



(51) International Patent Classification:

H04N 19/103 (2014.01) H04N 19/186 (2014.01)

H04N 19/176 (2014.01)

(21) International Application Number:

PCT/CN2022/140409

(22) International Filing Date:

20 December 2022 (20.12.2022)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

63/291,997 21 December 2021 (21.12.2021) US

(71) Applicant: **MEDIATEK INC.** [CN/CN]; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN).

(72) Inventors: **CHIANG, Man-Shu**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **CHUBACH, Olena**; 2840 Junction Ave, San Jose, California 95134 (US). **TSAI, Chia-Ming**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **HSIAO, Yu-Ling**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **CHEN,**

Chun-Chia; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **HSU, Chih-Wei**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **CHUANG, Tzu-Der**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **CHEN, Ching-Yeh**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **HUANG, Yu-Wen**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN).

(74) Agent: **BEIJING SANYOU INTELLECTUAL PROPERTY AGENCY LTD.**; 16th Fl., Block A, Corporate Square, No. 35 Jinrong Street, Beijing 100033 (CN).

(81) Designated States (*unless otherwise indicated, for every kind of national protection available*): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, 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,

(54) Title: METHOD AND APPARATUS FOR CROSS COMPONENT LINEAR MODEL WITH MULTIPLE HYPOTHESES INTRA MODES IN VIDEO CODING SYSTEM

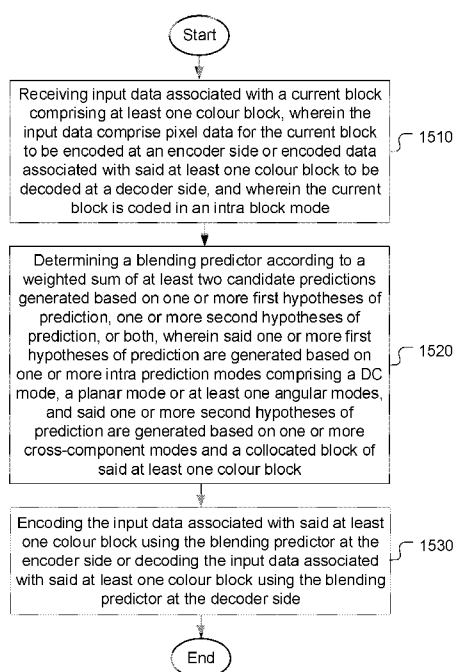


Fig. 15

(57) Abstract: A method and apparatus for inter prediction in video coding system are disclosed. According to the method, input data associated with a current block comprising at least one colour block are received. A blending predictor is determined according to a weighted sum of at least two candidate predictions generated based on one or more first hypotheses of prediction, one or more second hypotheses of prediction, or both. The first hypotheses of prediction are generated based on one or more intra prediction modes comprising a DC mode, a planar mode or at least one angular modes. The second hypotheses of prediction are generated based on one or more cross-component modes and a collocated block of said at least one colour block. The input data associated with the colour block is encoded or decoded using the blending predictor.

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, CV, 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, ME, 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))*

METHOD AND APPARATUS FOR CROSS COMPONENT LINEAR MODEL WITH MULTIPLE HYPOTHESES INTRA MODES IN VIDEO CODING SYSTEM

CROSS REFERENCE TO RELATED APPLICATIONS

The present invention is a non-Provisional Application of and claims priority to U.S. Provisional Patent Application No. 63/291,997, filed on December 21, 2021. The U.S. Provisional Patent Application is hereby incorporated by reference in its entirety.

FIELD OF THE INVENTION

The present invention relates to video coding system. In particular, the present invention relates to a new video coding tool for intra prediction using cross-component linear model in a video coding system.

BACKGROUND

Versatile video coding (VVC) is the latest international video coding standard developed by the Joint Video Experts Team (JVET) of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG). The standard has been published as an ISO standard: ISO/IEC 23090-3:2021, Information technology - Coded representation of immersive media - Part 3: Versatile video coding, published Feb. 2021. VVC is developed based on its predecessor HEVC (High Efficiency Video Coding) by adding more coding tools to improve coding efficiency and also to handle various types of video sources including 3-dimensional (3D) video signals.

Fig. 1A illustrates an exemplary adaptive Inter/Intra video encoding system incorporating loop processing. For Intra Prediction, the prediction data is derived based on previously coded video data in the current picture. For Inter Prediction 112, Motion Estimation (ME) is performed at the encoder side and Motion Compensation (MC) is performed based of the result of ME to provide prediction data derived from other picture(s) and motion data. Switch 114 selects Intra Prediction 110 or Inter-Prediction 112 and the selected prediction data is supplied to Adder 116 to form prediction errors, also called residues. The prediction error is then processed by Transform (T) 118 followed by Quantization (Q) 120. The transformed and quantized residues are then coded by Entropy Encoder 122 to be included in a video bitstream corresponding to the compressed video data. The bitstream associated with the transform coefficients is then packed with side information such as motion and coding modes associated with Intra prediction and Inter prediction, and other information such as parameters associated with loop filters applied to underlying image area. The side information associated with Intra Prediction 110, Inter prediction 112 and in-loop filter 130, are provided to Entropy Encoder 122 as shown in Fig. 1A. When an Inter-prediction mode is used, a reference picture or pictures have to be reconstructed at the encoder end as well. Consequently, the transformed and quantized residues are processed by Inverse Quantization (IQ) 124 and Inverse Transformation (IT) 126 to recover the residues. The residues are then added back to prediction data 136 at Reconstruction (REC) 128 to reconstruct video data. The reconstructed video data may be stored in Reference Picture Buffer 134 and used for prediction of other frames.

As shown in Fig. 1A, incoming video data undergoes a series of processing in the encoding system. The reconstructed video data from REC 128 may be subject to various impairments due to a series of processing. Accordingly, in-loop filter 130 is often applied to the reconstructed video data before the reconstructed video data are stored in the Reference Picture Buffer 134 in order to improve video quality. For example, deblocking filter (DF), Sample Adaptive Offset (SAO) and Adaptive Loop Filter (ALF) may be used. The loop filter information may need to be incorporated in the bitstream so that a decoder can properly recover the required information. Therefore, loop filter information is also provided to Entropy Encoder 122 for incorporation into the bitstream. In Fig. 1A, Loop filter 130 is applied to the reconstructed video before the reconstructed samples are stored in the reference picture buffer 134. The system in Fig. 1A is intended to illustrate an exemplary structure of a typical video encoder. It may correspond to the High Efficiency Video Coding (HEVC) system, VP8, VP9, H.264 or VVC.

The decoder, as shown in Fig. 1B, can use similar or portion of the same functional blocks as the encoder

except for Transform 118 and Quantization 120 since the decoder only needs Inverse Quantization 124 and Inverse Transform 126. Instead of Entropy Encoder 122, the decoder uses an Entropy Decoder 140 to decode the video bitstream into quantized transform coefficients and needed coding information (e.g. ILPF information, Intra prediction information and Inter prediction information). The Intra prediction 150 at the decoder side does not need to perform the mode search. Instead, the decoder only needs to generate Intra prediction according to Intra prediction information received from the Entropy Decoder 140. Furthermore, for Inter prediction, the decoder only needs to perform motion compensation (MC 152) according to Inter prediction information received from the Entropy Decoder 140 without the need for motion estimation.

According to VVC, an input picture is partitioned into non-overlapped square block regions referred as CTUs (Coding Tree Units), similar to HEVC. Each CTU can be partitioned into one or multiple smaller size coding units (CUs). The resulting CU partitions can be in square or rectangular shapes. Also, VVC divides a CTU into prediction units (PUs) as a unit to apply prediction process, such as Inter prediction, Intra prediction, etc.

The VVC standard incorporates various new coding tools to further improve the coding efficiency over the HEVC standard. In the present disclosure, various new coding tools are presented to improve the coding efficiency beyond the VVC. In particular, coding tools related to CCLM are disclosed.

BRIEF SUMMARY OF THE INVENTION

A method and apparatus for inter prediction in video coding system are disclosed. According to the method, input data associated with a current block comprising at least one colour block are received, where the input data comprise pixel data for the current block to be encoded at an encoder side or encoded data associated with said at least one colour block to be decoded at a decoder side, and wherein the current block is coded in an intra block mode. A blending predictor is determined according to a weighted sum of at least two candidate predictions generated based on one or more first hypotheses of prediction, one or more second hypotheses of prediction, or both, where said one or more first hypotheses of prediction are generated based on one or more intra prediction modes comprising a DC mode, a planar mode or at least one angular modes, and said one or more second hypotheses of prediction are generated based on one or more cross-component modes and a collocated block of said at least one colour block. The input data associated with said at least one colour block is encoded using the blending predictor at the encoder side or the input data associated with said at least one colour block is decoded using the blending predictor at the decoder side.

In one embodiment, said one or more second hypotheses of prediction correspond to CCLM (Cross-Colour Linear Model) prediction, wherein model parameters a and b for the CCLM prediction are determined based on neighbouring reconstructed pixels of the collocated block and neighbouring reconstructed pixels of said at least one colour block, and a pixel value for said at least one colour block is predicted according to $a \cdot rec_L' + b$ and rec_L' corresponds to one down-sampled pixel value for the collocated block. In another embodiment, said one or more second hypotheses of prediction correspond to MMLM (Multiple Model CCLM) and one of two models of CCLM (Cross-Colour Linear Model) prediction is selected according to a reconstructed pixel value for the collocated block.

In one embodiment, equal weights are used for the weighted sum of said at least two candidate predictions. In another embodiment, weights for the weighted sum of said at least two candidate predictions are determined according to neighbouring coding information, sample position, block width, block height, block area, coding mode or a combination thereof. For example, when more neighbouring blocks of the current block are coded with a target mode, a larger weight can be used for one hypothesis of prediction associated with the target mode. Furthermore, the target mode may correspond to one cross-component mode. In another example, the current block is partitioned into multiple regions and sample positions in a same region share a same weight. In one embodiment, if a current region is close to L-shaped reference neighbours, one first hypotheses of prediction uses a higher weight than one second hypotheses of prediction.

In one embodiment, said at least two candidate predictions comprise at least one first hypothesis of prediction and at least one second hypothesis of prediction.

In another embodiment, the said blending predictor is only applied to a region within the current block. For

example, the region may correspond to a right-bottom region of the current block.

In one embodiment, said at least one colour block corresponds to a Cr block and the collocated block corresponds to a Cb block, or said at least one colour block corresponds to the Cb block and the collocated block corresponds to the Cr block. In another example, said at least two second hypotheses of prediction are generated from one or more pre-defined cross-component modes.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1A illustrates an exemplary adaptive Inter/Intra video coding system incorporating loop processing.

Fig. 1B illustrates a corresponding decoder for the encoder in Fig. 1A.

Fig. 2 illustrates the 67 intra modes as adopted by the VVC (versatile video coding) standard.

Fig. 3 illustrates an example of Multiple Reference Line (MRL) intra prediction, where 4 reference lines are used for intra prediction.

Fig. 4A illustrates an example of Intra Sub-Partition (ISP), where a block is partitioned in two subblocks horizontally or vertically.

Fig. 4B illustrates an example of Intra Sub-Partition (ISP), where a block is partitioned in four subblocks horizontally or vertically.

Fig. 5 illustrates an example of processing flow for Matrix weighted intra prediction (MIP).

Fig. 6 illustrates the reference region of IBC Mode, where each block represents 64x64 luma sample unit and the reference region depends on the location of the current coded CU.

Fig. 7 shows the relative sample locations of $M \times N$ chroma block, the corresponding $2M \times 2N$ luma block and their neighbouring samples (shown as filled circles and triangles) of “type-0” content.

Fig. 8 illustrates an example of the reconstructed neighbouring samples pre-processed before being becoming the inputs for deriving model parameters.

Fig. 9 illustrates an example of the relationship between the cr prediction, cb prediction and JCCLM predictors.

Fig. 10 illustrates an example of Adaptive Intra-mode selection, where the chroma block is divided into 4 sub-blocks.

Figs. 11A-C illustrate some possible ways to partition the current block and the weight selection for prediction from CCLM associated with these partitions.

Fig. 12 illustrates an example of Cross-CU LM, where the block has an irregular pattern that no angular intra prediction can provide a good prediction.

Fig. 13 illustrates an example that a luma picture area associated with a node contains irregular patterns and the picture area is divided into various blocks for applying inter or intra prediction.

Figs. 14A-B illustrate examples of using LM mode to generate the right-bottom region within (Fig. 14A) or outside (Fig. 14B) the current block.

Fig. 15 illustrates a flowchart of an exemplary video coding system that utilizes cross-colour linear model for intra mode according to an embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

It will be readily understood that the components of the present invention, as generally described and illustrated in the figures herein, may be arranged and designed in a wide variety of different configurations. Thus, the following more detailed description of the embodiments of the systems and methods of the present invention, as represented in the figures, is not intended to limit the scope of the invention, as claimed, but is merely representative of selected embodiments of the invention. References throughout this specification to “one embodiment,” “an embodiment,” or similar language mean that a particular feature, structure, or characteristic described in connection with the embodiment may be included in at least one embodiment of the present invention. Thus, appearances of the phrases “in one embodiment” or “in an embodiment” in various places throughout this specification are not necessarily all referring to the same embodiment.

Furthermore, the described features, structures, or characteristics may be combined in any suitable manner in

one or more embodiments. One skilled in the relevant art will recognize, however, that the invention can be practiced without one or more of the specific details, or with other methods, components, etc. In other instances, well-known structures, or operations are not shown or described in detail to avoid obscuring aspects of the invention. The illustrated embodiments of the invention will be best understood by reference to the drawings, wherein like parts are designated by like numerals throughout. The following description is intended only by way of example, and simply illustrates certain selected embodiments of apparatus and methods that are consistent with the invention as claimed herein.

