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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
09 January 2025 (09.01.2025)

WIPO | PCT

(10) International Publication Number
WO 2025/007693 A1

(51) International Patent Classification:
H04N 19/186 (2014.01) *H04N 19/50* (2014.01)

(21) International Application Number:
PCT/CN2024/097398

(22) International Filing Date:
05 June 2024 (05.06.2024)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
63/511,920 05 July 2023 (05.07.2023) US

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

(72) Inventors: **TSENG, Hsin-Yi**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **CHIANG, Man-Shu**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **TSAI, Chia-Ming**; 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).

(74) Agent: **BEIJING SANYOU INTELLECTUAL PROPERTY AGENCY LTD.**; 16th Fl., Block A, Corporate Square, No.35 Jinrong Street, Xicheng District, 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, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, 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,

(54) Title: METHODS AND APPARATUS OF INHERITING CROSS-COMPONENT MODELS FROM NON-INTRA CODED BLOCKS FOR CROSS-COMPONENT PREDICTION MERGE MODE

(57) Abstract: A method and apparatus for coding colour pictures using coding tools including one or more cross component models related modes are disclosed. According to this method, input data associated with a current block comprising a first-colour block and a second-colour block is received, wherein the input data comprise pixel data to be encoded at an encoder side or data associated with the current block to be decoded at a decoder side, and wherein the current block is coded in a non-intra mode. A cross-component model for the current block is determined. A target cross-component predictor is derived by applying the cross-component model to reconstructed first-colour samples. The second-colour block is encoded or decoded by using prediction data comprising the target cross-component predictor.

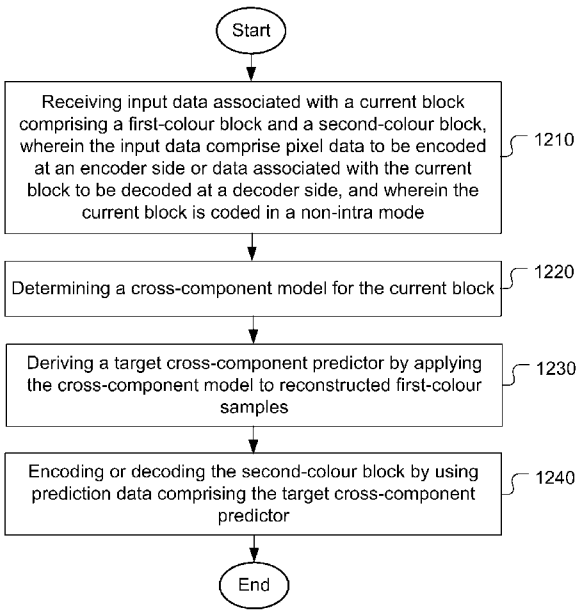


Fig. 12

[Continued on next page]

WO 2025/007693 A1

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))*

METHODS AND APPARATUS OF INHERITING CROSS-COMPONENT MODELS FROM NON-INTRA CODED BLOCKS FOR CROSS-COMPONENT PREDICTION MERGE MODE

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] The present invention is a non-Provisional Application of and claims priority to U.S. Provisional Patent Application No. 63/511,920, filed on July 5, 2023. The U.S. Provisional Patent Application is hereby incorporated by reference in its entirety.

FIELD OF THE INVENTION

[0002] The present invention relates to video coding. In particular, the present invention relates to inter prediction for chroma component by applying a cross-component model to reconstructed luma samples.

BACKGROUND AND RELATED ART

[0003] 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.

[0004] Fig. 1A illustrates an exemplary adaptive Inter/Intra video encoding system incorporating loop processing. For Intra Prediction 110, 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 on 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, is 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.

[0005] 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.

[0006] 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.

[0007] **I. Related Art**

[0008] **I.1 Intra Chroma Partitioning and Prediction Restriction**

[0009] In typical hardware video encoders and decoders, processing throughput drops when a picture has smaller intra blocks because of sample processing data dependency between neighbouring intra blocks. The predictor generation of an intra block requires top and left boundary reconstructed samples from neighbouring blocks. Therefore, intra prediction has to be sequentially processed block by block.

[0010] In HEVC, the smallest intra CU is 8x8 luma samples. The luma component of the smallest intra CU can be further split into four 4x4 luma intra prediction units (PUs), but the chroma components of the smallest intra CU cannot be further split. Therefore, the worst case hardware processing throughput occurs when 4x4 chroma intra blocks or 4x4 luma intra blocks are processed. In VVC, in order to improve worst case throughput, chroma intra CBs smaller than 16 chroma samples (e.g. size 2x2, 4x2, and 2x4) and chroma intra CBs with width smaller than 4 chroma samples (size 2xN) are disallowed by constraining the partitioning of chroma intra CBs.

[0011] In a single coding tree, a smallest chroma intra prediction unit (SCIPU) is defined as a coding tree node whose chroma block size is larger than or equal to 16 chroma samples and has at least one child luma block smaller than 64 luma samples, or a coding tree node whose chroma block size is not 2xN and has at least one child luma block with 4xN luma samples. It is required that in each SCIPU, all CBs are inter, or all CBs are non-inter, i.e, either intra or intra block copy (IBC). In case of a non-inter SCIPU, it is further required that the chroma block of the non-inter SCIPU shall not be further split and the luma block of the SCIPU is allowed to be further split. In this way, the small chroma intra CBs with size less than 16 chroma samples or with size 2xN are removed. In addition, chroma scaling is not applied in case of a non-inter SCIPU. Here, no additional syntax is signalled, and whether a SCIPU is non-inter can be derived by the prediction mode of the first luma CB in the SCIPU. The type of a SCIPU is inferred to be non-inter if the current slice is an I-slice or the current SCIPU has a 4x4 luma partition in it after further split one time (because no inter 4x4 is allowed in VVC); otherwise, the type of the SCIPU (inter or non-inter) is indicated

by one flag before parsing the CUs in the SCIPU.

[0012] For the dual tree in intra picture, the 2xN intra chroma blocks are removed by disabling vertical binary and vertical ternary splits for 4xN and 8xN chroma partitions, respectively. The small chroma blocks with size 2x2, 4x2, and 2x4 are also removed by partitioning restrictions.

[0013] In addition, a restriction on picture size is considered to avoid 2x2/2x4/4x2/2xN intra chroma blocks at the corner of pictures by considering the picture width and height to be multiple of max (8, MinCbSizeY).

[0014] **I.2 Cross-Component Linear Model Prediction**

[0015] A cross-component linear model (CCLM) prediction mode is used in the VVC, for which the chroma samples are predicted based on the reconstructed luma samples of the same CU by using a linear model as follows:

$$\mathrm{pred_C}(i,j) = \alpha \cdot \mathrm{rec_L}'(i,j) + \beta \quad (1)$$

where $\mathrm{pred_C}(i,j)$ represents the predicted chroma samples in a CU and $\mathrm{rec_L}(i,j)$ represents the downsampled reconstructed luma samples of the same CU.

[0016] In this disclosure, the terms {CCLM_LT, CCLM_L, CCLM_T} and {CCLM_LA, CCLM_L, CCLM_A} and {LM_LA, LM_L, LM_A} are used interchangeably.

[0017] The CCLM parameters (α and β) are derived with at most four neighbouring chroma samples and their corresponding down-sampled luma samples. Suppose the current chroma block dimensions are W×H, then W'' and H' are set as:

- W' = W, H' = H when LM_LA mode is applied;
- W' =W + H when LM_A mode is applied; and
- H' = H + W when LM_L mode is applied.

[0018] The above neighbouring positions are denoted as S[0, −1]…S[W' − 1, −1] and the left neighbouring positions are denoted as S[−1, 0], …, S[−1, H' − 1]. Then the four samples are selected as:

- S[W' / 4, −1], S[3 * W' / 4, −1], S[−1, H' / 4], S[−1, 3 * H' / 4] when LM_LA mode is applied and both above and left neighbouring samples are available;
- S[W' / 8, −1], S[3 * W' / 8, −1], S[5 * W' / 8, −1], S[7 * W' / 8, −1] when LM_A mode is applied or only the above neighbouring samples are available; and
- S[−1, H' / 8], S[−1, 3 * H' / 8], S[−1, 5 * H' / 8], S[−1, 7 * H' / 8] when LM_L mode is applied or only the left neighbouring samples are available.

