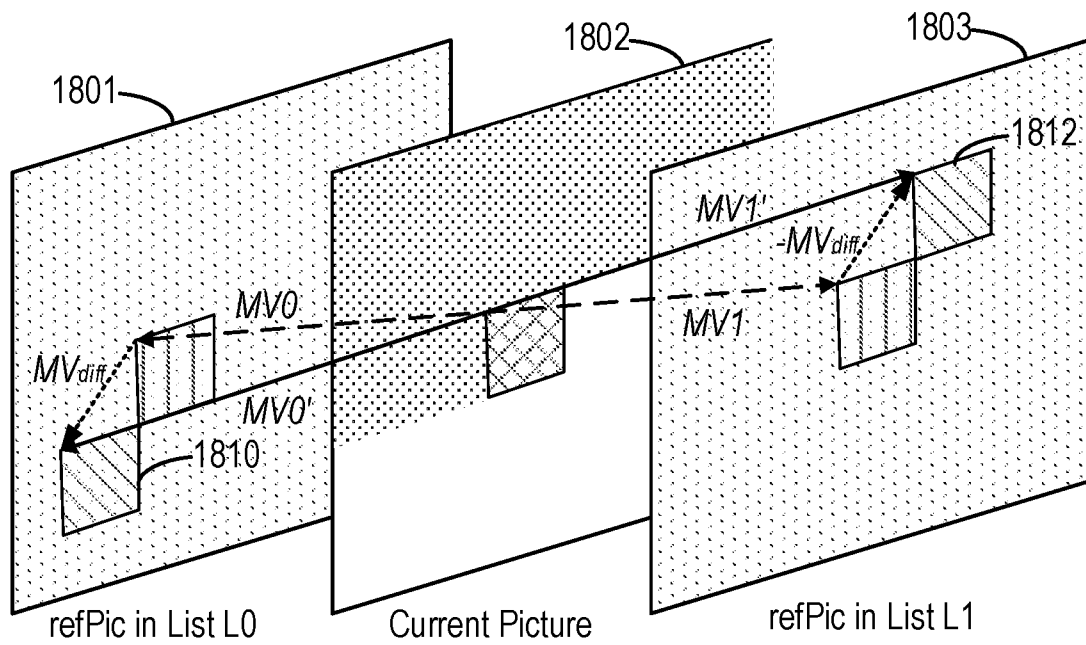


Fig. 18

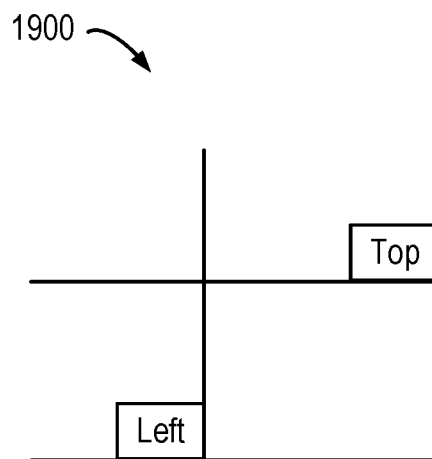


Fig. 19

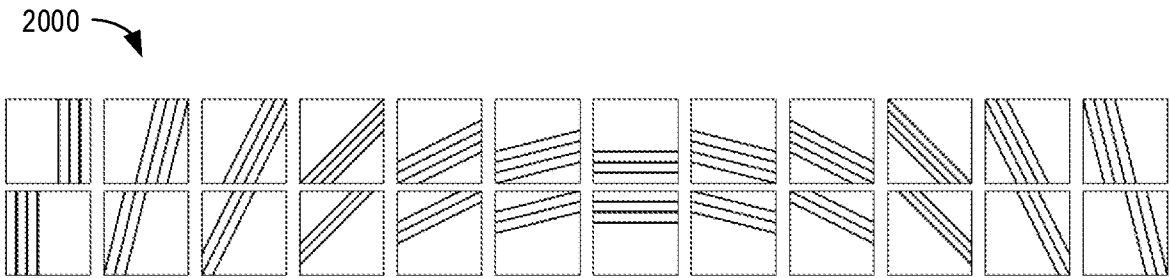


Fig. 20

2110