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 (angular) modes not in HEVC are depicted as red dotted arrows in Fig. 2, and the planar and DC modes remain the same. These denser directional intra prediction modes are applied for all block sizes and for both luma and chroma intra predictions.

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 neighbouring intra modes. The following three aspects are considered to construct the MPM list:

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

Multiple Reference Line Intra Prediction

Multiple reference line (MRL) intra prediction uses more reference lines for intra prediction. In Fig. 3, an example of 4 reference lines is depicted, where the samples of segments A and F are not fetched from reconstructed neighbouring samples but padded with the closest samples from segments 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 that 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 (Position-Dependent Prediction Combination) is disabled when an 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 neighbouring luma reference lines with a CTU to generate predictions. The Cross-Component Linear Model (CCLM) tool also requires 3 neighbouring luma reference lines for its down-sampling filters. The definition of MRL to use the same 3 lines is aligned with CCLM to reduce the storage requirements for decoders.

Intra Sub-partitions

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, the minimum block size for ISP is 4x8 (or 8x4). If block size is greater than 4x8 (or 8x4), then the corresponding block is divided by 4 sub-partitions. It has been noted that the $M \times 128$ (with $M \leq 64$) and $128 \times N$ (with $N \leq 64$) ISP blocks could generate a potential issue with the 64×64 VDPU (Virtual Decoder Pipeline Unit). 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 size that can use ISP is restricted to a maximum of 64×64 . Fig. 4A and Fig. 4B shows examples of the two possibilities. All sub-partitions fulfil the condition of having at least 16 samples.

In ISP, the dependence of 1xN and 2xN subblock prediction on the reconstructed values of previously decoded 1xN and 2xN subblocks of the coding block is not allowed so that the minimum width of prediction for subblocks becomes four samples. For example, an 8xN (N > 4) coding block that is coded using ISP with vertical split is partitioned into two prediction regions each of size 4xN and four transforms of size 2xN. Also, a 4xN coding block that is coded using ISP with vertical split is predicted using the full 4xN block; four transform each of 1xN is used. Although the transform sizes of 1xN and 2xN are allowed, it is asserted that the transform of these blocks in 4xN regions can be performed in parallel. For example, when a 4xN prediction region contains four 1xN transforms, there is no transform in the horizontal direction; the transform in the vertical direction can be performed as a single 4xN transform in the vertical direction. Similarly, when a 4xN prediction region contains two 2xN transform blocks, the transform operation of the two 2xN blocks in each direction (horizontal and vertical) can be conducted in parallel. Thus, there is no delay added in processing these smaller blocks compared to processing 4x4 regular-coded intra blocks.

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 consecutively. 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.

Matrix Weighted Intra Prediction

Matrix weighted intra prediction (MIP) method is a newly added intra prediction technique in 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, i.e., averaging, matrix vector multiplication and linear interpolation as shown in Fig. 5. One line of H reconstructed neighbouring boundary samples left of the block and one line of W reconstructed neighbouring boundary samples above the block are shown as dot-filled small squares. After the averaging process, the boundary samples are down-sampled to top boundary line and left boundary line. The down-sampled samples are provided to the matrix-vector multiplication unit to generate the down-sampled prediction block. An interpolation process is then applied to generate the prediction block.

Averaging neighbouring samples

Among the boundary samples, four samples or eight samples are selected by averaging based on the block size and shape. Specifically, the input boundaries $bdry_{red}^{top}$ and $bdry_{red}^{left}$ are reduced to smaller boundaries $bdry_{red}^{top}$ and $bdry_{red}^{left}$ by averaging neighbouring boundary samples according to a predefined rule depending 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:

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

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 down-sampled block of width W_{red} and height H_{red} is generated. Here, W_{red} and H_{red} are defined as:

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

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

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 for $W = H = 4$ and 8 columns for 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:

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

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

Interpolation

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

Signalling 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 ($predModeIntra$) is signalled. For an MIP mode, a transposed flag ($isTransposed$), which determines whether the mode is transposed, and MIP mode Id ($modeId$), which determines which matrix is to be used for the given MIP mode is derived as follows

$$isTransposed = predModeIntra \& 1$$

$$modeId = predModeIntra >> 1$$

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

- LFNST (Low-Frequency Non-Separable Transform) 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 up-sampling step used in the MIP-prediction, original reference samples are used instead of down-sampled ones
- Clipping is performed before up-sampling and not after up-sampling
- MIP is allowed up to 64×64 regardless of the maximum transform size
- The number of MIP modes is 32 for $sizeId=0$, 16 for $sizeId=1$ and 12 for $sizeId=2$

Intra Block Copy

Intra block copy (IBC) is a tool adopted in HEVC extensions on SCC (Screen Content Coding). It is well known that it significantly improves the coding efficiency of screen content materials. Since IBC mode is implemented as a block level coding mode, block matching (BM) is performed at the encoder to find the optimal block vector (or motion vector) for each CU. Here, a block vector is used to indicate the displacement from the current block to a reference block, which is already reconstructed inside the current picture. The luma block vector of an IBC-coded CU is in integer precision. The chroma block vector is rounded to integer precision as

well. When combined with AMVR (Adaptive Motion Vector Resolution), the IBC mode can switch between 1-pel and 4-pel motion vector precisions. An IBC-coded CU is treated as the third prediction mode other than intra or inter prediction modes. The IBC mode is applicable to the CUs with both width and height smaller than or equal to 64 luma samples.

At the encoder side, hash-based motion estimation is performed for IBC. The encoder performs RD check for blocks with either width or height no larger than 16 luma samples. For non-merge mode, the block vector search is performed using hash-based search first. If hash search does not return a valid candidate, block matching based local search will be performed.

In the hash-based search, hash key matching (32-bit CRC) between the current block and a reference block is extended to all allowed block sizes. The hash key calculation for every position in the current picture is based on 4x4 subblocks. For the current block of a larger size, a hash key is determined to match that of the reference block when all the hash keys of all 4x4 subblocks match the hash keys in the corresponding reference locations. If hash keys of multiple reference blocks are found to match that of the current block, the block vector costs of each matched reference are calculated and the one with the minimum cost is selected.

In block matching search, the search range is set to cover both the previous and current CTUs.

At CU level, IBC mode is signalled with a flag and it can be signalled as IBC AMVP (Advanced Motion Vector Prediction) mode or IBC skip/merge mode as follows:

- IBC skip/merge mode: a merge candidate index is used to indicate which of the block vectors in the list from neighbouring candidate IBC coded blocks is used to predict the current block. The merge list consists of spatial, HMVP (History based Motion Vector Prediction), and pairwise candidates.
- IBC AMVP mode: block vector difference is coded in the same way as a motion vector difference. The block vector prediction method uses two candidates as predictors, one from left neighbour and one from above neighbour (if IBC coded). When either neighbour is not available, a default block vector will be used as a predictor. A flag is signalled to indicate the block vector predictor index.

IBC Reference Region

To reduce memory consumption and decoder complexity, the IBC in VVC allows only the reconstructed portion of the predefined area including the region of current CTU and some region of the left CTU. Fig. 6 illustrates the reference region of IBC Mode, where each block represents 64x64 luma sample unit. Depending on the location of the current coded CU within the current CTU, the following applies:

- If the current block falls into the top-left 64x64 block of the current CTU (case 610 in Fig. 6), then in addition to the already reconstructed samples in the current CTU, it can also refer to the reference samples in the bottom-right 64x64 blocks of the left CTU, using current picture referencing (CPR) mode. (More details of CPR can be found in JVET-T2002 (Jianle Chen, et. al., “Algorithm description for Versatile Video Coding and Test Model 11 (VTM 11)”, Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29, 20th Meeting, by teleconference, 7 – 16 October 2020, Document: JVET-T2002)). The current block can also refer to the reference samples in the bottom-left 64x64 block of the left CTU and the reference samples in the top-right 64x64 block of the left CTU, using CPR mode.
- If the current block falls into the top-right 64x64 block of the current CTU (case 620 in Fig. 6), then in addition to the already reconstructed samples in the current CTU, if luma location (0, 64) relative to the current CTU has not yet been reconstructed, the current block can also refer to the reference samples in the bottom-left 64x64 block and bottom-right 64x64 block of the left CTU, using CPR mode; otherwise, the current block can also refer to reference samples in bottom-right 64x64 block of the left CTU.
- If the current block falls into the bottom-left 64x64 block of the current CTU (case 630 in Fig. 6), then in addition to the already reconstructed samples in the current CTU, if luma location (64, 0) relative to the current CTU has not yet been reconstructed, the current block can also refer to the reference samples in the top-right 64x64 block and bottom-right 64x64 block of the left CTU, using CPR mode. Otherwise, the current block can also refer to the reference samples in the bottom-right 64x64 block of the left CTU, using CPR mode.

- If current block falls into the bottom-right 64x64 block of the current CTU (case 640 in Fig. 6), it can only refer to the already reconstructed samples in the current CTU, using CPR mode.

This restriction allows the IBC mode to be implemented using local on-chip memory for hardware implementations.

5 Joint Coding of Chroma Residuals

VVC supports the joint coding of chroma residual (JCCR) tool where the chroma residuals are coded jointly. The usage (activation) of the JCCR mode is indicated by a TU-level flag *tu_joint_cbr_residual_flag* and the selected mode is implicitly indicated by the chroma CBFs. The flag *tu_joint_cbr_residual_flag* is present if either or both chroma CBFs for a TU are equal to 1. In the PPS (Picture Parameter Set) and slice header, chroma QP offset values are signalled for the JCCR mode to differentiate from the usual chroma QP offset values signalled for regular chroma residual coding mode. These chroma QP offset values are used to derive the chroma QP values for some blocks coded using the JCCR mode. The JCCR mode has 3 sub-modes. When a corresponding JCCR sub-mode (sub-mode 2 in Table 1) is active in a TU, this chroma QP offset is added to the applied luma-derived chroma QP during quantization and decoding of that TU. For the other JCCR sub-modes (sub-modes 1 and 3 in Table 1), the chroma QPs are derived in the same way as for conventional Cb or Cr blocks. The reconstruction process of the chroma residuals (resCb and resCr) from the transmitted transform blocks is depicted in Table 1. When the JCCR mode is activated, one single joint chroma residual block (resJointC[x][y] in Table 1) is signalled, and residual block for Cb (resCb) and residual block for Cr (resCr) are derived considering information such as *tu_cbf_cb*, *tu_cbf_cr*, and CSign, which is a sign value specified in the slice header.

At the encoder side, the joint chroma components are derived as explained in the following. Depending on the mode (listed in the tables above), resJointC{1,2} are generated by the encoder as follows:

- If mode is equal to 2 (single residual with reconstruction $C_b = C$, $C_r = \text{CSign} * C$), the joint residual is determined according to

$$\text{resJointC}[x][y] = (\text{resCb}[x][y] + \text{CSign} * \text{resCr}[x][y]) / 2$$

- Otherwise, if mode is equal to 1 (single residual with reconstruction $C_b = C$, $C_r = (\text{CSign} * C) / 2$), the joint residual is determined according to

$$\text{resJointC}[x][y] = (4 * \text{resCb}[x][y] + 2 * \text{CSign} * \text{resCr}[x][y]) / 5$$

- Otherwise (mode is equal to 3, i. e., single residual, reconstruction $C_r = C$, $C_b = (\text{CSign} * C) / 2$), the joint residual is determined according to

$$\text{resJointC}[x][y] = (4 * \text{resCr}[x][y] + 2 * \text{CSign} * \text{resCb}[x][y]) / 5$$

Table 1. Reconstruction of chroma residuals. The value CSign is a sign value (+1 or -1), which is specified in the slice header, resJointC[][] is the transmitted residual.

<i>tu_cbf_cb</i>	<i>tu_cbf_cr</i>	reconstruction of Cb and Cr residuals	mode
1	0	$\text{resCb}[x][y] = \text{resJointC}[x][y]$ $\text{resCr}[x][y] = (\text{CSign} * \text{resJointC}[x][y]) \gg 1$	1
1	1	$\text{resCb}[x][y] = \text{resJointC}[x][y]$ $\text{resCr}[x][y] = \text{CSign} * \text{resJointC}[x][y]$	2
0	1	$\text{resCb}[x][y] = (\text{CSign} * \text{resJointC}[x][y]) \gg 1$ $\text{resCr}[x][y] = \text{resJointC}[x][y]$	3

The three joint chroma coding sub-modes described above are only supported in I slices. In P and B slices, only mode 2 is supported. Hence, in P and B slices, the syntax element *tu_joint_cbr_residual_flag* is only present if both chroma cbfs are 1.

The JCCR mode can be combined with the chroma transform skip (TS) mode (more details of the TS mode can be found in Section 3.9.3 of JVET-T2002). To speed up the encoder decision, the JCCR transform selection depends on whether the independent coding of Cb and Cr components selects the DCT-2 or the TS as the best transform, and whether there are non-zero coefficients in independent chroma coding. Specifically, if one chroma component selects DCT-2 (or TS) and the other component is all zero, or both chroma components select DCT-2

(or TS), then only DCT-2 (or TS) will be considered in JCCR encoding. Otherwise, if one component selects DCT-2 and the other selects TS, then both, DCT-2 and TS, will be considered in JCCR encoding.