[0019] The four neighbouring luma samples at the selected positions are down-sampled and compared four times to find two larger values: x^0_A and x^1_A, and two smaller values: x^0_B and x^1_B. Their corresponding chroma sample values are denoted as y^0_A, y^1_A, y^0_B and y^1_B. Then x_A, x_B, y_A and y_B are derived as:

$$X_a=(x^0_A + x^1_A +1)>>1;\ X_b=(x^0_B + x^1_B +1)>>1;\ Y_a=(y^0_A + y^1_A +1)>>1;\ Y_b=(y^0_B + y^1_B +1)>>1 \quad (2)$$

[0020] Finally, the linear model parameters α and β are obtained according to the following equations.

$$\alpha = \frac{Y_a - Y_b}{X_a - X_b} \quad (3)$$

$$\beta = Y_b - \alpha \cdot X_b \quad (4)$$

[0021] Fig. 2 shows an example of the location of the left and above samples and the sample of the current block involved in the LM_LA mode.

[0022] Besides the above template and left template can be used to calculate the linear model coefficients together, they also can be used alternatively in the other 2 LM modes, called LM_A, and LM_L modes.

[0023] In LM_A mode, only the above template is used to calculate the linear model coefficients. To get more samples, the above template is extended to (W+H) samples. In LM_L mode, only left template is used to calculate the linear model coefficients. To get more samples, the left template is extended to (H+W) samples.

[0024] In LM_LA mode, left and above templates are used to calculate the linear model coefficients.

[0025] **I.3 Multiple Model CCLM (MMLM)**

[0026] In the JEM (J. Chen, E. Alshina, G. J. Sullivan, J.-R. Ohm, and J. Boyce, Algorithm Description of Joint Exploration Test Model 7, document JVET-G1001, ITU-T/ISO/IEC Joint Video Exploration Team (JVET), Jul. 2017), multiple model CCLM mode (MMLM) is proposed for using two models for predicting the chroma samples from the luma samples for the whole CU. 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., a 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.

[0027] Fig. 3 shows an example of classifying the neighbouring samples into two groups. *Threshold* is calculated as the average value of the neighbouring reconstructed luma samples. A neighbouring sample with $Rec'_L[x,y] <= Threshold$ is classified into group 1; while a neighbouring sample with $Rec'_L[x,y] > Threshold$ is classified into group 2.

$$\begin{cases} Pred_C[x,y] = \alpha_1 \times Rec'_L[x,y] + \beta_1 & if\ Rec'[x,y] \le Threshold \\ (Pred_C[x,y] = \alpha_2 \times Rec'_L[x,y] + \beta_2 & if\ Rec'[x,y] > Threshold \end{cases} \quad (5)$$

[0028] **I.4 Convolutional Cross-Component Model (CCCM)**

[0029] In CCCM, a convolutional model is applied to improve the chroma prediction performance. The convolutional model has 7-tap filter consisting of a 5-tap plus sign shape spatial component, a nonlinear term and a bias term. The input to the spatial 5-tap component of the filter consists of a centre (C) luma sample which is collocated with the chroma sample to be predicted and its above/north (N), below/south (S), left/west (W) and right/east (E) neighbours as illustrated in Fig. 4.

[0030] The nonlinear term (denoted as P) is represented as power of two of the centre luma sample C and scaled to the sample value range of the content:

$$P = (C*C + midVal) >> bitDepth.$$

[0031] That is, for 10-bit content it is calculated as:

$$P = (C*C + 512) >> 10.$$

[0032] The bias term (denoted as B) represents a scalar offset between the input and output (similarly to the offset term in CCLM) and is set to the middle chroma value (512 for 10-bit content).

[0033] Output of the filter is calculated as a convolution between the filter coefficients c_i and the input values and clipped to the range of valid chroma samples:

$$predChromaVal = c_0C + c_1N + c_2S + c_3E + c_4W + c_5P + c_6B.$$

[0034] **I.5 Gradient Linear Model (GLM)**

[0035] For YUV 4:2:0 colour format, a gradient linear model (GLM) method can be used to predict the chroma samples from luma sample gradients. Two modes are supported: a two-parameter GLM mode and a three-parameter GLM mode.

[0036] Compared with the CCLM, instead of down-sampled luma values, the GLM utilizes luma sample gradients to derive the linear model. Specifically, when the GLM is applied, the input to the CCLM

process, i.e., the down-sampled luma samples L, are replaced by luma sample gradients G. The other parts of the CCLM (e.g., parameter derivation, prediction sample linear transform) are kept unchanged:

$$C = \alpha \cdot G + \beta.$$

[0037] In the three-parameter GLM, a chroma sample can be predicted based on both the luma sample gradients and down-sampled luma values with different parameters. The model parameters of the three-parameter GLM are derived from 6 rows and columns adjacent samples by the LDL decomposition based MSE minimization method as used in the CCCM.

$$C = \alpha_0 \cdot G + \alpha_1 \cdot L + \alpha_2 \cdot \beta$$

[0038] For signalling, when the CCLM mode is enabled to the current CU, one flag is signalled to indicate whether GLM is enabled for both Cb and Cr components; if the GLM is enabled, another flag is signalled to indicate which of the two GLM modes is selected and one syntax element is further signalled to select one of 4 gradient filters for the gradient calculation.

[0039] Four gradient filters (510-540) are enabled for the GLM, as illustrated in Fig. 5.

[0040] **I.6 Cross-Component Residual Model (CCRM)**

[0041] As in JVET-AD0108 (Pekka Astola, et. al., "AHG12: Cross-component residual model (CCRM) for inter prediction", Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29, 30th Meeting, Antalya, TR, 21–28 April 2023, Document: JVET- AD0108), it is to apply cross-component residual model (CCRM) to predict chroma samples from reconstructed luma samples when the block uses inter prediction or intra block copy (IBC). The following figure illustrates the decoder side of the method. The cross-component filters are derived using the prediction signals of luma and chroma. The derived filters are applied to the reconstructed luma signal producing the final chroma predictions. The input to the filter consists of 6 spatial luma samples, a non-regression-model term, and a bias term. Filter coefficients are derived in step 620 for each block separately using the prediction signals (i.e., predY 610, predCb 612 and predCr 614) and the filters are applied to the reconstructed luma signal in step 630 as shown in Fig. 6. The reconstructed luma signal is formed by combining the luma prediction (PredY) 610 and residual luma signal (resY) using an adder 622. After applying the filters, the step 630 generates filtered-predicted Cb 640 and filtered-predicted Cr 650. The reconstructed Cb signal is formed by combining the filtered-predicted Cb 640 and residual Cb signal (i.e., resCb) using an adder 642. Similarly, the reconstructed Cr signal is formed by combining the filtered-predicted Cr 650 and residual Cr signal (i.e., resCr) using an adder 652.

[0042] **I.6.1 Convolutional Filter and Calculation of Filter Coefficients**

[0043] The proposed 8-tap filter consist of 6 spatial luma samples, a nonlinear term, and a bias term. The spatial luma samples (L0, …, L5) are obtained from the luma grid selecting the 6 luma samples closest to the chroma position C without down sampling as shown in Fig. 7. The predicted chroma value is obtained as,

$$\begin{aligned}\text{predChromaVal} = c_0\,\text{L0} + c_1\text{L1} + c_2\text{L2} + c_3\text{L3} + c_4\text{L4} + c_5\text{L5} + c_6\,\text{nonlinear}((\text{L0}+\text{L3}+1) >> 1)\\ + c_7\,\text{B},\end{aligned}$$

where nonlinear is CCCM's nonlinear operator and B is bias.

[0044] The filter coefficients are derived using ECM's division-free Gaussian elimination method and the necessary offsets are applied to samples prior to filter derivation.

[0045] Intra reference samples are used as additional input samples in filter derivation when the block has less than 64 chroma samples. CCCM's design of at most 6 rows and columns of intra reference samples is used.

[0046] Blocks having 256 chroma samples or more are divided into subblocks that have at most 256 chroma samples. Subblocks containing zero luma residual are skipped.

[0047] In order to improve the coding efficiency for inter prediction of chroma component, methods and apparatus of generating inter predictor for the chroma component by applying a cross-component model to reconstructed luma samples are disclosed.