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

Fig. 21

2200

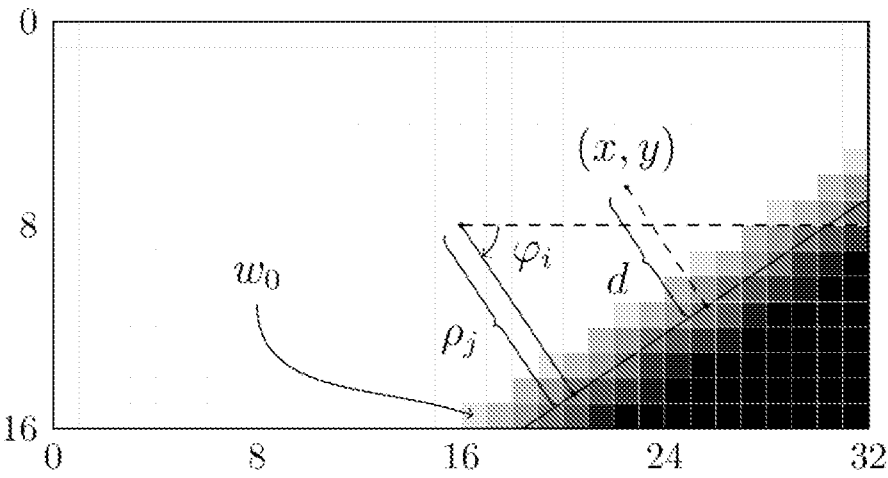
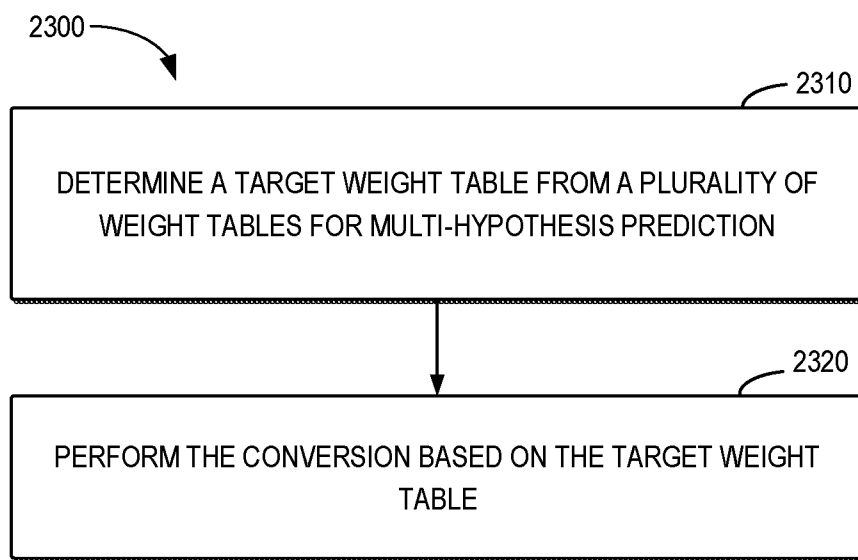
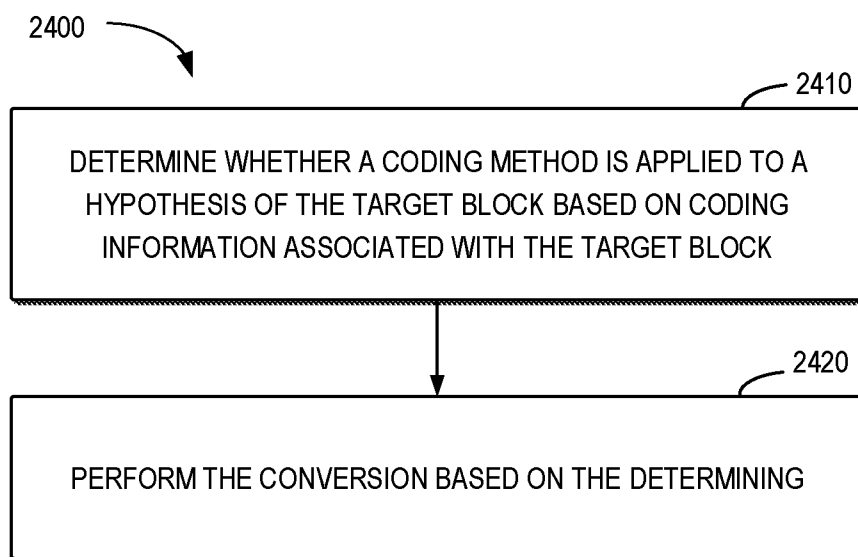