CCLM (Cross Component Linear Model)

The main idea behind CCLM mode (sometimes abbreviated as LM mode) is as follows: chroma components of a block can be predicted from the collocated reconstructed luma samples by linear models whose parameters are derived from already reconstructed luma and chroma samples that are adjacent to the block.

In VVC, the CCLM mode makes use of inter-channel dependencies by predicting the chroma samples from reconstructed luma samples. This prediction is carried out using a linear model in the form

$$P(i, j) = a \cdot rec'_L(i, j) + b. \quad (1)$$

Here, $P(i, j)$ represents the predicted chroma samples in a CU and $rec'_L(i, j)$ represents the reconstructed luma samples of the same CU which are down-sampled for the case of non-4:4:4 colour format. The model parameters a and b are derived based on reconstructed neighbouring luma and chroma samples at both encoder and decoder side without explicit signalling.

Three CCLM modes, i.e., CCLM_LT, CCLM_L, and CCLM_T, are specified in VVC. These three modes differ with respect to the locations of the reference samples that are used for model parameter derivation. Samples only from the top boundary are involved in the CCLM_T mode and samples only from the left boundary are involved in the CCLM_L mode. In the CCLM_LT mode, samples from both the top boundary and the left boundary are used.

Overall, the prediction process of CCLM modes consists of three steps:

- 1) Down-sampling of the luma block and its neighbouring reconstructed samples to match the size of corresponding chroma block,
- 2) Model parameter derivation based on reconstructed neighbouring samples, and
- 3) Applying the model equation (1) to generate the chroma intra prediction samples.

Down-sampling of the Luma Component: To match the chroma sample locations for 4:2:0 or 4:2:2: colour format video sequences, two types of down-sampling filter can be applied to luma samples, both of which have a 2-to-1 down-sampling ratio in the horizontal and vertical directions. These two filters correspond to “type-0” and “type-2” 4:2:0 chroma format content, respectively and are given by

$$f_1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \quad f_2 = \begin{pmatrix} 1 & 2 & 1 \\ 1 & 2 & 1 \end{pmatrix}. \quad (2)$$

Based on the SPS-level flag information, the 2-dimensional 6-tap (i.e., f_2) or 5-tap (i.e., f_1) filter is applied to the luma samples within the current block as well as its neighbouring luma samples. The SPS-level refers to Sequence Parameter Set level. An exception happens if the top line of the current block is a CTU boundary. In this case, the one-dimensional filter $[1, 2, 1]/4$ is applied to the above neighbouring luma samples in order to avoid the usage of more than one luma line above the CTU boundary.

Model Parameter Derivation Process: The model parameters a and b from eqn. (1) are derived based on reconstructed neighbouring luma and chroma samples at both encoder and decoder sides to avoid the need for any signalling overhead. In the initially adopted version of the CCLM mode, the linear minimum mean square error (LMMSE) estimator was used for derivation of the parameters. In the final design, however, only four samples are involved to reduce the computational complexity. Fig. 7 shows the relative sample locations of $M \times N$ chroma block 710, the corresponding $2M \times 2N$ luma block 720 and their neighbouring samples (shown as filled circles and triangles) of “type-0” content.

In the example of Fig. 7, the four samples used in the CCLM_LT mode are shown, which are marked by triangular shape. They are located at the positions of $M/4$ and $M \cdot 3/4$ at the top boundary and at the positions of $N/4$ and $N \cdot 3/4$ at the left boundary. In CCLM_T and CCLM_L modes, the top and left boundary are extended to a size of $(M+N)$ samples, and the four samples used for the model parameter derivation are located at the positions $(M+N)/8$, $(M+N) \cdot 3/8$, $(M+N) \cdot 5/8$, and $(M+N) \cdot 7/8$.

Once the four samples are selected, four comparison operations are used to determine the two smallest and the two largest luma sample values among them. Let X_l denote the average of the two largest luma sample values

and let X_s denote the average of the two smallest luma sample values. Similarly, let Y_l and Y_s denote the averages of the corresponding chroma sample values. Then, the linear model parameters are obtained according to the following equation:

$$a = \frac{Y_l - Y_s}{X_l - X_s}$$

$$b = Y_{s-a} \cdot X_s . \quad (3)$$

In this equation, the division operation to calculate the parameter a is implemented with a look-up table. To reduce the memory required for storing this table, the *diff* value, which is the difference between the maximum and minimum values, and the parameter a are expressed by an exponential notation. Here, the value of *diff* is approximated with a 4-bit significant part and an exponent. Consequently, the table for $1/diff$ only consists of 16 elements. This has the benefit of both reducing the complexity of the calculation and decreasing the memory size required for storing the tables.

MMLM Overview

As indicated by the name, the original CCLM mode employs one linear model for predicting the chroma samples from the luma samples for the whole CU, while in MMLM (Multiple Model CCLM), there can be two models. In MMLM, neighbouring luma samples and neighbouring chroma samples of the current block are classified into two groups, each group is used as a training set to derive a linear model (i.e., particular α and β are derived for a particular group). Furthermore, the samples of the current luma block are also classified based on the same rule for the classification of neighbouring luma samples.

- *Threshold* is calculated as the average value of the neighbouring reconstructed luma samples. A neighbouring sample with $Rec'L[x,y] \leq Threshold$ is classified into group 1; while a neighbouring sample with $Rec'L[x,y] > Threshold$ is classified into group 2.
- Correspondingly, a prediction for chroma is obtained using linear models:

$$\begin{cases} Pred_c[x,y] = \alpha_1 \times Rec'_L[x,y] + \beta_1 & \text{if } Rec'_L[x,y] \leq Threshold \\ Pred_c[x,y] = \alpha_2 \times Rec'_L[x,y] + \beta_2 & \text{if } Rec'_L[x,y] > Threshold \end{cases}$$

Chroma DM (Derived Mode) Mode

For Chroma DM mode, the intra prediction mode of the corresponding (collocated) luma block covering the centre position of the current chroma block is directly inherited.

Reconstructed Neighbouring Sample Pre-processing

When deriving model parameters, reconstructed neighbouring samples for the first component and second component are used. Take the CCLM described in the overview section as an example. The first component is luma and the second component is cb or cr. To improve the model performance, the reconstructed neighbouring samples are pre-processed before becoming the inputs for deriving model parameters.

Fig. 8 illustrates an example of the reconstructed neighbouring samples pre-processed before being becoming the inputs for deriving model parameters, where a neighbouring region 810 of a luma block 812 and a neighbouring region 820 of a chroma (cb or cr) block 812 are pre-processed before being provided to the model parameter derivation block 830.

In one embodiment, the reconstructed neighbouring samples of the first component are pre-processed.

In one embodiment, the reconstructed neighbouring samples of the second component are pre-processed.

In another embodiment, the reconstructed neighbouring samples of only one of the first and the second component are pre-processed.

In one embodiment, the pre-processing methods can be (but are not limited to) any one or any combination of following processes: 3x3 or 5x5 filtering, biasing, clipping, filtering or clipping like ALF or CCALF, SAO-like filtering, filter sets (e.g. ALF sets)

In another embodiment, the first component is any one of luma, cb, and cr. For example, when the first component is luma, the second component is cb or cr. For another example, when the first component is cb, the second component is luma or cr. For another example, when the first component is cr, the second component is luma or cb. For another example, when the first component is luma, the second component is based on weighted combination of cb and cr.

In one embodiment, the pre-processing method of one component (e.g. cr) depends on another component (e.g. cb). For example, the selection of pre-processing method for cb is derived according to signalling/bitstream and cr follows cb's selection. For another example, it is assumed that high correlation exists between cb and cr, so the selection of pre-processing method for cr is shown as follows:

- The cb reconstruction (without pre-processing) plus cb residuals are treated as golden
- Choosing cr's pre-processing method according to cb's pre-processed reconstruction and golden
- For example, if the cb's pre-processed reconstruction is very similar to golden, use cb's pre-processing method as cr's pre-processing method.

In another embodiment, the pre-processing method is applied right after reconstructing neighbouring samples of the first and/or second component.

In another embodiment, the pre-processing method is applied to the reconstructed neighbouring samples before generating the model parameters for the current block.

Prediction Sample Post-processing

After applying CCLM to the current block, the prediction of the current block is generated and can be further adjusted with post-processing methods. The post-processing methods can be (but are not limited to) any one or any combination of following processes: 3x3 or 5x5 filtering, biasing, clipping, filtering or clipping like ALF or CCALF, SAO-like filtering, filter sets (e.g. ALF sets).

In one embodiment, the current block refers to luma, cb and/or cr. For example, when LM (e.g. proposed inverse LM described in a later section of this disclosure) is used to generate luma prediction, the post-processing is applied to luma. For another example, when CCLM is used to generate chroma prediction, the post-processing is applied to chroma.

In another embodiment, when the block size (width and/or height) is larger than a threshold, the post-processing is applied.

In another embodiment, the post-processing method of one component (e.g. cr) depends on another component (e.g. cb). For example, the selection of post-processing method for cb is derived according to signalling/bitstream and cr follows cb's selection. For another example, it is assumed that high correlation exists between cb and cr, so that the selection of post-processing method for cr is shown as follows:

- The cb prediction (without post-processing) plus cb residuals are treated as golden
- Choosing cr's post-processing method according to cb's post-processed prediction and golden
- For example, if the cb's post-processed prediction is very similar to the golden, use cb's post-processing method as cr's post-processing method.

Delta-pred LM

A novel LM method is proposed in this section. Different from the CCLM as disclosed earlier in the background section, the inputs of deriving model parameters are the predicted samples (used as X) for the first component and the delta samples (used as Y) between reconstructed and predicted samples for the first component. The derived parameters and the initial predicted samples of the second component can decide the current predicted samples of the second component. For example, the predictors of cb and cr can be calculated based on:

$$\text{delta_cb} = \alpha * \text{initial_pred_cb} + \beta, \text{pred_cb} = \text{initial_pred_cb} + \text{delta_cb},$$

$$\text{delta_cr} = \alpha * \text{initial_pred_cr} - \beta, \text{pred_cr} = \text{initial_pred_cr} + \text{delta_cr}.$$

For another example, the predictors of cb and cr can be calculated as:

$$\text{delta_cb} = \alpha * \text{initial_pred_cb} + \beta, \text{pred_cb} = \text{initial_pred_cb} + \text{delta_cb},$$

$$\text{delta_cr} = -\alpha * \text{initial_pred_cr} + \beta, \text{pred_cr} = \text{initial_pred_cr} + \text{delta_cr}.$$

Embodiments for pred-reco LM can be used for delta-pred LM.

Pred-reco LM

A novel LM method is proposed in this section. Different from the CCLM as disclosed earlier in the background section, the inputs of deriving model parameters are the predicted samples (used as X) for the first component and the reconstructed samples (used as Y) for the first component. The derived parameters and the initial predicted samples of the second component can decide the current predicted samples of the second

component. For example, the predictors of cb and cr can be calculated based on:

$$\text{Pred_cb} = \alpha * \text{initial_pred_cb} + \beta$$

$$\text{Pred_cr} = \alpha * \text{initial_pred_cr} - \beta$$

For another example, the predictors of cb and cr can be calculated as

$$\text{Pred_cb} = \alpha * \text{initial_pred_cb} + \beta,$$

$$\text{Pred_cr} = -\alpha * \text{initial_pred_cr} + \beta.$$

In one embodiment, the first component is luma and the second component is cb or cr.

In another embodiment, the first component is cb and the second component is cr.

In another embodiment, the first component is weighted cb and cr and the second component is luma, where inverse LM is applied. For example, the inputs of deriving model parameters are the weighted predictions of cb and cr and the weighted reconstructed samples of cb and cr.

In one sub-embodiment, the weight for (cb, cr) can be equal.

In another sub-embodiment, the weight for (cb, cr) can be (1, 3) or (3, 1). Take (3, 1) as an example, the weighting formula can be:

$$\text{weighted_pred} = (3 * \text{pred_cb} + 1 * \text{pred_cr} + \text{offset}) \gg 2,$$

$$\text{weighted_reco} = (3 * \text{reco_cb} + 1 * \text{reco_cr} + \text{offset}) \gg 2.$$

In another embodiment, the initial predicted samples of the second component are generated by chroma DM.

In another embodiment, the initial prediction samples of the second component are generated by one or more traditional intra prediction modes (e.g. angular intra prediction modes, DC, planar).

Joint LM

Different from CCLM as disclosed earlier in the background section, joint linear model is proposed to share a single model for chroma components (cb and cr).

In one embodiment, the parameters of the derived single model include alpha and beta. For example, the predictors of cb and cr can be calculated based on luma reconstructed samples and the parameters.

$$\text{Pred_cb} = \alpha * \text{reco_luma} + \beta,$$

$$\text{Pred_cr} = \alpha * \text{reco_luma} - \beta.$$

For another example, the predictors of cb and cr can be calculated as

$$\text{Pred_cb} = \alpha * \text{reco_luma} + \beta,$$

$$\text{Pred_cr} = -\alpha * \text{reco_luma} + \beta.$$

In another embodiment, when deriving model parameters, luma, cb, and cr are used. The luma parts are kept as original and the chroma parts are changed. For example, the cb's and cr's reconstructed neighbouring samples are weighted before being the inputs of deriving model parameters. The weighted method can be any one or any combination of the methods to be described in section JCCLM-method 1/-method 2.

In another embodiment, when deriving model parameters, luma and one of chroma components are used. For example, luma and cb are used to decide model parameters.

In another embodiment, instead of using neighbouring reconstructed samples, neighbouring residuals are used for deriving model parameters. Then, the joint residuals of cb and cr are derived as follows:

$$\text{resi}_C(i, j) = a * \text{resi}_L^{(i, j)} + b$$

In one sub-embodiment, if JCCR is applied, LM parameters for Cb and Cr are the same (i.e., joint LM is applied).