BRIEF SUMMARY OF THE INVENTION

[0048] A method and apparatus for coding colour pictures using coding tools including one or more cross component models related modes are disclosed. According to this method, input data associated with a current block comprising a first-colour block and a second-colour block is received, wherein the input data comprise pixel data to be encoded at an encoder side or data associated with the current block to be decoded at a decoder side, and wherein the current block is coded in a non-intra mode. A cross-component model for the current block is determined. A target cross-component predictor is derived by applying the cross-component model to reconstructed first-colour samples. The second-colour block is encoded or decoded by using prediction data comprising the target cross-component predictor.

[0049] In one embodiment, the cross-component model for the current block is stored, referenced by following coding blocks, or both. In one embodiment, all model parameters of the cross-component model are stored. In another embodiment, a subset of model parameters of the cross-component model is stored. In one embodiment, if a target following coding block is coded in an intra mode, the target following coding block is allowed to use the cross-component model stored. In one embodiment, if a target following coding block is coded in the non-intra mode, the target following coding block is allowed to use the cross-component model stored.

[0050] In one embodiment, the cross-component model is derived from reconstructed first-colour samples and reconstructed second-colour samples. In one embodiment, the reconstructed first-colour samples and the reconstructed second-colour samples are from regions indicated by a motion vector or a block vector.

[0051] In one embodiment, the cross-component model is derived based on CCRM (Cross-Component Residual Model).

[0052] In one embodiment, the cross-component model is inherited from a neighbouring block. In one embodiment, the neighbouring block corresponds to a spatial neighbouring block or a temporal neighbouring block.

[0053] In one embodiment, different cross-component models associated with blocks coded in different modes are stored in different buffers. In one embodiment, a first cross-component model associated with intra-coded blocks and a second cross-component model associated with inter-coded blocks are stored in two different buffers.

BRIEF DESCRIPTION OF THE DRAWINGS

[0054] Fig. 1A illustrates an exemplary adaptive Inter/Intra video encoding system incorporating loop processing.

[0055] Fig. 1B illustrates a corresponding decoder for the encoder in Fig. 1A.

[0056] Fig. 2 illustrates an example of sample locations used for deriving α and β of the cross-component model.

[0057] Fig. 3 illustrates an example of classifying the neighbouring samples into two groups for

MMLM.

[0058] Fig. 4 illustrates the spatial part of the convolutional filter.

[0059] Fig. 5 shows 4 gradient patterns for GLM.

[0060] Fig. 6 illustrates a system block diagram for CCRM decoder.

[0061] Fig. 7 illustrates the six luma samples L0, ..., L5 in relation to the chroma sample C.

[0062] Fig. 8 illustrates the 5 neighbouring blocks used for deriving spatial merge candidates for VVC.

[0063] Fig. 9 illustrates an example of inheriting temporal neighbouring model parameters.

[0064] Figs. 10A-B illustrates two search patterns for inheriting non-adjacent spatial neighbouring models.

[0065] Fig. 11 illustrates an example of the neighbouring templates for calculating model error for reordering the candidates.

[0066] Fig. 12 illustrates a flowchart of an exemplary video coding system that generates inter predictor for the chroma component by applying a cross-component model to reconstructed luma samples according to an embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0067] 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.

[0068] 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.

[0069] The following methods are proposed to improve the cross-component prediction accuracy or coding performance by generating inter predictor for the chroma component by applying a cross-component model to reconstructed luma samples.

[0070] **II. Proposed Method**

[0071] **II.1 Inheriting Neighbouring Model Parameters for Refining the Cross-Component Model Parameters**

[0072] When applying cross-component prediction coding tool on the current block to generate prediction signals, the cross-component model (CCM) information (more details in section entitled:

"Inheriting CCM information") including model parameters can be inherited from neighbouring blocks.

[0073] In one embodiment, if the inherited neighbouring block is coded in CCLM mode, the final scaling parameter of the current block is inherited from the neighbouring blocks. Once the final scaling parameter is determined, the offset parameter (e.g., β in CCLM) is derived based on the inherited scaling parameter and/or the average value of neighbouring luma and chroma samples of the current block.

[0074] In one embodiment, if the inherited neighbour block is coded in CCLM mode, after inheriting model parameters, the offset parameter can be inherited or can be further refined by dB. For example, if the final offset parameter is inherited from a selected neighbouring block, and the inherited offset parameter is β'_{nei}, then the final scaling parameter is (β'_{nei} + dB). dB can be zero.

[0075] In one embodiment, if the inherited neighbour block is coded with CCCM, the filter coefficients (c_i) are inherited. The offset parameter (e.g., $c_6 \times B$ or c_6 in CCCM) can be re-derived based on the inherited parameter and the average value of neighbouring corresponding position luma and chroma samples of the current block.

[0076] In one embodiment, if the inherited neighbour block is coded with CCCM, the filter coefficients (c_i) are inherited. The offset parameter (e.g., $c_6 \times B$ or c_6 in CCCM) is also inherited and is not re-derived.

[0077] In one embodiment, if the inherited candidate applies GLM gradient pattern to its luma reconstruction samples, the current block shall also inherit the GLM gradient pattern of the candidate and apply to the current luma reconstruction samples.

[0078] **II.2 Inheriting CCM Information**

[0079] In one embodiment, the cross-component model (CCM) information of inherited cross-component model can be stored together with the inherited model parameters. The CCM information can be inherited together with the inherited model parameters. The prediction of the current block can be generated based on the inherited CCM information and inherited model parameters. The CCM information can include but not limited to prediction mode (e.g., CCLM, MMLM, CCCM, 2-parameter GLM, 3-parameter GLM), model index for indicating which model shape is used in convolutional model, classification threshold for multi-model, information to indicate that non-downsampled samples are used in convolutional model, down-sampling filter flag, down-sampling filtering index when multiple down-sampling filters are used, number of neighbouring lines used to derive model, types of templates used to derive model, post-filtering flag and model parameters.

[0080] In one embodiment, a mixed CCCM model consisting of various terms (e.g., spatial term, gradient term, location term, non-linear term and bias term) can be inherited. In addition to storing model parameters, a prediction mode can be stored in the CCM information to indicate that the inherited model is a mixed CCCM model consisting of various terms. If there are multiple types of mixed CCCM models, a model index can also be stored in the CCM information to indicate which type of mixed CCCM model is inherited. For example, gradient and location based CCCM (GL-CCCM) proposed in JVET-AB0119 (Ramin G. Youvalari, et al., "Non-EE2: Gradient and location based convolutional cross-component model (GL-CCCM) for intra prediction", Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 28th Meeting, Mainz, DE, 20–28 October 2022, Document: JVET- AB0119) is a mixed CCCM model which consists of one spatial term in centre position, two gradient terms for the horizontal direction and vertical direction, two location terms X and Y for the relative horizontal location and relative vertical location, one non-linear term and one bias term. A prediction mode can be stored in the CCM information to indicate that the inherited model is a GL-CCCM model.

[0081] **II.3 Inheriting Spatial Neighbouring Model Parameters**

[0082] In one embodiment, the inherited model parameters can be from a block that is an immediate neighbouring block. The models from blocks at pre-defined positions are added into the candidate list in a pre-defined order.

[0083] In one embodiment, the pre-defined positions and the pre-defined order can be the same as those of spatial candidates for inter merge mode.

[0084] In one embodiment, the pre-defined positions can be the positions depicted in Fig. 8. The pre-defined order can be B_0, A_0, B_1, A_1 and B_2.

[0085] In one embodiment, assume the position, width and height of the current block are (x, y), W and H respectively, the pre-defined positions can include positions immediate above the current block, such as (x + W >> 1, y-1) or (x + (W+1) >> 1, y-1), if W is greater than or equal to a threshold TH. The pre-defined positions can also include positions immediate left to the current blocks, such as (x-1, y+H>>1) or (x-1, y+(H+1)>>1), if H is greater than or equal to a threshold TH. TH can be 2, 4, 8, 16, 32, or 64.

[0086] In one embodiment, there is a maximum number of inherited models from spatial neighbours that can be added into the candidate list, and the maximum number is smaller than the number of pre-defined positions.