Fig. 22

**Fig. 23**

**Fig. 24**

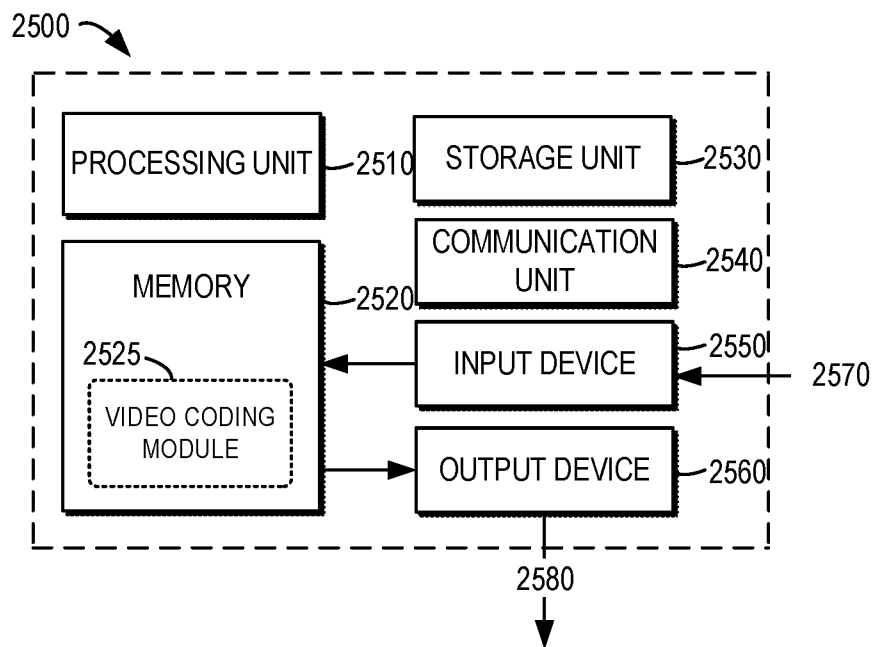


Fig. 25

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2022/101685

A. CLASSIFICATION OF SUBJECT MATTER

H04N 19/176(2014.01)i; H04N 19/107(2014.01)i; H04N 19/186(2014.01)i; H04N 19/159(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)

WPI, EPODOC, CNPAT, IEEE, CNKI: conversion, video, block, bitstream, weight, multi-hypothesis, multiple, hypothesis, prediction, table

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2021144392 A1 (BEIJING BYTEDANCE NETWORK TECHNOLOGY CO., LTD. et al.) 13 May 2021 (2021-05-13) description, paragraphs [0180], [0392]-[0396], [0613], [0674], [0780]	1-114
A	US 2021195200 A1 (TENCENT AMERICA LLC) 24 June 2021 (2021-06-24) the whole document	1-114
A	CN 112544082 A (MEDIATEK INC.) 23 March 2021 (2021-03-23) the whole document	1-114
A	CN 112840654 A (WILUS INST STANDARDS AND TECHNOLOGY INC. et al.) 25 May 2021 (2021-05-25) the whole document	1-114



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

07 September 2022

Date of mailing of the international search report

27 September 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

LI,Yiwei

Telephone No. 86-(10)-53961395

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2022/101685

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2021144392	A1	13 May 2021	EP	3857889	A1	04 August 2021
				WO	2020098782	A1	22 May 2020
				CN	111630858	A	04 September 2020
				KR	20210089149	A	15 July 2021
				JP	2022506119	A	17 January 2022
				CN	111448797	A	24 July 2020
				WO	2020098780	A1	22 May 2020
				EP	3857889	A4	22 September 2021
US	2021195200	A1	24 June 2021	KR	20210135311	A	12 November 2021
				CA	3134981	A1	01 July 2021
				WO	2021133456	A1	01 July 2021
				US	2022210427	A1	30 June 2022
				JP	2022526161	A	23 May 2022
				AU	2020414351	A1	28 October 2021
				SG	11202110603	A1	28 October 2021
				IN	202147050599	A	08 April 2022
				US	11317094	B2	26 April 2022
CN	112544082	A	23 March 2021	WO	2020015706	A1	23 January 2020
				TW	202007155	A	01 February 2020
				TW	720551	B1	01 March 2021
				EP	3824631	A1	26 May 2021
				US	2021274166	A1	02 September 2021
				US	2021329287	A1	21 October 2021
				US	11218721	B2	04 January 2022
				US	2022086478	A1	17 March 2022
CN	112840654	A	25 May 2021	WO	2020076143	A1	16 April 2020
				KR	20210057187	A	20 May 2021
				EP	3866468	A1	18 August 2021
				US	2021235072	A1	29 July 2021
				KR	2020041508	A	22 April 2020
				VN	79568	A	26 July 2021
				IN	202127017390	A	10 September 2021
				JP	2022502972	W	11 January 2022