In another sub-embodiment, the neighbouring residuals for chroma are the weighted sum of neighbouring cb and cr residuals.

In another sub-embodiment, if joint LM is applied, JCCR is inferred as enabled.

In another sub-embodiment, when joint LM is used, the prediction of current chroma block is generated by chroma DM mode.

In another sub-embodiment, when joint LM is used, an initial prediction of current chroma block is generated by chroma DM mode and the final prediction of current chroma block is generated based on the initial prediction and resi_C . (e.g. initial prediction + resi_C)

Residual LM

Instead of using neighbouring reconstructed samples, neighbouring residuals are used for deriving model parameters. Then, the joint residuals of current chroma block are derived as follows. (cb and cr have their own models, respectively.)

$$\text{resi}_c(i, j) = a \cdot \text{resi}_L^{(i, j)} + b$$

5 In one embodiment, the prediction of current chroma block (denoted as pred_c) is generated by chroma DM and the reconstruction of current chroma block is formed by $\text{pred}_c + \text{resi}_c$.

In another embodiment, an initial prediction of current chroma block is generated by chroma DM mode and the final prediction of current chroma block is generated based on the initial prediction and resi_c . (e.g. initial prediction + resi_c).

10 JCCLM (JCCR with CCLM) – Method 1

JCCLM-mothod1 is proposed as a novel LM derivation scheme. Different from the CCLM as disclosed earlier in the background section, neighbouring luma reconstructed samples and weighted reconstructed neighbouring cb and cr samples are used as the inputs X and Y of model derivation. The derived model is called as JCCLM and the model parameters are called as JCCLM parameters in this disclosure. Then, JCCLM predictors are decided according to JCCLM parameters and reconstructed samples of the collocated luma block. Finally, the predictions for cb and cr are calculated by the JCCLM predictors.

In one embodiment, the weighting for generating weighted reconstructed neighbouring cb and cr samples can be (1, -1) for (cb, cr).

20 In another embodiment, the weighting for generating weighted reconstructed neighbouring cb and cr samples can be (1/2, 1/2) for (cb, cr).

In another embodiment, the predictions for cb and cr are calculated as follows:

$$\text{pred}_{cb} = 1 * \text{JCCLM_predictor}, \text{pred}_{cr} = -1 * \text{JCCLM_predictor} + k$$

In another sub-embodiment, k can be any positive value. For example, $k = 512$.

In another sub-embodiment, k varies with the bit depth. For example, if the bit depth is 10, $k = 512$.

25 In another sub-embodiment, k is pre-defined in the standard or depends on the signalling at block, SPS, PPS, and/or picture level.

In another embodiment, the predictions for cb and cr are calculated as follows:

$$\text{pred}_{cb} = 1 * \text{JCCLM_predictor}, \text{pred}_{cr} = 1 * \text{JCCLM_predictor}.$$

30 In another embodiment, when the weighting for generating weighted reconstructed neighbouring cb and cr samples is (1, -1) for (cb, cr), the predictions for cb and cr are calculated as follows:

$$\text{pred}_{cb} = 1 * \text{JCCLM_predictor}, \text{pred}_{cr} = -1 * \text{JCCLM_predictor} + k$$

In the above equation, the value of k can reference the sub-embodiments mentioned above. In another embodiment, when the weighting for generating weighted reconstructed neighbouring cb and cr samples is (1/2, 1/2) for (cb, cr), the predictions for cb and cr are calculated as follows.

35 $\text{pred}_{cb} = 1 * \text{JCCLM_predictor}, \text{pred}_{cr} = 1 * \text{JCCLM_predictor}$

In another embodiment, when JCCLM is applied, residual coding use JCCR automatically.

JCCLM (JCCR with CCLM) – Method 2

40 JCCLM-method 2 is proposed as a novel LM derivation scheme. Different from the CCLM as disclosed earlier in the background section, two models are used for generating prediction of the current block. The derivation process of the two models and their corresponding predictors are shown below:

- JCCLM: Neighbouring luma reconstructed samples and weighted reconstructed neighbouring cb and cr samples are used as the inputs X and Y of model derivation. The derived model is called as JCCLM and the model parameters are called as JCCLM parameters in this disclosure. Then, JCCLM predictors are decided according to JCCLM parameters and reconstructed samples of the collocated luma block.
- Cb_CCLM: Neighbouring luma reconstructed samples and neighbouring cb reconstructed samples are used as the inputs X and Y of model derivation. The derived model is called as cb_CCLM and the model parameters are called as cb_CCLM parameters in this disclosure. Then, cb_CCLM predictors are decided

according to cb_CCLM parameters and reconstructed samples of the collocated luma block.

Finally, the predictions for cb and cr are calculated by the JCCLM predictors and cb_CCLM predictors. Fig. 9 illustrates an example of the relationship between the cr prediction 910, cb prediction 920 and JCCLM predictors 930.

In one embodiment, the weighting for generating weighted reconstructed neighbouring cb and cr samples can be (1/2, 1/2) for (cb, cr).

In another embodiment, the prediction for cb is calculated as follows:

$$\text{pred_cb} = \text{cb_CCLM_predictors}.$$

In another embodiment, the prediction for cr is calculated as follows:

$$\text{pred_cr} = 2 * \text{JCCLM_predictor} - \text{cb_CCLM_predictor}$$

In another embodiment, when JCCLM is applied, residual coding uses JCCR automatically.

Multiple-hypothesis of CCLM Prediction

In addition to CCLM as disclosed earlier in the background section (for cb, deriving model parameters from luma and cb; for cr, deriving model parameters from luma and cr), more CCLM variations are disclosed. The following shows some examples.

- In one variation, cr prediction is derived by:
 - o Deriving model parameters by using neighbouring reconstructed samples of cb and cr as the inputs X and Y of model derivation
 - o Then generating cr prediction by the derived model parameters and cb reconstructed samples.
- In another variation, MMLM is used.
- In yet another variation, model parameters for cb (or cr) prediction are derived from multiple collocated luma blocks.

Each CCLM method is suitable for different scenarios. For some complex features, the combined prediction may result in better performance. Therefore, multiple-hypothesis CCLM is disclosed to blend the predictions from multiple CCLM methods. The to-be-blended CCLM methods can be from (but are not limited to) the above mentioned CCLM methods. A weighting scheme is used for blending.

In one embodiment, the weights for different CCLM methods are pre-defined at encoder and decoder.

In another embodiment, the weights vary based on the distance between the sample (or region) positions and the reference sample positions.

In another embodiment, the weights depend on the neighbouring coding information.

In another embodiment, a weight index is signalled/parsed. The code words can be fixed or vary adaptively. For example, the code words vary with template-based methods.

Adaptive Intra-mode Selection

With improvement of video coding, more coding tools are created. The syntax overhead of selecting a coding tool becomes an issue. Several straightforward methods can be used to reduce the syntax overhead. For example, a large block can use the same coding mode. In another example, multiple components (e.g. cb and cr) can share the same coding mode.

However, with these straightforward methods, the accuracy/performance for intra prediction decreases. The possible reasons may be following:

- Intra prediction is highly related to neighbouring reference samples. When the whole block uses a single intra prediction mode, the intra prediction mode may be suitable for those samples which are close to the reference samples but may not be good for those samples which are far away from the reference samples.
- when processing cr, the reconstructions of cb and luma were generated and can be used to choose the coding mode for cr.

In this section, it is proposed to adaptively change the intra prediction mode for one or more sample(s) or subblock(s) within the current block according to previous coding/decoding of components.

In one embodiment, with the reconstruction of the previously encoded/decoded components, the performance for the different coding modes is decided. Then, the better mode is used for the rest component(s) (subsequently encoded and decoded component(s)). For example, for cb, if the prediction from traditional intra prediction modes

(e.g. angular intra prediction modes, DC, planar) is better than the prediction from LM mode. (e.g. “better” means similar to cb’s reconstruction.) Then, the traditional intra prediction mode is preferable for cr.

In one sub-embodiment, the proposed method can be subblock based. For example, a chroma block is divided into several sub-blocks. For each subblock, if for cb, the subblock’s prediction from LM mode is better than the subblock’s prediction from traditional intra prediction modes (e.g. angular intra prediction modes, DC, planar). (e.g. “better” means similar to cb’s reconstruction and reducing the cb’s residual), then the LM mode is preferable for the corresponding subblock of cr. An example is shown in Fig. 10, where the chroma block is divided into 4 sub-blocks. If sub-blocks 1 and 2 of cb block 1010 have better prediction results using LM mode, then sub-blocks 1 and 2 of cr block 1020 also use LM mode.

In another embodiment, the adaptive changing rule can be performed at both encoder and/or decoder and doesn’t need an additional syntax.

Inverse LM

For the CCLM mode as disclosed earlier in the background section, luma reconstructed samples are used to derive the predictors in the chroma block. In this disclosure, inverse LM is proposed to use chroma information to derive the predictors in the luma block. When supporting inverse LM, chroma are encoded/decoded (signalled/parsed) before luma.

In one embodiment, the chroma information refers to the chroma reconstructed samples. When deriving model parameters for inverse LM, reconstructed neighbouring chroma samples are used as X and reconstructed neighbouring luma samples are used as Y. Moreover, the reconstructed samples in the chroma block (collocated to the current luma block) and the derived parameters are used to generate the predictors in the current luma block. An alternative way is that “information” in this embodiment can refer to predicted samples.

In one embodiment, chroma refers to cb and/or cr component(s).

In one sub-embodiment, only one of cb’s and cr’s information is used.

In another sub-embodiment, the chroma information is from both cb and cr. For example, the neighbouring reconstructed cb and cr samples are weighted and then used as the inputs of deriving model parameters. In another example, the reconstructed cb and cr samples in the chroma block (collocated with the current luma block) are weighted and then used to derive the predictors in the current luma block.

In another embodiment, for the current luma block, the prediction (generated by the proposed inverse LM) can be combined with one or more hypotheses of predictions (generated by one or more other intra prediction modes).

In one sub-embodiment, “other intra prediction modes” can refer to angular intra prediction modes, DC, planar, MIP, ISP, MRL, any other existing intra modes (supported in HEVC/VVC) and/or any other intra prediction modes.

In another sub-embodiment, when combining multiple hypotheses of predictions, weighting for each hypothesis can be fixed or adaptively changed. For example, equal weights are applied to each hypothesis. In another example, weights vary with neighbouring coding information, sample position, block width, height, prediction mode or area. Some examples of neighbouring coding information usage are shown as follows:

- One possible rule related to sample position is described as follows.
 - o When the sample position is further away from the reference samples, the weight for the prediction from other intra prediction modes decreases.
- Another possible rule related to neighbouring coding information is described as follows.
 - o When more neighbouring blocks (left, above, left-above, right-above, and/or left-bottom) are coded with a particular (e.g. Mode A), the weight for the prediction from Mode A gets higher.
- Another possible rule related to sample position is described as follows.
 - o The current block is partitioned into several regions. The sample positions in the same region share the same weighting. If the current region is close to the reference L neighbour, the weight for prediction from other intra prediction modes is higher than the weight for prediction from CCLM. The following shows some possible ways to partition the current block. (as the dotted lines in the Figs. 11A-C):

- Fig. 11A (ratio of width and height close to or exactly 1:1): The distance between the current region and the left and top reference L neighbour is considered.
- Fig. 11B (width > n* height, where n can be any positive integer): The distance between the current region and the top reference L neighbour is considered.
- Fig. 11C (height > n* width, where n can be any positive integer): The distance between the current region and the left reference L neighbour is considered.

CCLM for Inter Block

In the overview section, CCLM is used for intra blocks to improve chroma intra prediction. For an inter block, chroma prediction may be not as accurate as luma. Possible reasons are listed below:

- Motion vectors for chroma components are inherited from luma, (Chroma doesn't have its own motion vectors.)
- Less coding tools are designed to improve inter chroma prediction.

Therefore, CCLM is proposed as an alternative way to code inter blocks. With this proposed method, chroma prediction according to luma for an inter block can be improved. According to CCLM for inter block, the corresponding luma block is coded in the inter mode, i.e., using motion compensation and one or more motion vectors to access previous reconstructed luma blocks in one or more previously coded reference frames. Cross-colour linear mode based on this inter-coded luma may provide better prediction than the inter prediction based on previous reconstructed chroma blocks in one or more previously coded reference frames. The CCLM for intra mode has been described in the background section. The CCLM process described earlier can be applied here. However, while the conventional CCLM utilizes a reconstructed luma block in the same frame as the chroma block, CCLM inter mode utilizes a reconstructed or predicted luma block derived from the reconstructed luma blocks in one or more previously coded reference frames.

In one embodiment, for chroma components, in addition to original inter prediction (generated by motion compensation), one or more hypotheses of predictions (generated by CCLM and/or any other LM modes) are used to form the current prediction.

In one sub-embodiment, the current prediction is the weighted sum of inter prediction and CCLM prediction. Weights are designed according to neighbouring coding information, sample position, block width, height, mode or area. Some examples are shown as follows:

- In one example, for a small block (e.g. area < threshold), weights for CCLM prediction are higher than weights for inter prediction.
- In another example, when most neighbouring coded blocks are intra blocks, weights for CCLM prediction are higher than weights for inter prediction.
- In yet another example, weights are fixed values for the whole block.

In another embodiment, original inter prediction (generated by motion compensation) is used for luma and the predictions of chroma components are generated by CCLM and/or any other LM modes.