[0087] **II.4 Inheriting Temporal Neighbouring Model Parameters**

[0088] In one embodiment, if the current slice/picture is a non-intra slice/picture, the inherited model parameters can be from the block in the previous coded slices/pictures.

[0089] In one embodiment, if the current slice/picture is a non-intra slice/picture, the inherited model parameters can be from the block in the previous coded slices/pictures. For example, as shown in Fig. 9, the current block position is at (x, y) and the block size is $w \times h$. The inherited model parameters can be from the block at position (x', y'), (x', y' + h/2), (x' + w/2, y'), (x' + w/2, y' + h/2), (x' + w, y'), (x', y' + h), or (x' + w, y' + h) of the previous coded slices/picture, where x' = x + Δx and y' = y + Δy. In one embodiment, if the prediction mode of the current block is intra, Δx and Δy are set to 0. If the prediction mode of the current block is inter, Δx and Δy are set to the horizontal and vertical motion vectors of the current block. In one embodiment, if the current block is inter bi-prediction, Δx and Δy are set to the horizontal and vertical motion vectors in reference picture list 0. In one embodiment, if the current block is inter bi-prediction, Δx and Δy are set to the horizontal and vertical motion vectors in reference picture list 1.

[0090] In one embodiment, if the current block is inter bi-prediction, the inherited model parameters can be from the block in the previous coded slices/pictures in the reference lists. For example, if the horizontal and vertical motion vectors in reference picture list 0 is Δx_{L0} and Δy_{L0}, the motion vector can be scaled to other reference pictures in the reference list 0 and 1. If the motion vector is scaled to the i^{th} reference picture in the reference list 0 as $(\Delta x_{L0,i0}, \Delta y_{L0,i0})$. The model can be from the block in the i^{th} reference picture in the reference list 0, and Δx and Δy are set to $(\Delta x_{L0,i0}, \Delta y_{L0,i0})$. For another example, if the horizontal and vertical motion vectors in reference picture list 0 is Δx_{L0} and Δy_{L0}, the motion vector is scaled to the i^{th} reference picture in the reference list 1 as $(\Delta x_{L0,i1}, \Delta y_{L0,i1})$. The model can be from the block in the i^{th} reference picture in the reference list 1, and Δx and Δy are set to $(\Delta x_{L0,i1}, \Delta y_{L0,i1})$.

[0091] In one embodiment, if the current slice/picture is a non-intra slice/picture, the inherited model parameters can be from the block in the previous coded slices/pictures. In one embodiment, the current block position is at (x, y) and the block size is $w \times h$. The two value sets α_x and α_y are defined as:

$$\alpha_x = \{\alpha_{x1}, \alpha_{x2}, \alpha_{x3}, \ldots, \alpha_{xn}\}, \alpha_{xi} < \alpha_{xj} \; if \; i < j$$

$\alpha_y = \{\alpha_{y1}, \alpha_{y2}, \alpha_{y3}, \ldots, \alpha_{yn}\}, \alpha_{yi} < \alpha_{yj}\ if\ i < j$

[0092] All values in α_x and α_y are positive numbers. Let $(x_{mid}, y_{mid}) = (x + w/2, y + h/2)$. The inherited model parameters can be from the block at positions $(x_{mid} \pm \alpha_{xi} \times w,\ y_{mid} \pm \alpha_{yi} \times h),, (x_{mid} \pm \alpha_{xi} \times w, y_{mid}),\ (x_{mid}, y_{mid} \pm \alpha_{yi} \times h)$ of the previous coded slices/picture.

[0093] In one sub-embodiment, $\alpha_x = \alpha_y$. For example, $\alpha_x = \alpha_y = \{1, 2, 3, 4, 5\}$.

[0094] In another sub-embodiment, $\alpha_x \neq \alpha_y$. For example, $\alpha_x = \{1/2,\ 1,\ 3/2,\ 2,\ 5/2\}$ and $\alpha_y = \{1, 2, 3, 4, 5\}$.

[0095] In another embodiment, the current block position is at (x, y) and the block size is $w \times h$. The inherited model parameters can be from the block at positions $(x + \alpha_{xi} \times w,\ y + \alpha_{yi} \times h), (x + \alpha_{xi} \times w, y - \alpha_{yi} \times h), (x - \alpha_{xi} \times w,\ y + \alpha_{yi} \times h), (x - \alpha_{xi} \times w,\ y - \alpha_{yi} \times h), (x + \alpha_{xi} \times w, y), (x - \alpha_{xi} \times w, y), (x, y + \alpha_{yi} \times h)$, or $(x, y - \alpha_{yi} \times h)$ of the previous coded slices/picture.

[0096] In one sub-embodiment, $\alpha_x = \alpha_y$. For example, $\alpha_x = \alpha_y = \{1, 2, 3, 4, 5\}$.

[0097] In another sub-embodiment, $\alpha_x \neq \alpha_y$. For example, $\alpha_x = \{1/2,\ 1,\ 3/2,\ 2,\ 5/2\}$ and $\alpha_y = \{1, 2, 3, 4, 5\}$.

[0098] In one embodiment, the models from the positions closer to (x_{mid}, y_{mid}) are added into the final merge candidate list first. In another embodiment, the models from the positions closer to (x, y) are added into the final merge candidate list first.

[0099] In one embodiment, let δx and δy be two fixed positive numbers. The inherited model parameters can be from the block at positions $(x_{mid} + \alpha_{xi} \times \delta x,\ y_{mid} + \alpha_{yi} \times \delta y), (x_{mid} + \alpha_{xi} \times \delta x, y_{mid} - \alpha_{yi} \times \delta y), (x_{mid} - \alpha_{xi} \times \delta x,\ y_{mid} + \alpha_{yi} \times \delta y), (x_{mid} - \alpha_{xi} \times \delta x,\ y_{mid} - \alpha_{yi} \times \delta y), (x_{mid} + \alpha_{xi} \times \delta x, y_{mid}), (x_{mid} - \alpha_{xi} \times \delta x, y_{mid}), (x_{mid}, y_{mid} + \alpha_{yi} \times \delta y)$, or $(x_{mid}, y_{mid} - \alpha_{yi} \times \delta y)$ of the previous coded slices/picture.

[0100] In another embodiment, the current block position is at (x, y) and the block size is $w \times h$. Let δx and δy be two fixed positive numbers The inherited model parameters can be from the block at positions $(x + \alpha_{xi} \times \delta x,\ y + \alpha_{yi} \times \delta y), (x + \alpha_{xi} \times \delta x,\ y - \alpha_{yi} \times \delta y), (x - \alpha_{xi} \times \delta x,\ y + \alpha_{yi} \times \delta y), (x - \alpha_{xi} \times \delta x,\ y - \alpha_{yi} \times \delta y), (x + \alpha_{xi} \times \delta x, y), (x - \alpha_{xi} \times \delta x, y), (x, y + \alpha_{yi} \times \delta y)$, or $(x, y - \alpha_{yi} \times \delta y)$ of the previous coded slices/picture.

[0101] In another embodiment, the current block position is at (x, y) and the block size is $w \times h$. The inherited model parameters can be from the block at some pre-defined positions $(x',\ y')$ of the previous coded slices/picture. For one example, the positions are inside the corresponding area of the current encoding block, i.e., $x \leq x' < x + w$ and $y \leq y' < y + h$. The inherited model parameters can be from the block at $(x,\ y), (x + w - 1,\ y), (x,\ y + h - 1), (x + w - 1,\ y + h - 1)$, or $(x + w/2, y + h/2)$. For another example, the positions are outside of the corresponding area of the current encoding block, i.e., $x' < x +$ *or* $x' \geq x + w$, and $y' < y$ *or* $y' \geq y + h$. The inherited model parameters can be from the block at $(x - 1,\ y), (x,\ y - 1), (x - 1,\ y - 1), (x + w,\ y), (x + w - 1,\ y - 1),\ (x + w,\ y - 1), (x,\ y + h), (x - 1,\ y + h - 1), (x - 1,\ y + h), (x + w,\ y + h - 1), (x + w - 1,\ y + h)$, or $(x + w,\ y + h)$.

[0102] The previous coded picture which the inherited parameter model is from is referred as the collocated picture hereafter.

[0103] In one embodiment, the previous coded picture where the inherited parameter model is from (i.e., the collocated picture) is one of the pictures in the reference lists.