In one sub-embodiment, the current CU is viewed as an inter CU, intra CU, or a new type of prediction mode (neither intra nor inter).

The above proposed methods can be also applied to IBC blocks. ("inter" in this section can be changed to IBC.) That is, for chroma components, the block vector prediction can be combined or replaced with CCLM prediction.

Cross-CU LM

Compared with traditional intra prediction modes (e.g. angular intra prediction modes, DC, and planar), the benefit from LM mode is to predict irregular patterns as shown in Fig. 12, where the block has an irregular pattern that no angular intra prediction can provide a good prediction. However, the luma block 1210 can provide a good prediction for the chroma block 1220 using LM mode.

For encoding/decoding of irregular patterns in an inter picture, the distribution of intra and inter coding modes may look as follows. For some regions (highly related to neighbour), intra mode is used. For other regions, inter mode is preferable.

To handle the situation shown as above, a cross-CU LM mode is proposed. Based on the observation of

current CU's ancestor node, LM mode is applied. For example, if the ancestor node contains irregular patterns (e.g. partial intra with partial inter), the blocks belonging to this ancestor node are encoded/decoded with LM mode. With the proposed method, the CU-level on/off flag for LM mode is not required. Fig. 13 illustrates an example that a luma picture area associated with a node contains irregular patterns. The area associated with the node is partitioned into luma blocks according to the irregular patterns. The luma blocks (the dashed-line blocks) that the irregular patterns occupy a noticeable portion of the blocks processed as intra blocks; and otherwise the luma blocks (the dotted-line blocks) are processed as inter luma blocks.

In one embodiment, the block-level on/off flag for LM mode is defined/signalled at the ancestor node level. For example, when the flag at the ancestor node indicates the cross-CU LM is enabled, the CUs belongs to (i.e., those partitioned from) the ancestor node use LM. In another example, when the flag at the ancestor node indicates the cross-CU LM is disabled, the CUs belongs to (i.e., those partitioned from) the ancestor node do not use LM.

In another embodiment, the ancestor node refers to a CTU.

In another embodiment, whether to enable cross-CU LM is implicitly derived according to the analysis of ancestor node's block properties.

In this section, CU can be changed to any block. For example, it can be PU.

LM assisted Angular/Planar Mode

For traditional intra prediction modes (e.g. angular intra prediction modes, chroma DM, DC, planar and/or an intra prediction mode, which is not a cross-component mode), the reference samples are from top and left neighbouring reconstructed samples. Therefore, the accuracy of intra prediction decreases for right-bottom samples within the current block. In this section, one or more cross-component modes are used to improve the prediction from traditional intra prediction modes.

In one embodiment, the current block's prediction is formed by a weighted sum of one or more hypotheses of predictions from traditional intra prediction mode(s) and one or more hypotheses of predictions from cross-component mode(s). In the following examples, the cross-component mode(s) refer to one or more LM mode(s) and/or extensions/variations of CCLM. In one sub-embodiment, equal weights are applied to both. In another sub-embodiment, weights vary with neighbouring coding information, sample position, block width, height, mode or area. For example, when the sample position is far away from the top-left region, the weight for the prediction from traditional intra prediction modes decreases. More weighting schemes can reference "inverse LM" section. The following shows some examples.

- One possible rule related to sample position is described as follows.
 - o When the sample position is further away from the reference samples, the weight for the prediction from other intra prediction modes decreases.
- Another possible rule related to neighbouring coding information is described as follows.
 - o When more neighbouring blocks (left, above, left-above, right-above, and/or left-bottom) are coded with a particular (e.g. Mode A), the weight for the prediction from Mode A gets higher. For example, Mode A refers to a specific cross-component mode such as CCCM_LT or a specific cross-component family such as CCCM family (including CCCM_LT, CCCM_L, and/or CCCM_T). A weighting set is pre-defined to include multiple weighting candidates such as {1, 3}, {3, 1}, and/or equal weighting {2, 2} for one prediction from a traditional intra prediction mode and the other prediction from a cross-component mode. When all or most of the neighbouring blocks are coded by Mode A, the weighting candidate with a higher weight for the prediction from a cross-component mode is used. When only partial or one of the neighbouring blocks are coded by Mode A, the weighting candidate with an equal weighting is used. When few or none of the neighbouring blocks are coded by Mode A, the weighting candidate with a smaller weight for the prediction from a cross-component mode is used. For example, the neighbouring blocks include any subset of the coded blocks which are spatially adjacent to the top boundary or left boundary of the current block. In this case, the neighbouring blocks may refer to the top neighbouring block (located on the top of the top-right corner of the current block) and the left neighbouring block

(located on the left of the bottom-left corner of the current block). For another example, the neighbouring blocks include any subset of the coded blocks located in a pre-defined range spatially nearing the top boundary or left boundary of the current block. In this case, the neighbouring blocks can be adjacent or non-adjacent to the current block.

- Another possible rule related to sample position is described as follows.

- The current block is partitioned into several regions. The sample positions in the same region share the same weighting. If the current region is close to the reference L neighbour, the weight for prediction from other intra prediction modes is higher than the weight for prediction from CCLM. The following shows some possible ways to partition the current block. (as the dotted lines in the Figs. 11A-C):

- Fig. 11A (ratio of width and height close to or exactly 1:1): The distance between the current region and the left and top reference L neighbour is considered.
- Fig. 11B (width > n* height, where n can be any positive integer): The distance between the current region and the top reference L neighbour is considered.
- Fig. 11C (height > n* width, where n can be any positive integer): The distance between the current region and the left reference L neighbour is considered.

In another embodiment, it is proposed to use LM mode to generate the right-bottom region within or near the current block. When doing intra prediction, the reference samples can be based on not only original left and top neighbouring reconstructed samples but also proposed right and bottom LM-predicted samples. The following shows an example.

- Before doing intra prediction for a chroma block, the collocated luma block is reconstructed.
- “The neighbouring luma reconstructed samples of the collocated luma block” and “the neighbouring chroma reconstructed samples of the current chroma block” are used for deriving LM parameters.
- “The reconstructed samples of the collocated luma block” with the derived parameters are used for obtaining the right-bottom LM-predicted samples of the current chroma block. Right-bottom region of the current chroma block can be any subset of the region in Figs. 14A-B. Fig. 14A illustrates an example where the right-bottom region 1414 is outside the current chroma block 1410. Fig. 14B illustrates an example where the right-bottom region 1422 is outside the current chroma block 1420.
- The prediction of the current block is generated bi-directionally by referencing original L neighbouring region(original top and left region, obtained using a traditional intra prediction mode) and the proposed inverse-L region (obtained using LM).

In one sub-embodiment, the predictors from the original top and left region and the predictors from bottom and left region are combined with weighting. In one example, equal weights are applied to both. In another example, weights vary with neighbouring coding information, sample position, block width, height, mode or area. For example, when the sample position is far from the top and left region, the weight for the prediction from the traditional intra prediction mode decreases.

In another embodiment, this proposed method can be applied to inverse LM. Then, when doing luma intra prediction, the final prediction is bi-directional, which is similar to the above example for a chroma block.

In another embodiment, after doing segmentation to know the curve pattern for luma, the proposed LM assisted Angular/Planar Mode assists chroma with getting the correct curved angle.

The proposed methods in this disclosure can be enabled and/or disabled according to implicit rules (e.g. block width, height, or area) or according to explicit rules (e.g. syntax in block, slice, picture, SPS, or PPS level). For example, when the LM assisted Angular/Planar Mode is used to improve one or more traditional intra prediction modes and the enabling conditions of the improvement are satisfied, a block-level flag is signalled/parsed to indicate whether to apply the improvement on the one or more traditional intra prediction modes. When the enabling conditions of the improvement are not satisfied, the syntax for the improvement is not signalled/parsed and/or is inferred as disabled. The enabling conditions may include (1) one or more intra prediction modes for the current block belonging to a pre-defined set of intra prediction modes, (2) the block size of the current block larger than a pre-defined threshold, (3) the current slice type equal to a predefined type such

as intra or inter, and/or (4) the type of the current splitting tree equal to single tree or dual tree. Therefore, in one way, the block-level flag for the improvement may be signalled/parsed after the one or more traditional intra prediction modes are decided to be used for the current block. In another way, the block-level flag for the improvement may be signalled/parsed before the one or more traditional intra prediction modes are decided to be used for the current block. The signalling and/or operations for the improvement can be the same, unified, or different for a set of traditional intra prediction modes available for the improvement. In one way of using different rules of the improvement, if an intra mode, which decides its intra prediction mode by an implicit derivation (such as calculating distortion between the predicted samples and reconstructed samples on the neighbouring region of the current block or calculating histogram or gradient values for the reconstructed samples on the neighbouring region of the current block) instead of signalled syntax, is used for the current block, the improvement may or may not be available for this intra mode. If the current slice type is an inter slice, the available intra prediction modes for the improvement may not be the same as intra slices. In another way of using unified rules of the improvement, for all intra prediction modes available for the improvement, the signalling and/or operations for the improvement are unified regardless of the slice type.

The term “block” in this disclosure can refer to TU/TB, CU/CB, PU/PB, or CTU/CTB.

The term “LM” in this disclosure can be viewed as one kind of CCLM/MMLM modes or any other extension/variation of CCLM (e.g. the proposed CCLM extension/variation in this disclosure). The variations of CCLM here mean that some optional modes can be selected when the block indication refers to using one of cross-component modes (e.g. CCLM_LT, MMLM_LT, CCLM_L, CCLM_T, MMLM_L, MMLM_T, and/or an intra prediction mode, which is not one of traditional DC, planar, and angular modes) for the current block. The following shows an example of being convolutional cross-component mode (CCCM) as an optional mode. When this optional mode is applied to the current block, cross-component information with a model, including non-linear term, is used to generate the chroma prediction. The optional mode may follow the template selection of CCLM, so CCCM family includes CCCM_LT CCCM_L, and/or CCCM_T.

The proposed methods (for CCLM) in this disclosure can be used for any other LM modes.

Any combination of the proposed methods in this disclosure can be applied.

Any of the foregoing proposed methods can be implemented in encoders and/or decoders. For example, any of the proposed methods can be implemented in an intra coding module (e.g. Intra Pred. 110 in Fig. 1A) of an encoder, an intra decoding module (e.g., Intra Pred. 150 in Fig. 1B), a merge candidate derivation module of a decoder. Alternatively, any of the proposed methods can be implemented as a circuit coupled to the intra coding module of an encoder and/or intra decoding module, a merge candidate derivation module of the decoder.

Fig. 15 illustrates a flowchart of an exemplary video coding system that utilizes cross-colour linear model for intra mode according to an embodiment of the present invention. The steps shown in the flowchart may be implemented as program codes executable on one or more processors (e.g., one or more CPUs) at the encoder side. The steps shown in the flowchart may also be implemented based hardware such as one or more electronic devices or processors arranged to perform the steps in the flowchart. According to this method, input data associated with a current block comprising at least one colour block in step 1510, wherein the input data comprise pixel data for the current block to be encoded at an encoder side or encoded data associated with said at least one colour block to be decoded at a decoder side, and wherein the current block is coded in an intra block mode. A blending predictor is determined according to a weighted sum of at least two candidate predictions generated based on one or more first hypotheses of prediction, one or more second hypotheses of prediction, or both in step 1520, wherein said one or more first hypotheses of prediction are generated based on one or more intra prediction modes comprising a DC mode, a planar mode or at least one angular modes, and said one or more second hypotheses of prediction are generated based on one or more cross-component modes and a collocated block of said at least one colour block. The input data associated with said at least one colour block is encoded using the blending predictor at the encoder side or the input data associated with said at least one colour block is decoded using the blending predictor at the decoder side in step 1530.

The flowchart shown is intended to illustrate an example of video coding according to the present invention. A person skilled in the art may modify each step, re-arranges the steps, split a step, or combine steps to practice

the present invention without departing from the spirit of the present invention. In the disclosure, specific syntax and semantics have been used to illustrate examples to implement embodiments of the present invention. A skilled person may practice the present invention by substituting the syntax and semantics with equivalent syntax and semantics without departing from the spirit of the present invention.

5 The above description is presented to enable a person of ordinary skill in the art to practice the present invention as provided in the context of a particular application and its requirement. Various modifications to the described embodiments will be apparent to those with skill in the art, and the general principles defined herein may be applied to other embodiments. Therefore, the present invention is not intended to be limited to the particular embodiments shown and described, but is to be accorded the widest scope consistent with the principles
10 and novel features herein disclosed. In the above detailed description, various specific details are illustrated in order to provide a thorough understanding of the present invention. Nevertheless, it will be understood by those skilled in the art that the present invention may be practiced.

Embodiment of the present invention as described above may be implemented in various hardware, software codes, or a combination of both. For example, an embodiment of the present invention can be one or more circuit
15 circuits integrated into a video compression chip or program code integrated into video compression software to perform the processing described herein. An embodiment of the present invention may also be program code to be executed on a Digital Signal Processor (DSP) to perform the processing described herein. The invention may also involve a number of functions to be performed by a computer processor, a digital signal processor, a
20 microprocessor, or field programmable gate array (FPGA). These processors can be configured to perform particular tasks according to the invention, by executing machine-readable software code or firmware code that defines the particular methods embodied by the invention. The software code or firmware code may be developed in different programming languages and different formats or styles. The software code may also be compiled for different target platforms. However, different code formats, styles and languages of software codes and other
25 means of configuring code to perform the tasks in accordance with the invention will not depart from the spirit and scope of the invention.