[0104] In one embodiment, the collocated picture is signalled in the picture/slice header. The reference list and the reference index are signalled in the picture/slice header. For example, the collocated picture is

selected as L0[0]. For another example, the collocated picture is selected as L1[0].

[0105] In one embodiment, the collocated picture is selected as the picture in the reference lists whose POC difference between the respective picture and the current picture is the smallest. For example, if the POC of current picture is 8, the POCs of pictures in reference list 0 are {7, 6, 5, 0} and POCs of pictures in reference list 1 are {7, 6, 5, 4}, then L0[0] (equivalent to L1[0]) is selected since its POC difference is the smallest.

[0106] In one sub-embodiment, if there are two pictures whose POC difference between the respective picture and the current picture are both the smallest, the picture with the smaller POC is selected.

[0107] In one sub-embodiment, if there are two pictures whose POC difference between the respective picture and the current picture are both the smallest, the picture with the larger POC is selected.

[0108] In another sub-embodiment, if there are two pictures whose POC difference between the respective picture and the current picture are both the smallest, the picture with smaller QP difference between the respective picture and the current picture is selected.

[0109] In another sub-embodiment, if there are two pictures whose POC difference between the respective picture and the current picture are both the smallest, the picture with the smaller QP is selected.

[0110] In another sub-embodiment, if there are two pictures whose POC difference between the respective picture and the current picture are both the smallest, the picture with the larger QP is selected.

[0111] In one embodiment, the collocated picture is selected as the picture in the reference lists whose QP difference between the respective picture and the current picture is the smallest. For example, if the QP of current picture is 28, and the QPs of the pictures in reference list 0 are {19, 26, 23} and the QPs of the pictures in reference list 1 are {23, 22, 21}. Then L0[1] is selected.

[0112] In one sub-embodiment, if there are more than one picture in the reference lists whose QP difference between the respective picture and the current picture are the smallest, the picture with the smaller QP is selected.

[0113] In another sub-embodiment, if there are more than one picture in the reference lists whose QP difference between the respective picture and the current picture are the smallest, the picture with the larger QP is selected.

[0114] In another sub-embodiment, if there are more than one picture whose QP difference between the respective picture and the current picture are the smallest, the picture with the smaller POC distance is selected.

[0115] In one embodiment, the collocated picture is selected as the picture in the reference lists whose QP is the smallest.

[0116] In another embodiment, the collocated picture is selected as the picture in the reference lists whose QP is the largest.

[0117] In one embodiment, the previous coded picture which the inherited parameter model is from, i.e., the collocated picture, is the most recently coded I-picture. The cross-component model information of the most recently coded I-slice/picture is stored in a long-term reference buffer.

[0118] In one embodiment, the collocated picture and the pre-defined positions where the inherited parameter model is from is determined by the motion vector of a neighbouring block. For example, if the current block position is at (x, y) and the block size is $w \times h$. The inherited model parameters can be from the block at position (x', y'), (x', y' + h/2) , (x' + w/2, y'), (x' + w/2, y' + h/2), (x' + w, y'), (x', y' + h), or (x' + w, y' + h) of the collocated picture, where $x' = x + \Delta x$ and $y' = y + \Delta y$.

[0119] In one sub-embodiment, Δx and Δy are set to the L0 horizontal and vertical motion vectors of the neighbouring block, and the collocated picture is the L0 reference picture indicated by the L0 motion vector of the neighbouring block.

[0120] In one sub-embodiment, if the neighbouring block is inter bi-prediction, Δx and Δy are set to the L1 horizontal and vertical motion vectors of the neighbouring block, and the collocated picture is the L1 reference picture indicated by the L1 motion vector of the neighbouring block.

[0121] In one sub-embodiment, the neighbouring block is the left block of the current block.

[0122] In another sub-embodiment, the neighbouring block is the above block of the current block.

[0123] In one embodiment, the pre-defined positions in the previous coded slices/pictures where the inherited parameter model is from is determined by the motion vector of a neighbouring block. Let Δx and Δy be the horizontal and vertical displacements determined based on the selected motion vector of the neighbouring block, the current block position is at (x, y) and the block size is $w \times h$. The inherited model parameters can be from the block at position (x', y'), where x' = x + Δx and y' = y + Δy, or where x' = x + w/2 +Δx and y' = y + h/2 +Δy.

[0124] In another embodiment, the inherited model parameters can also be from the block positions in the patterns described in earlier paragraphs. The positions are centred at (x', y'), where x' = x + Δx and y' = y + Δy, or where x' = x + w/2 +Δx and y' = y + h/2 +Δy. That is, denote the pre-defined positions as $(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots, (x_N, y_N)$, the inherited model parameters can be from $(x_1 + \Delta x, y_1 + \Delta y), (x_2 + \Delta x, y_2 + \Delta y), (x_3 + \Delta x, y_3 + \Delta y), \dots, (x_N + \Delta x, y_N + \Delta y)$, where Δx and Δy are the horizontal and vertical displacements determined based on the selected motion vector of the neighbouring block. For example, let the current block size be $w \times h$. The two value sets α_x and α_y are defined as:

$$\alpha_x = \{\alpha_{x1}, \alpha_{x2}, \alpha_{x3}, \dots, \alpha_{xn}\}, \alpha_{xi} < \alpha_{xj} \; if \; i < j$$
$$\alpha_y = \{\alpha_{y1}, \alpha_{y2}, \alpha_{y3}, \dots, \alpha_{yn}\}, \alpha_{yi} < \alpha_{yj} \; if \; i < j$$

[0125] All values in α_x and α_y are positive numbers. The inherited model parameters can be from the block at positions $(x' + \alpha_{xi} \times w, \; y' + \alpha_{yi} \times h), (x' + \alpha_{xi} \times w, \; y' - \alpha_{yi} \times h), (x' - \alpha_{xi} \times w, \; y' + \alpha_{yi} \times h), (x' - \alpha_{xi} \times w, \; y' - \alpha_{yi} \times h), (x' + \alpha_{xi} \times w, y'), (x' - \alpha_{xi} \times w, y'), (x', y' + \alpha_{yi} \times h)$, or $(x', y' - \alpha_{yi} \times h)$ of the previous coded slices/picture. For another example, let δx and δy be two fixed positive numbers. The inherited model parameters can be from the block at positions $(x' + \alpha_{xi} \times \delta x, \; y' + \alpha_{yi} \times \delta y), (x' + \alpha_{xi} \times \delta x, \; y' - \alpha_{yi} \times \delta y), (x' - \alpha_{xi} \times \delta x, \; y' + \alpha_{yi} \times \delta y), (x' - \alpha_{xi} \times \delta x, \; y' - \alpha_{yi} \times \delta y), (x' + \alpha_{xi} \times \delta x, y'), (x' - \alpha_{xi} \times \delta x, y'), (x', y' + \alpha_{yi} \times \delta y)$, or $(x', y' - \alpha_{yi} \times \delta y)$ of the previous coded slices/picture. For another example, the inherited model parameters can be from the block at some pre-defined positions relative to $(x', \; y')$ of the previous coded slices/picture. The positions can be $(x', \; y'), (x' + w - 1, \; y'), (x', y' + h - 1), (x' + w - 1, \; y' + h - 1)$, or $(x' + w/2, y' + h/2)$. For another example, the positions can be $(x' - 1, \; y'), (x', \; y' - 1), (x' - 1, \; y' - 1), (x' + w, \; y'), (x' + w - 1, \; y' - 1), \; (x' + w, \; y' - 1), (x', \; y' + h), (x' - 1, \; y' + h - 1), (x' - 1, \; y' + h), (x' + w, \; y' + h - 1), (x' + w - 1, \; y' + h)$, or $(x' + w, \; y' + h)$.

[0126] In one embodiment, the neighbouring block can be at a pre-defined position. For example, the position can be at the A_0 position as depicted in Fig. 8. The pre-defined position can also be at A_1, B_0, B_1, B_2. If the block at the pre-defined position is not an inter block, no neighbouring block is selected.

[0127] In another embodiment, when selecting the neighbouring block, there can be a list of pre-defined positions. The positions are placed according to the checking order. For example, the positions can be B_0, A_0, B_1, A_1 and B_2 as depicted in Fig. 8. The selected neighbouring block can be the first position in the list that is an inter block. The L0 motion vector is selected. If the L0 motion vector is not available, select the L1

motion vector. For another example, the L1 motion vector is selected. If the L1 motion vector is not available, select the L0 motion vector

[0128] In another embodiment, if the collocated picture has been determined (e.g., it can be determined with the methods described in earlier paragraphs in this section), the positions in the list of pre-defined positions are checked in the pre-defined checking order. The selected motion vector is the first whose reference picture is the collocated picture. For example, the positions can be B_0, A_0, B_1, A_1 and B_2 as depicted in Fig. 8. For each position, the L0 motion vector is first checked, and then the L1 motion vector. That is, the checking order is (B_0, L_0), (B_0, L_1), (A_0, L_0), (A_0, L_1), …, (B_2, L_1). For another example, the L1 motion vector is first checked, and then the L0 motion vector.

[0129] In one embodiment, the inherited model parameters can also be from the block positions in the patterns described in earlier paragraphs. The positions are centred at (x', y'), where x' = x + Δx and y' = y + Δy. The horizontal and vertical displacements Δx and Δy are determined based on the selected motion vector of the neighbouring block. For example, if the reference picture of the selected motion vector and the collocated picture are the same picture, Δx equals to the horizontal part of the selected motion vector and Δy equals to the vertical part of the selected motion vector. If the horizontal part or the vertical part of the selected motion vector is fractional, Δx equals to the horizontal part of the selected motion vector after rounding and Δy equals to the vertical part of the selected motion vector after rounding. The rounding method used can be but not limited to the following methods: rounding toward negative infinity, rounding toward positive infinity, rounding toward zero, or rounding to the nearest integer (e.g., rounding away from zero, rounding half up, rounding half down, …). For another example, if the reference picture of the selected motion vector and the collocated picture are not the same. The reference picture can be one of the pictures in the reference list, while the collocated picture is signalled in the picture/slice header. Let the POC distance between the current picture and the reference picture of the selected motion vector be tb, and the POC distance between the current picture and the collocated picture be td, the selected motion vector be (mv_x, mv_y). Δx = mv_x * (td/tb) and Δy = mv_y * (td/tb). If mv_x * (td/tb) or mv_y * (td/tb) is fractional, Δx equals to mv_x * (td/tb) after rounding or the horizontal part of the selected motion vector after rounding and Δy equals to mv_y * (td/tb) after rounding or the vertical part of the selected motion vector after rounding. The rounding method used can be, but not limited to, the following methods: rounding toward negative infinity, rounding toward positive infinity, rounding toward zero, or rounding to the nearest integer (e.g., rounding away from zero, rounding half up, rounding half down, …).

[0130] In one embodiment, the inherited model parameters are derived by using the luma and chroma reconstruction samples of the collocated block. Let the current block position be at (x, y) and the block size is $w \times h$. The collocated block is a block positioned at (x', y') in the collocated picture with block size $w \times h$, when the inherited model is from position (x', y'). For another example, the collocated block can be a block positioned at (x', y') in the collocated picture with block size $m \times n$, where m and n are fixed positive values. For example, the collocated block can be at (x, y). For another example, if Δx and Δy are the L0 horizontal and vertical motion vectors of the neighbouring block, and the collocated picture is the L0 reference picture indicated by the L0 motion vector of the neighbouring block, the collocated block can be at $(x + \Delta x,\ y + \Delta y)$ in the collocated picture. (x', y') can be the block positions in the patterns described in earlier paragraphs. For example, (x', y') can be $(x + \alpha_{xi} \times w,\ y + \alpha_{yi} \times h), (x + \alpha_{xi} \times w,\ y - \alpha_{yi} \times h), (x - \alpha_{xi} \times w,\ y + \alpha_{yi} \times h), (x - \alpha_{xi} \times w,\ y - \alpha_{yi} \times h), (x + \alpha_{xi} \times w, y), (x - \alpha_{xi} \times w, y), (x, y + \alpha_{yi} \times h)$, or $(x, y - \alpha_{yi} \times h)$.

[0131] In one embodiment, the cross-component parameter model can be inherited from more than one previous coded picture. The cross-component parameter model can be inherited from any picture in a picture set, which contains N previous coded pictures. An index can be signalled/parsed in the bitstream to indicate the selected picture. The index ranges from 0 to N-1. In one sub-embodiment, the picture whose POC difference between itself and the current picture is smaller is associated with the smaller index. In another sub-embodiment, the picture whose QP difference between itself and the current picture is smaller is associated with the smaller index. In another sub-embodiment, the picture whose QP is smaller is associated with the smaller index. In another sub-embodiment, the picture whose QP is larger is associated with the smaller index.

[0132] **II.5 Inheriting Non-Adjacent Spatial Neighbouring Models**

[0133] In one embodiment, the inherited model parameters can be from blocks that are non-adjacent spatial neighbouring blocks. The models from blocks at pre-defined positions are added into the candidate list in a pre-defined order.

[0134] In one sub-embodiment, the pre-defined positions and the pre-defined order are the same as those of non-adjacent spatial neighbouring candidates for inter merge mode.

[0135] In one sub-embodiment, the pre-defined positions and the pre-defined order are as depicted in Fig. 10A and Fig. 10B. The positions of the numbered squares are the pre-defined positions. The number inside each square indicates the pre-defined order. Positions in Pattern 1 (1010) is added into the list before positions in Pattern 2 (1020). The distance between each pre-defined positions are proportional to the width and height of the current block.

[0136] In one embodiment, there is a maximum number of inherited models from non-adjacent spatial neighbours that can be added into the candidate list, and the maximum number is smaller than the number of pre-defined positions.

[0137] In one embodiment, let the current block position be (x, y) and the block size be $w \times h$. The two value sets α_x and α_y are defined as:

$$\alpha_x = \{\alpha_{x1}, \alpha_{x2}, \alpha_{x3}, \dots, \alpha_{xn}\}, \alpha_{xi} < \alpha_{xj} \; if \; i < j$$
$$\alpha_y = \{\alpha_{y1}, \alpha_{y2}, \alpha_{y3}, \dots, \alpha_{yn}\}, \alpha_{yi} < \alpha_{yj} \; if \; i < j$$

[0138] All values in α_x and α_y are positive numbers. Let x' = x + Δx and y' = y + Δy. The inherited model parameters can be from the positions determined by x' and y'. For example, the inherited model parameters can be from the block at positions $(x' + \alpha_{xi} \times w,\ y' + \alpha_{yi} \times h), (x' + \alpha_{xi} \times w,\ y' - \alpha_{yi} \times h), (x' - \alpha_{xi} \times w,\ y' + \alpha_{yi} \times h), (x' - \alpha_{xi} \times w,\ y' - \alpha_{yi} \times h), (x' + \alpha_{xi} \times w, 0), (x' - \alpha_{xi} \times w, 0), (0, y' + \alpha_{yi} \times h), (0, y' - \alpha_{yi} \times h), (x' + \alpha_{xi} \times w, y'), (x' - \alpha_{xi} \times w, y'), (x', y' + \alpha_{yi} \times h), (x', y' - \alpha_{yi} \times h)$. For another example, let δx and δy be two fixed positive numbers. The inherited model parameters can be from the block at positions $(x' + \alpha_{xi} \times \delta x,\ y' + \alpha_{yi} \times \delta y), (x' + \alpha_{xi} \times \delta x,\ y' - \alpha_{yi} \times \delta y), (x' - \alpha_{xi} \times \delta x,\ y' + \alpha_{yi} \times \delta y), (x' - \alpha_{xi} \times \delta x,\ y' - \alpha_{yi} \times \delta y), (x' + \alpha_{xi} \times \delta x, 0), (x' - \alpha_{xi} \times \delta x, 0), (0, y' + \alpha_{yi} \times \delta y), (0, y' - \alpha_{yi} \times \delta y), (x' + \alpha_{xi} \times \delta x, y'), (x' - \alpha_{xi} \times \delta x, y'), (x', y' + \alpha_{yi} \times \delta y), (x', y' - \alpha_{yi} \times \delta y)$. For another example, the inherited model parameters can be from the block at some pre-defined positions relative to $(x',\ y')$ of the previous coded slices/picture. The positions can be $(x',\ y'), (x' + w - 1,\ y'), (x',\ y' + h - 1), (x' + w - 1,\ y' + h - 1)$, or $(x' + w/2, y' + h/2)$. For another example, the positions can be $(x' - 1,\ y'), (x',\ y' - 1), (x' - 1,\ y' - 1), (x' + w,\ y'), (x' + w - 1,\ y' - 1), (x' + w,\ y' - 1), (x',\ y' + h), (x' - 1,\ y' + h - 1), (x' - 1,\ y' + h), (x' + w,\ y' + h - 1), (x' + w - 1,\ y' + h)$, or $(x' + w,\ y' + h)$. For another example, the position can be (x', y'), (x', y' + h/2), (x' +

w/2, y'), (x' + w/2, y' + h/2), (x' + w, y'), (x', y' + h), or (x' + w, y' + h).