The invention may be embodied in other specific forms without departing from its spirit or essential characteristics. The described examples are to be considered in all respects only as illustrative and not restrictive. The scope of the invention is therefore, indicated by the appended claims rather than by the foregoing description. All changes which come within the meaning and range of equivalency of the claims are to be embraced within
30 their scope.

CLAIMS

1. A method of intra prediction for colour pictures, the method comprising:
 - receiving input data associated with a current block comprising at least one colour block, wherein the input data comprise pixel data for the current block to be encoded at an encoder side or encoded data associated with said at least one colour block to be decoded at a decoder side, and wherein the current block is coded in an intra block mode;
 - determining a blending predictor according to a weighted sum of at least two candidate predictions generated based on one or more first hypotheses of prediction, one or more second hypotheses of prediction, or both, wherein said one or more first hypotheses of prediction are generated based on one or more intra prediction modes comprising a DC mode, a planar mode or at least one angular modes, and said one or more second hypotheses of prediction are generated based on one or more cross-component modes and a collocated block of said at least one colour block; and
 - encoding the input data associated with said at least one colour block using the blending predictor at the encoder side or decoding the input data associated with said at least one colour block using the blending predictor at the decoder side.
2. The method of Claim 1, wherein said one or more second hypotheses of prediction correspond to CCLM (Cross-Colour Linear Model) prediction, wherein model parameters a and b for the CCLM prediction are determined based on neighbouring reconstructed pixels of the collocated block and neighbouring reconstructed pixels of said at least one colour block, and a pixel value for said at least one colour block is predicted according to $a \cdot rec_L' + b$ and rec_L' corresponds to one down-sampled pixel value for the collocated block.
3. The method of Claim 1, wherein said one or more second hypotheses of prediction correspond to MMLM (Multiple Model CCLM) and one of two models of CCLM (Cross-Colour Linear Model) prediction is selected according to a reconstructed pixel value for the collocated block.
4. The method of Claim 1, wherein equal weights are used for the weighted sum of said at least two candidate predictions.
5. The method of Claim 1, wherein weights for the weighted sum of said at least two candidate predictions are determined according to neighbouring coding information, sample position, block width, block height, block area, coding mode or a combination thereof.
6. The method of Claim 5, wherein, when more neighbouring blocks of the current block are coded with a target mode, a larger weight is used for one hypothesis of prediction associated with the target mode.
7. The method of Claim 6, wherein the target mode corresponds to one cross-component mode.
8. The method of Claim 5, wherein the current block is partitioned into multiple regions and sample positions in a same region share a same weight.
9. The method of Claim 8, wherein if a current region is close to L-shaped reference neighbours, one first hypotheses of prediction uses a higher weight than one second hypotheses of prediction.
10. The method of Claim 1, wherein said at least two candidate predictions comprise at least one first hypothesis of prediction and at least one second hypothesis of prediction.
11. The method of Claim 1, wherein the said blending predictor is only applied to a region within the current block.
12. The method of Claim 11, wherein the region corresponds to a right-bottom region of the current block.
13. The method of Claim 1, wherein said at least two candidate predictions correspond to at least two second hypotheses of prediction.
14. The method of Claim 1, wherein said at least one colour block corresponds to a Cr block and the collocated block corresponds to a Cb block, or said at least one colour block corresponds to the Cb block and the collocated block corresponds to the Cr block.

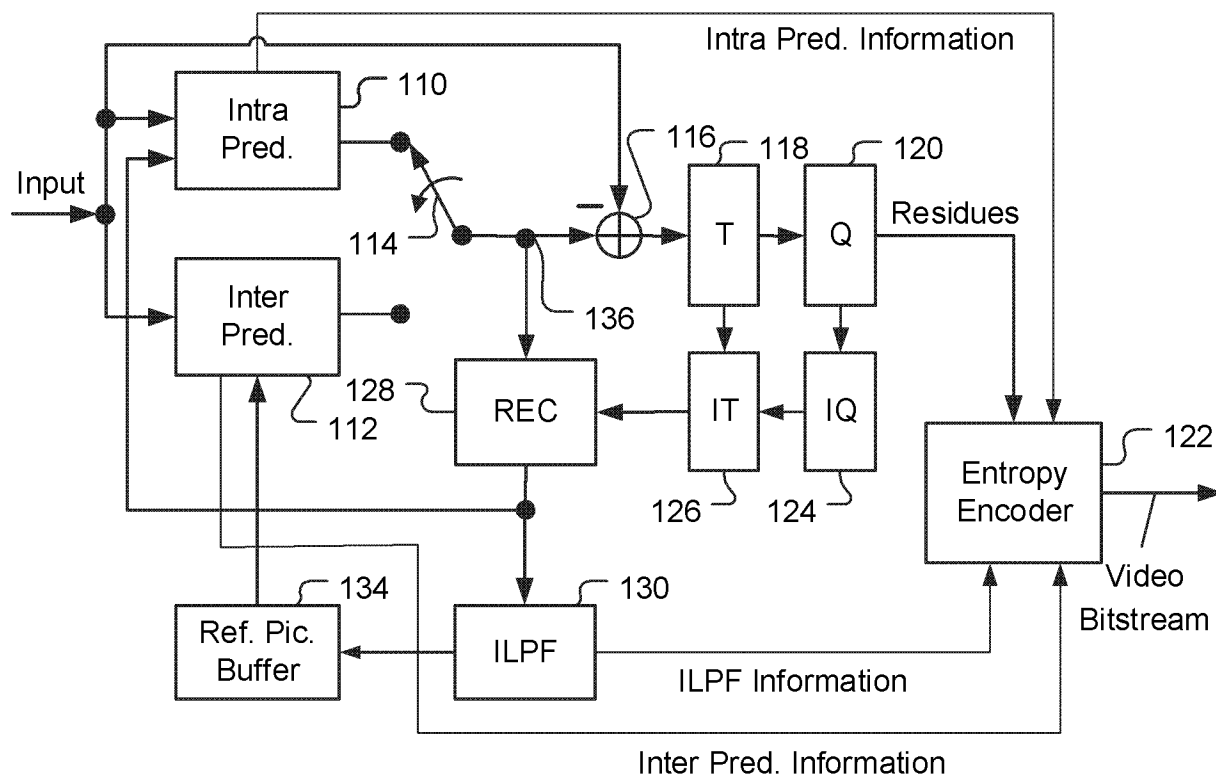
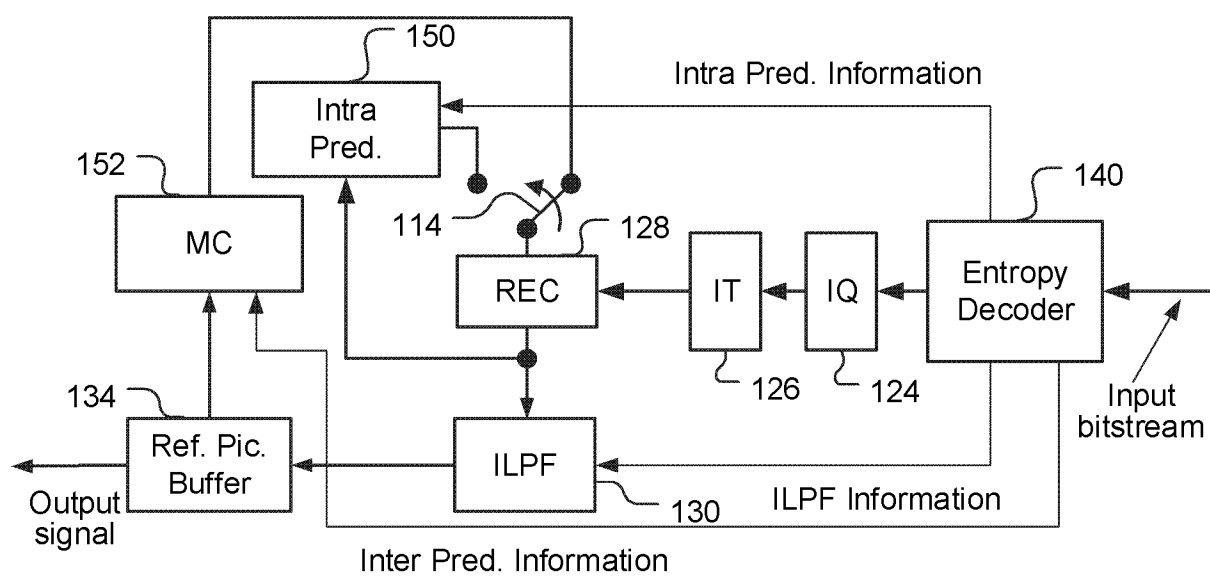
15. The method of Claim 13, wherein said at least two second hypotheses of prediction are generated from one or more pre-defined cross-component modes.

16. An apparatus for intra prediction for colour pictures, the apparatus comprising one or more electronics or processors arranged to:

5 receive input data associated with a current block comprising at least one colour block, wherein the input data comprise pixel data for the current block to be encoded at an encoder side or encoded data associated with said at least one colour block to be decoded at a decoder side, and wherein the current block is coded in an intra block mode;

10 determine a blending predictor according to a weighted sum of at least two candidate predictions generated based on one or more first hypotheses of prediction, one or more second hypotheses of prediction, or both, wherein said one or more first hypotheses of prediction are generated based on one or more intra prediction modes comprising a DC mode, a planar mode or at least one angular modes, and said one or more second hypotheses of prediction are generated based on one or more cross-component modes and a collocated block of said at least one colour block; and

15 encode the input data associated with said at least one colour block using the blending predictor at the encoder side or decode the input data associated with said at least one colour block using the blending predictor at the decoder side.