[0139] In one sub-embodiment, if the prediction mode of the current block is IBC, Δx and Δy can be set according to the horizontal and vertical block vectors of the current block. For example, Δx and Δy can be equal to the horizontal and vertical block vectors of the current block.

[0140] In another sub-embodiment, Δx and Δy can be set according to the horizontal and vertical block vectors of a neighbouring block. For example, Δx and Δy can be equal to the horizontal and vertical block vectors of a neighbouring block.

[0141] **II.6 Inheriting Model Parameters from History Table**

[0142] In one embodiment, the inherited model parameters can be from a cross-component model history table. The history table stores CCM information of valid previous coded blocks. The valid previous coded block refers to any blocks containing valid CCM information. The cross-component models in the history table can be added into the candidate list according to a pre-defined order. In one embodiment, the adding order of historical candidates can be from the beginning of the table to the end of the table. In another embodiment, the adding order of historical candidates can be from the end of the table to the beginning of the table.

[0143] In one embodiment, one cross-component model history table can be maintained for storing the previous cross-component model (i.e., CCM information), and the cross-component model history table can be reset at the start of the current picture, current slice, current tile, every M CTU rows or every N CTUs, where N and M can be any value greater than 0. In another embodiment, the cross-component model history table can be reset at the end of the current picture, current slice, current tile, current CTU row or current CTU.

[0144] In another embodiment, multiple history tables are used for storing different types of cross-component models. For example, the first history table is used for storing single model, and the second history table is used for storing multi-model. For another example, the first history table is used for storing gradient model, and the second history table is used for storing non-gradient model. For another example, the first history table is used for storing simple linear model (e.g., $y = ax + b$), and the second history table is used for storing complicated model (e.g., CCCM).

[0145] In one embodiment, when adding historical candidates from multiple history tables to the candidate list, the adding order can be from the beginning of to the end of a certain table, and then the next history table is added in the same order or in a reversed order.

[0146] **II.7 Inheriting from Fusion Mode**

[0147] Fusion mode refers to a mode that fuses two predictions to generate the final prediction. In the chroma intra fusion mode, a chroma intra prediction that is not generated using a cross-component prediction (CCP) coding tool (e.g., CCLM, MMLM, CCCM) is fused with another chroma intra prediction generated using a cross-component prediction coding tool. For example, a non-CCLM coded intra prediction and a CCLM coded intra prediction are fused together to obtain the final intra prediction.

[0148] In one embodiment, when inheriting the cross-component model parameters from the block/position coded by chroma intra fusion mode, the model parameters for obtaining the CCP coded intra prediction are inherited and further refined.

[0149] In one embodiment, in addition to inheriting and refining the CCP model parameters, the fusion weight, the coding mode of non-CCP coded intra prediction are also inherited. That is, the chroma intra fusion mode is inherited.

[0150] **II.8 Constructing a Candidate List**

[0151] In one embodiment, the candidate list is constructed by adding candidates in a pre-defined order until the maximum candidate number is reached. The candidates added can include all or some of the aforementioned candidates, but not limited to the aforementioned candidates. For example, the pre-defined order can be spatial adjacent candidates, temporal candidates, spatial non-adjacent candidates, historical candidates, and then default candidates.

[0152] In another embodiment, if all the pre-defined neighbouring and historical candidates are added but the maximum candidate number is not reached, some default candidates are added into the candidate list until the maximum candidate number is reached.

[0153] In one embodiment, the default candidates can be CCLM models. The scaling parameter α is from the set $\{0,\ 1/8, -1/8, +2/8, -2/8, +3/8, -3/8, +4/8, -4/8, \ldots, +N/8, -N/8\}$, where N is a positive integer. The offset parameter β can be $1/(1 \ll bit_depth)$ or can be derived based on neighbouring luma and chroma samples. For example, if the average value of neighbouring luma and chroma samples are lumaAvg and chromaAvg, $\beta = \text{chromaAvg} - \alpha \cdot \text{lumaAvg}$. In one sub-embodiment, the inclusion order of the default candidates can depend on the absolute value and the sign of the scaling parameter α. For example, the default candidates are added into the list in the following order: $\alpha = 0,\ 1/8, -1/8, +2/8, -2/8, +3/8, -3/8, +4/8, -4/8, \ldots, +N/8, -N/8$.

[0154] In another embodiment, a default candidate can be an earlier candidate with a delta scaling parameter refinement. The earlier candidate is a CCLM model. If the scaling parameter of an earlier candidate is α, the scaling parameter of a default candidate is $(\alpha + \Delta\alpha)$. For example, $\Delta\alpha$ can be $1/8, -1/8, +2/8, -2/8, +3/8, -3/8, +4/8, -4/8, \ldots, +N/8, -N/8$, where N is a positive integer. The offset parameter β can be derived based on $(\alpha + \Delta\alpha)$ and the average values of neighbouring luma and chroma samples of the current block. In one sub-embodiment, the earlier candidate is the first CCLM candidate added into the list. In one sub-embodiment, the inclusion order of the default candidates can depend on the absolute value and the sign of the refinement Δα. For example, the default candidates are added into the list in the following order: $\Delta\alpha = 0,\ 1/8, -1/8, +2/8, -2/8, +3/8, -3/8, +4/8, -4/8, \ldots, +N/8, -N/8$.

[0155] **II.9 Reordering the Candidates in the List**

[0156] The candidates in the list can be reordered to reduce the syntax overhead when signalling the selected candidate index. The reordering rules can depend on the coding information of neighbouring blocks or the model error. For example, if neighbouring above or left blocks are coded by MMLM, the MMLM candidates in the list can be moved to the head of the current list. Similarly, if neighbouring above or left blocks are coded by single model LM or CCCM, the single model LM or CCCM candidates in the list can be moved to the head of the current list. Similarly, if GLM is used by neighbouring above or left blocks, the GLM related candidates in the list can be moved to the head of the current list.

[0157] In still another embodiment, the reordering rule is based on the model error by applying the candidate model to the neighbouring templates of the current block, and then compare the error with the reconstructed samples of the neighbouring template. For example, as shown in Fig. 11, the size of above neighbouring template of the current block is $w_a \times h_a$, and the size of left neighbouring template of the current block is $w_b \times h_b$. Suppose K models are in the current candidate list, and α_k and β_k are the final scale and offset parameters after inheriting the candidate k. The model error of candidate k corresponding to the above neighbouring template is:

$$e_a^k = \sum_{i,j} \left| \left(\alpha_k \times recL'^{(i,j)}_a + \beta_k \right) - recC_a^{(i,j)} \right|$$

[0158] where, $recL'^{(i,j)}_a$ and $recC_a^{(i,j)}$ are the reconstructed samples of luma (e.g., after downsampling process or after applying GLM pattern) and reconstructed samples of chroma at position (i, j) in the above template, and $0 \le i < w_a$ and $0 \le j < h_a$.

[0159] Similarly, the model error of candidate k by the left neighbouring template is:

$$e_b^k = \sum_{m,n} \left| \left(\alpha_k \times recL'^{(m,n)}_b + \beta_k \right) - recC_b^{(m,n)} \right|$$

[0160] where $recL'^{(m,n)}_b$ and $recC_b^{(m,n)}$ are the reconstructed samples of luma (e.g., after applying downsampling process or GLM pattern) and reconstructed samples of chroma at position (m, n) in the left template, and $0 \le m < w_b$ and $0 \le n < h_b$.