**Fig. 1A****Fig. 1B**

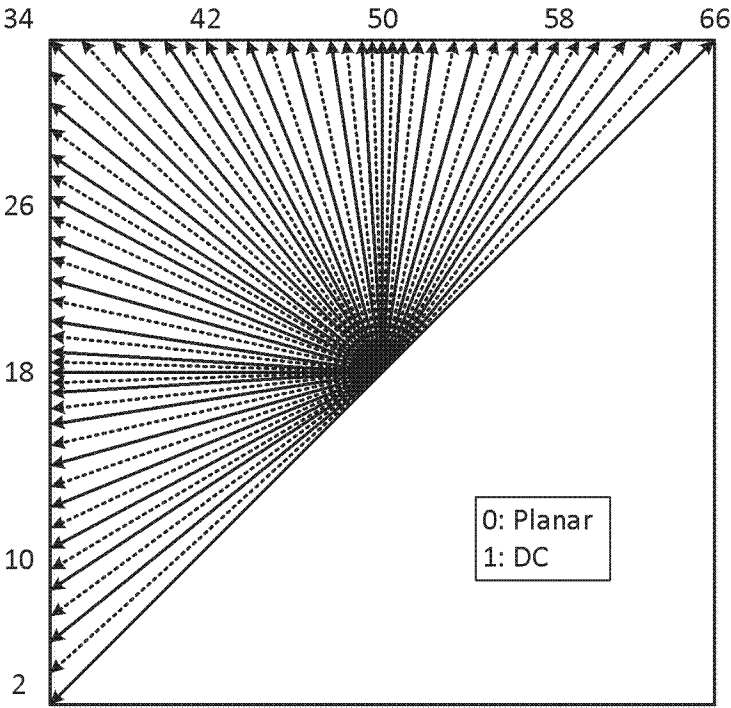


Fig. 2

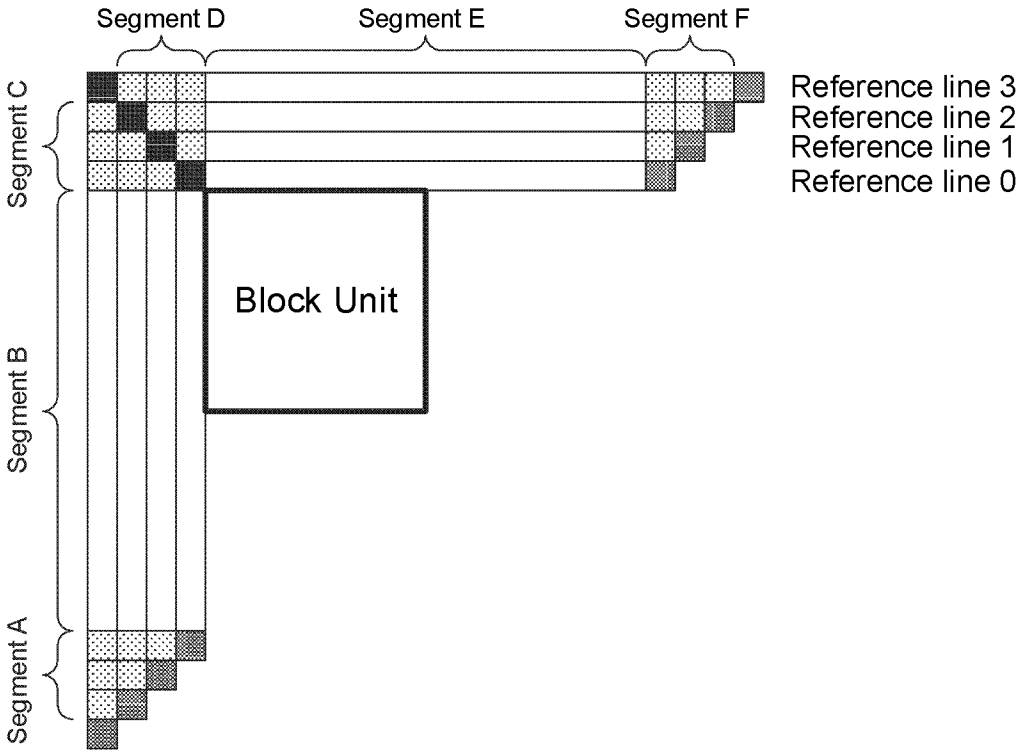


Fig. 3

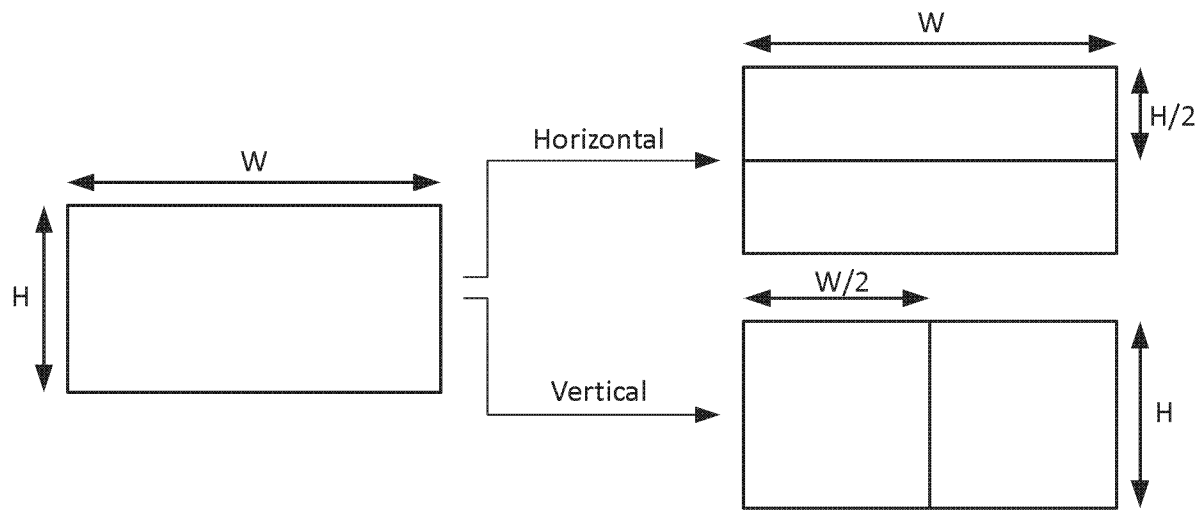


Fig. 4A

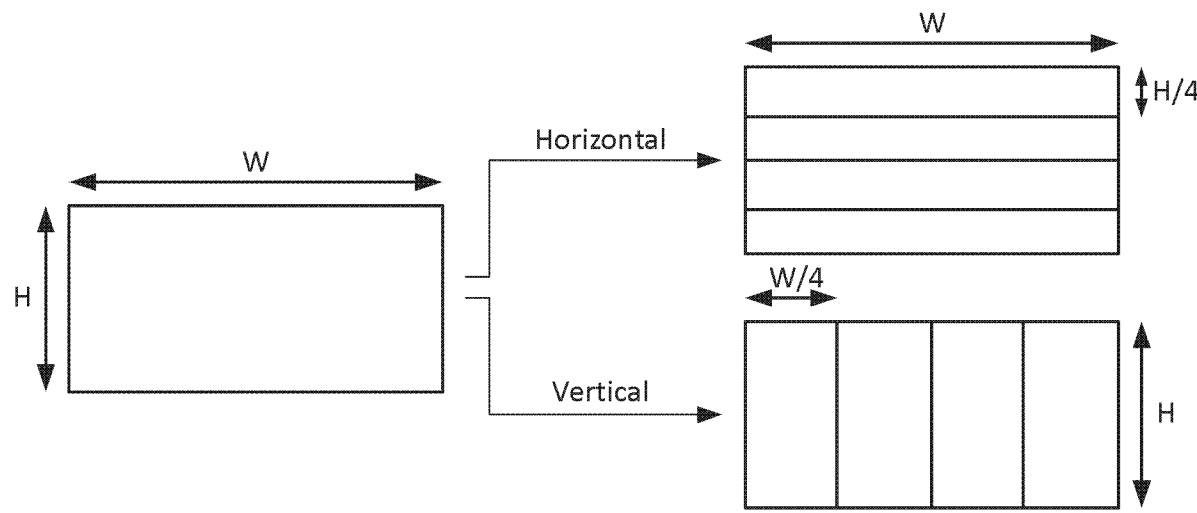


Fig. 4B

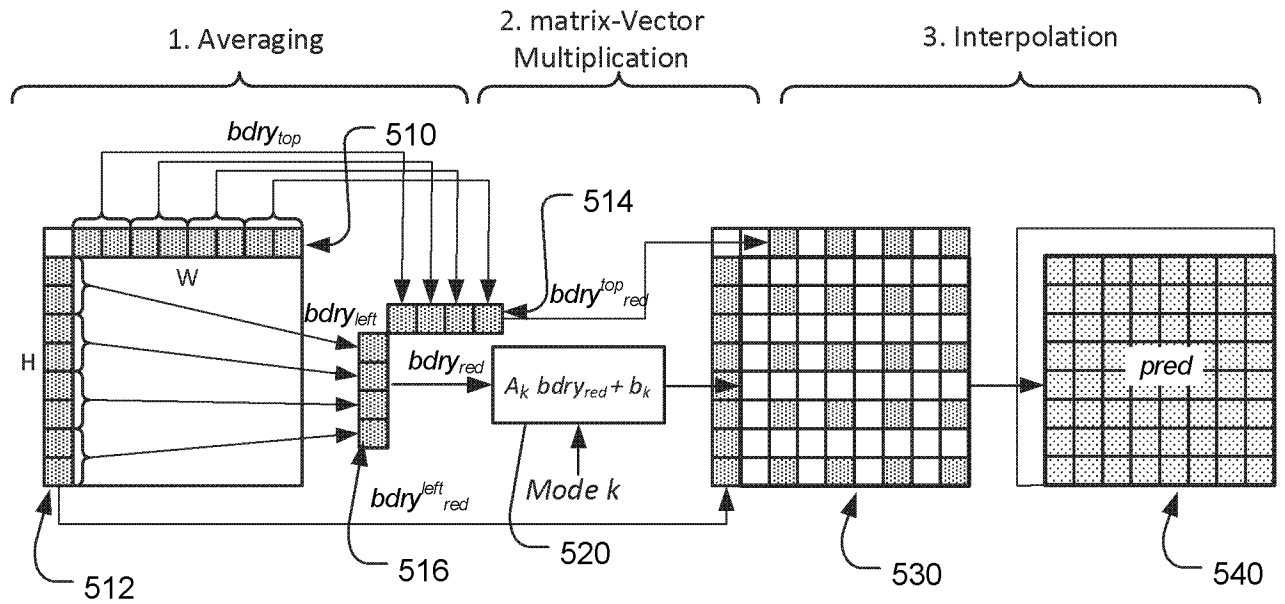


Fig. 5

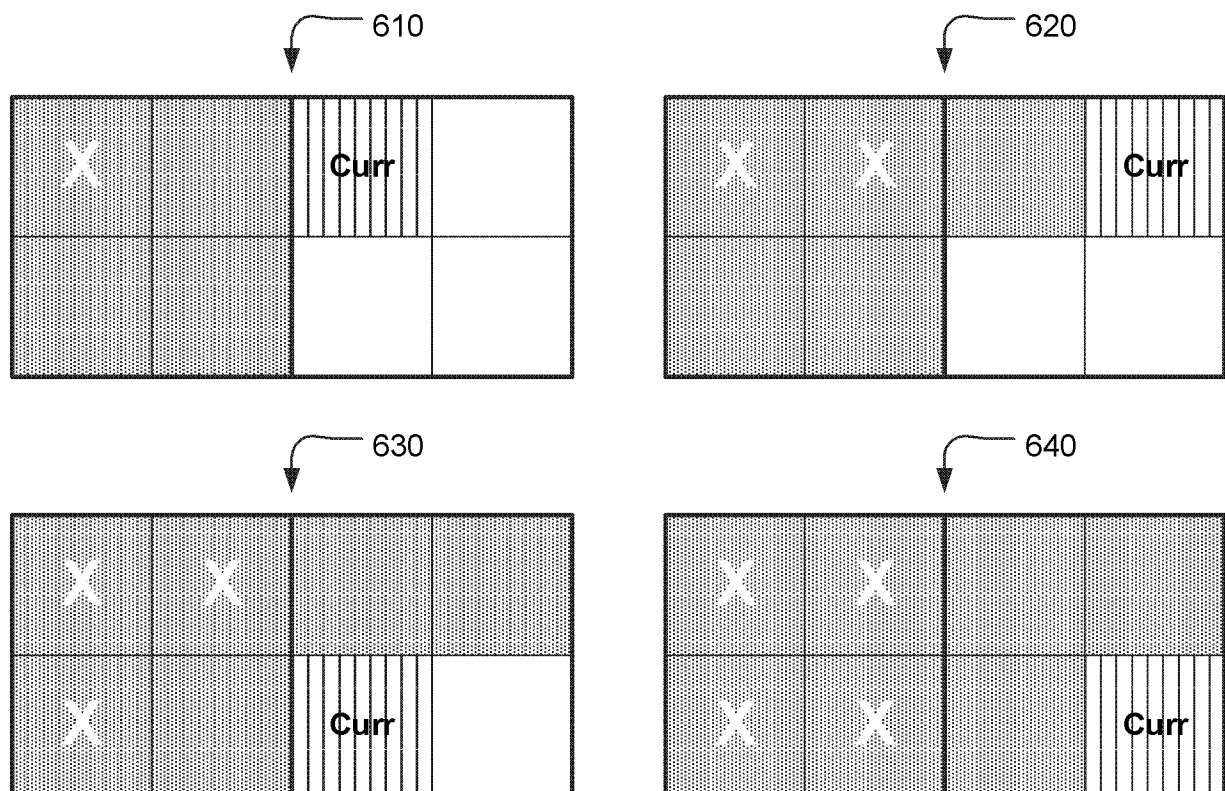


Fig. 6

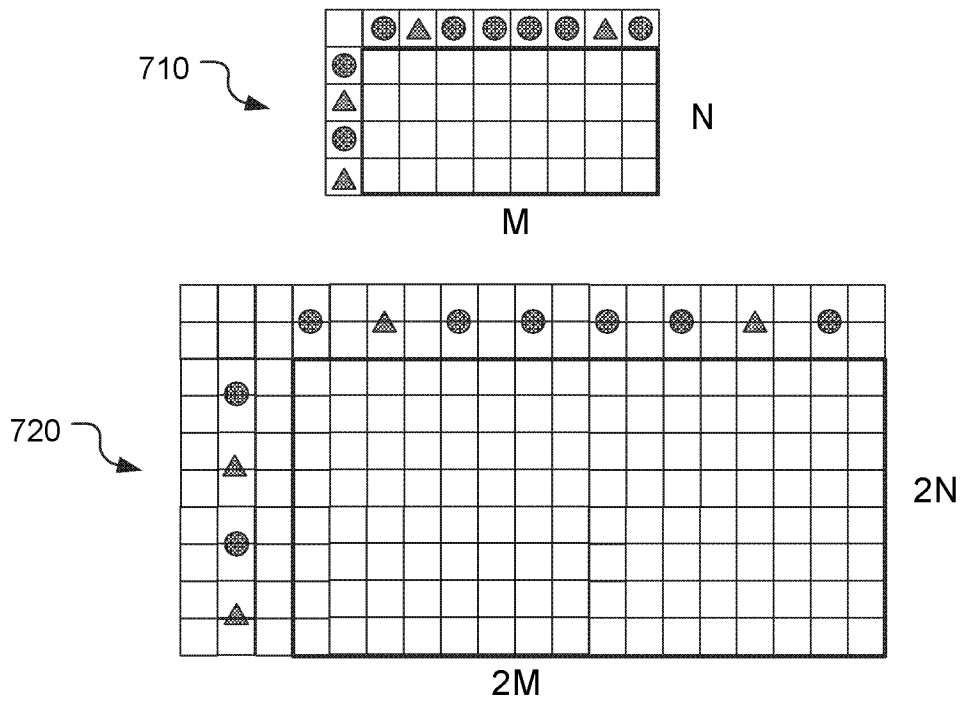


Fig. 7

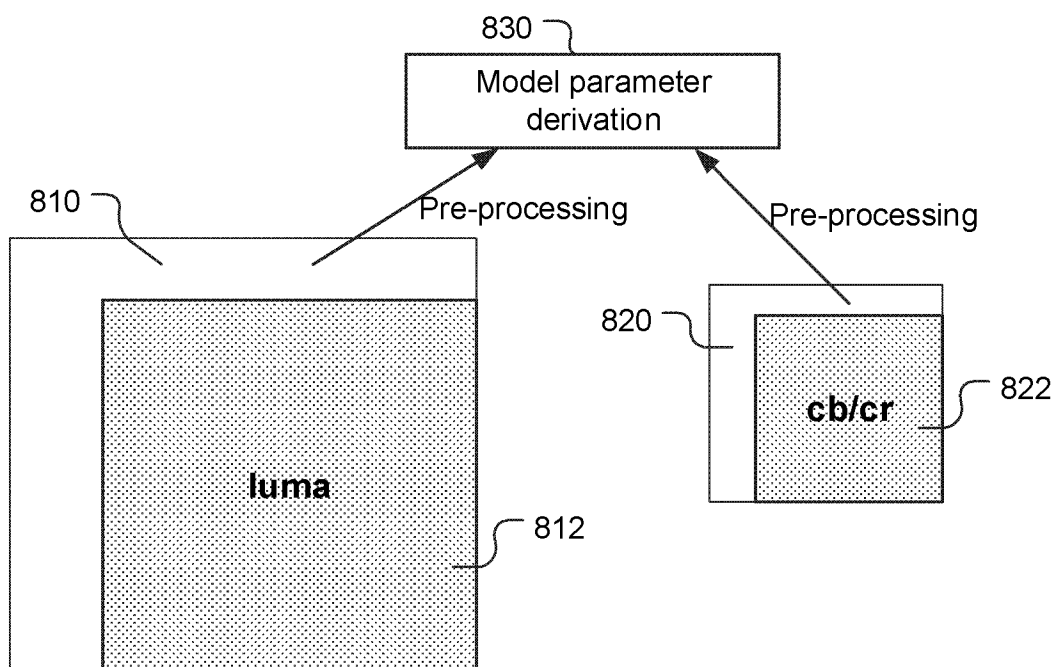


Fig. 8

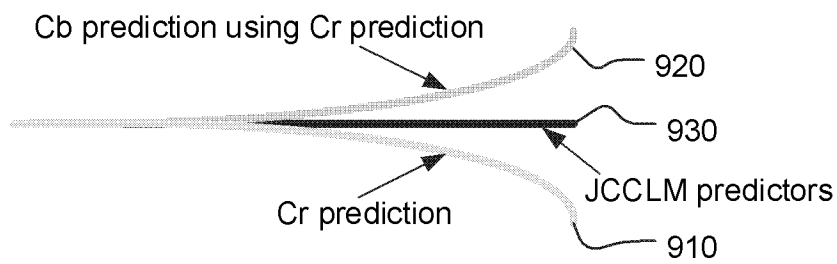


Fig. 9

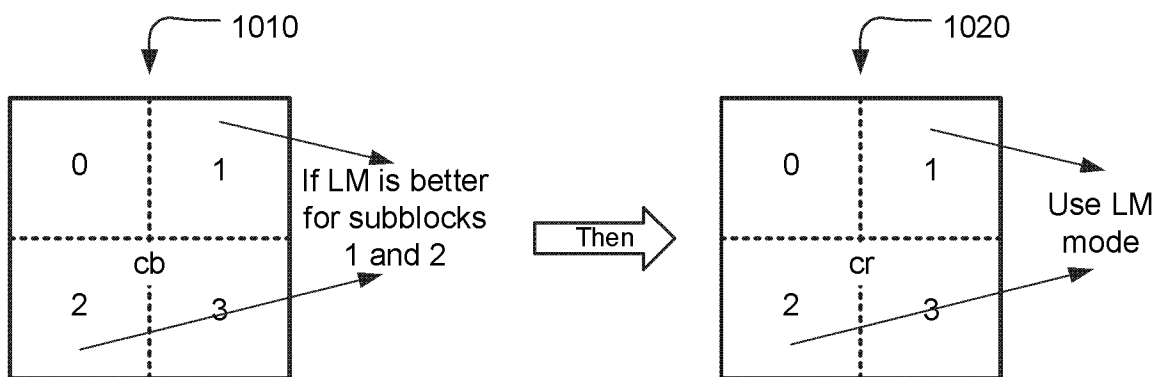


Fig. 10

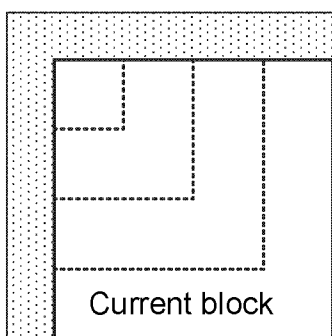


Fig. 11A

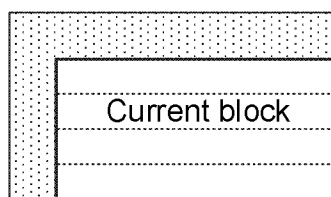


Fig. 11B

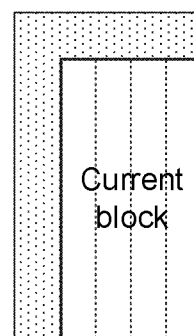


Fig. 11C

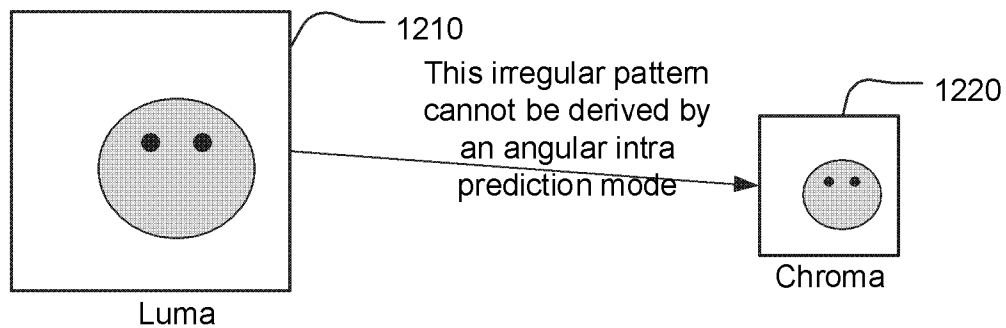


Fig. 12

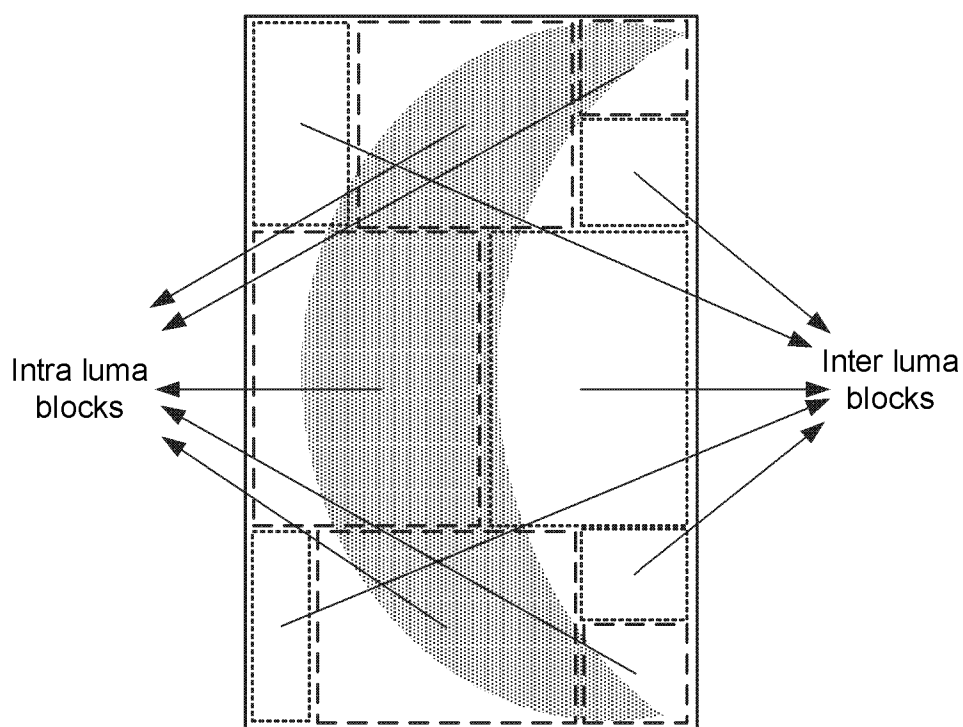


Fig. 13

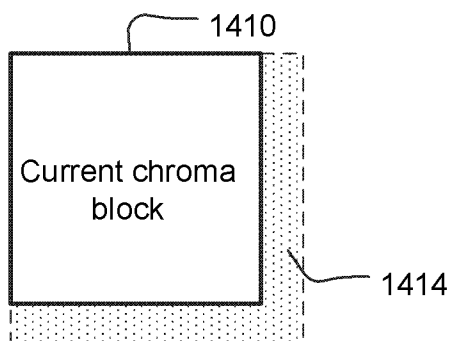


Fig. 14A

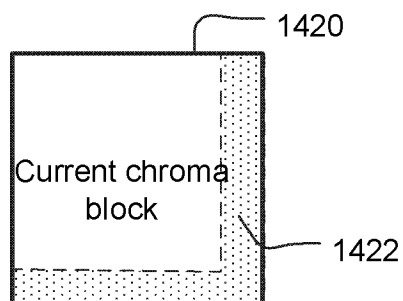
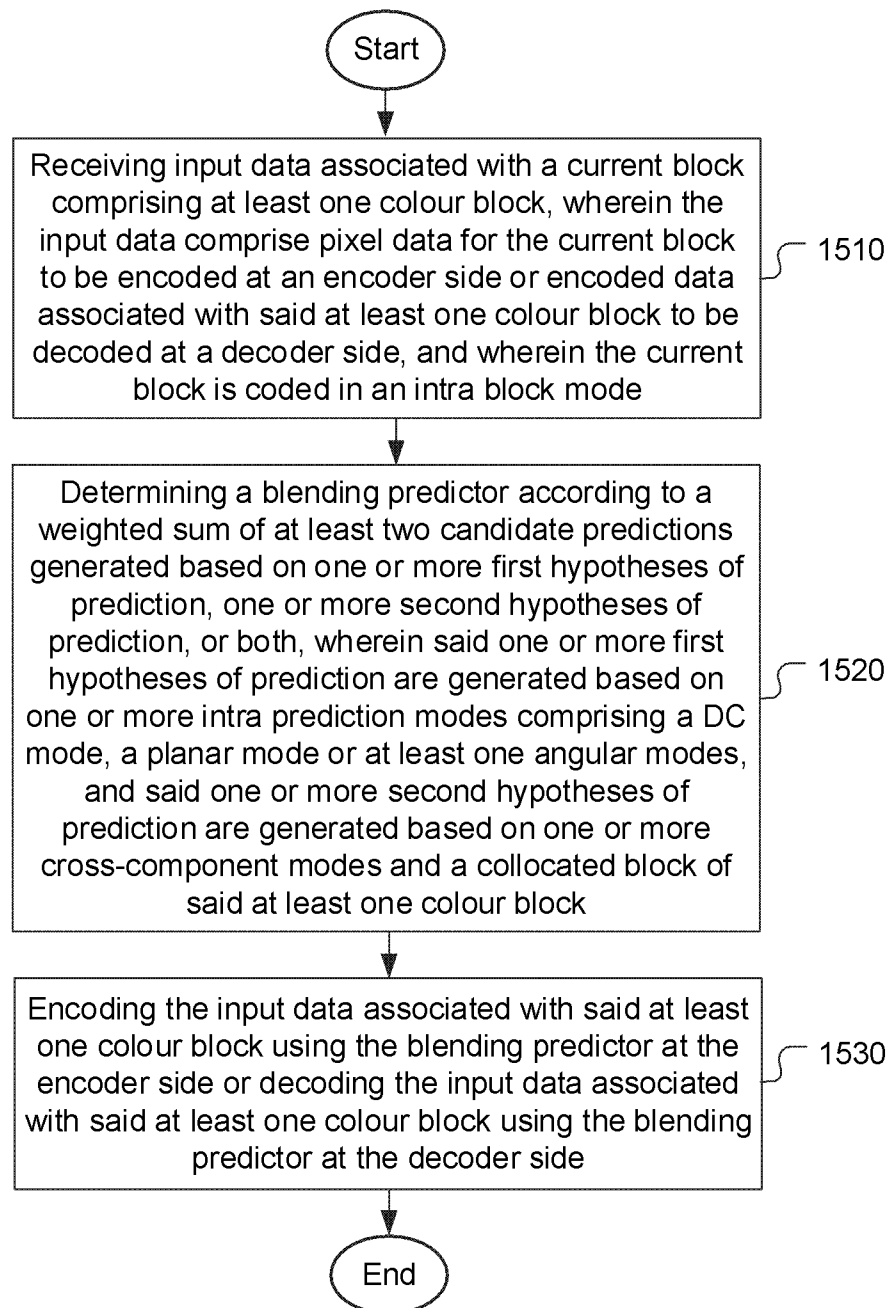


Fig. 14B

**Fig. 15**

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2022/140409

A. CLASSIFICATION OF SUBJECT MATTER

H04N19/103(2014.01)i;H04N19/176(2014.01)i;H04N19/186(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)

IPC:H04N

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

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

CNTXT,ENTXTC,ENTXT,VEN,IEEE,CNKI,JVET: prediction, intra block, decode, encode, blend, hypothes, angular mode, DC mode, planar mode, cross-component, linear model, CCLM, weight, sum, region, luma, chroma, collocated block

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2018053293 A1 (QUALCOMM INC) 22 March 2018 (2018-03-22) claims 1-18, paragraphs 0118-0200,0226, figs.15-19	1-16
A	WO 2020244536 A1 (BEIJING BYTEDANCE NETWORK TECH CO LTD et al.) 10 December 2020 (2020-12-10) the whole document	1-16
A	WO 2021110116 A1 (BEIJING BYTEDANCE NETWORK TECH CO LTD et al.) 10 June 2021 (2021-06-10) the whole document	1-16



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

“D” document cited by the applicant in the international application

“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

16 March 2023

Date of mailing of the international search report

17 March 2023

Name and mailing address of the ISA/CN

CHINA NATIONAL INTELLECTUAL PROPERTY
ADMINISTRATION
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing
100088, China

Authorized officer

BAI,JiaRong

Facsimile No. (86-10)62019451

Telephone No. (+86) 010-62411469

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2022/140409

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
WO	2018053293	A1	22 March 2018	None			
WO	2020244536	A1	10 December 2020	US	2022086447	A1	17 March 2022
WO	2021110116	A1	10 June 2021	None			