[0161] Then the model error of candidate k is:

$$e^k = e_a^k + e_b^k$$

[0162] After calculating the model error among all candidates, it can get a model error list $E = \{e^0, e^1, e^2, \dots, e^k, \dots, e^K\}$. Then, it can reorder the candidate index in the inherited candidate list by sorting the model error list in ascending order.

[0163] In still another embodiment, if the candidate k uses CCCM prediction, the e_a^k and e_b^k are defined as:

$$e_a^k = \sum_{i,j} \left| \left(c0_k \times recL'^{(i,j)}_a + c1_k \times recL'^{(i,j-1)}_a + c2_k \times recL'^{(i,j+1)}_a + c3_k \times recL'^{(i-1,j)}_a + c4_k \times recL'^{(i+1,j)}_a + c5_k \times P + c6_k \times B \right) - recC_a^{(i,j)} \right|$$

$$e_b^k = \sum_{i,j} \left| \left(c0_k \times recL'^{(i,j)}_b + c1_k \times recL'^{(i,j-1)}_b + c2_k \times recL'^{(i,j+1)}_b + c3_k \times recL'^{(i-1,j)}_b + c4_k \times recL'^{(i+1,j)}_b + c5_k \times P + c6_k \times B \right) - recC_b^{(i,j)} \right|$$

[0164] where $c0_k, c1_k, c2_k, c3_k, c4_k, c5_k$, and $c6_k$ are the final filtering coefficients after inheriting the candidate k. P and B are the nonlinear term and bias term.

[0165] In still another embodiment, if the above neighbouring template is not available, then $e^k = e_b^k$. Similarly, if the left neighbouring template is not available, then $e^k = e_a^k$. If both templates are not available, the candidate index reordering method using model error is not applied.

[0166] **II.10 Signalling the Inherited Candidate Index in the List**

[0167] An on/off flag is signalled to indicate whether the current block inherits the cross-component model parameters from neighbouring blocks or not. The flag can be signalled per CU/CB, per PU, per TU/TB, or per colour component, or per chroma colour component. A high level syntax can be signalled in SPS, PPS, PH or SH to indicate if the proposed method is allowed for the current sequence, picture, or slice.

[0168] The maximum allowed candidate number can be signalled to indicate the maximum size of the merge candidate list. The number can be signalled per CU/CB, per PU, per TU/TB, or per colour component, or per chroma colour component. A high level syntax can be signalled in SPS, PPS, PH or SH to indicate if the proposed method is allowed for the current sequence, picture, or slice. The maximum allowed candidate number of the proposed method can be shared with the maximum allowed candidate number for inter merge mode.

[0169] If the current block inherits the cross-component model parameters from neighbouring blocks, the inherited candidate index is signalled. The index can be signalled (e.g., signalled using truncate unary

code, Exp-Golomb code, or fix length code) and shared among both the current Cb and Cr blocks. For another example, the index can be signalled per colour component. For example, one inherited index is signalled for Cb component, and another inherited index is signalled for Cr component. For another example, it can use chroma intra prediction syntax (e.g., IntraPredModeC[xCb][yCb]) to store the inherited index.

[0170] **II.11 Inheriting Model Parameters from Non-Intra Blocks**

[0171] In one embodiment, when the current block is coded in inter mode or other modes (e.g., intra block copy (IBC)), the chroma samples can be predicted by applying a cross-component model to reconstructed luma samples. The cross-component model used to predict the chroma samples can be saved and/or referenced by following coding blocks.

[0172] In one sub-embodiment, the cross-component model can be derived from reconstructed luma and chroma samples. The samples can be from regions indicated by the motion vectors or the block vectors. For example, the model used can be derived using the CCRM method. All or any subset of the model parameters can be saved. If the following coding block is intra, the following block is allowed to use the saved model parameters. If the following coding block is inter or any mode-type (e.g. IBC), the following block is allowed to use the saved model parameters.

[0173] In another embodiment, the cross-component model can be inherited from neighbouring blocks, as described in earlier sub-sections under Section entitled "Inheriting neighbouring model parameters for refining the cross-component model parameters". All or any subset of the model parameters can be saved. If the following coding block is intra, the following block is allowed to use the saved model parameters. If the following coding block is inter or any mode-type (e.g. IBC), the following block is allowed to use the saved model parameters.

[0174] In one sub-embodiment, models from blocks coded in different modes can be stored in different buffers. For example, the models from intra blocks and the models from inter blocks can be stored in different buffers.

[0175] **II.12 Inheriting Multiple Cross-Component Models**

[0176] In another embodiment, it can combine cross-component model candidates with the prediction by non-cross-component coding tools. For example, one cross-component model candidate is selected from the list, and its prediction is denoted as p_ccm. Another prediction, denoted as p_(non-ccm), can be from chroma DM, chroma DIMD, or intra angular mode. The final prediction at (x, y) position of the current block is $p_final(x,y)=(1-\alpha)\times p_ccm(x,y)+\alpha\times p_(non\text{-}ccm)(x,y)$, where α is the weighting factor which can be predefined or implicitly derived by neighbouring template cost. For still the same example, the prediction by non-cross-component coding tool can be predefined or signalled. The prediction by non-cross-component coding tool is chroma DM or chroma DIMD. For another example, prediction by non-cross-component coding tool is signalled, but the index of cross-component model candidate is predefined or determined by neighbouring blocks coding mode. For still the same example, if at least one of neighbouring spatial blocks is coded with CCCM mode, the first candidate has CCCM model parameters is selected. If at least one of neighbouring spatial blocks is coded with GLM mode, the first candidate has GLM pattern parameters is selected. Similarly, if at least one of neighbouring spatial blocks is coded with MMLM mode, the first candidate has MMLM parameters is selected.

[0177] Any of the foregoing proposed methods of generating inter predictor for the chroma component by applying a cross-component model to reconstructed luma samples can be implemented in encoders and/or decoders. For example, any of the proposed methods can be implemented in an inter/intra/prediction module

of an encoder, and/or an inter/intra/prediction module of a decoder. Alternatively, any of the proposed methods can be implemented as circuits coupled to the inter/intra/prediction module of the encoder and/or the inter/intra/prediction module of the decoder, so as to provide the information needed by the inter/intra/prediction module

[0178] The method of generating inter predictor for the chroma component by applying a cross-component model to reconstructed luma samples can be implemented in an encoder side or a decoder side. For example, any of the proposed c method can be implemented in an Intra/Inter coding module (e.g. Intra Pred. 150/MC 152 in Fig. 1B) in a decoder or an Intra/Inter coding module in an encoder (e.g. Intra Pred. 110/Inter Pred. 112 in Fig. 1A). Any of the proposed candidate derivation method can also be implemented as a circuit coupled to the intra/inter coding module at the decoder or the encoder. However, the decoder or encoder may also use additional processing unit to implement the required cross-component prediction processing. While the Intra Pred. units (e.g. unit 110/112 in Fig. 1A and unit 150/152 in Fig. 1B) are shown as individual processing units, they may correspond to executable software or firmware codes stored on a media, such as hard disk or flash memory, for a CPU (Central Processing Unit) or programmable devices (e.g. DSP (Digital Signal Processor) or FPGA (Field Programmable Gate Array)).

[0179] Fig. 12 illustrates a flowchart of an exemplary video coding system that generates inter predictor for the chroma component by applying a cross-component model to reconstructed luma samples 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 a first-colour block and a second-colour block is received in step 1210, wherein the input data comprise pixel data to be encoded at an encoder side or data associated with the current block to be decoded at a decoder side, and wherein the current block is coded in a non-intra mode. A cross-component model for the current block is determined in step 1220. A target cross-component predictor is derived by applying the cross-component model to reconstructed first-colour samples in step 1230. The second-colour block is encoded or decoded by using prediction data comprising the target cross-component predictor in step 1240.

[0180] 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.

[0181] 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 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.

[0182] 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 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 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 means of configuring code to perform the tasks in accordance with the invention will not depart from the spirit and scope of the invention.

[0183] 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 their scope.

CLAIMS

1. A method of coding colour pictures using coding tools including one or more cross component models related modes, the method comprising:

receiving input data associated with a current block comprising a first-colour block and a second-colour block, wherein the input data comprise pixel data to be encoded at an encoder side or data associated with the current block to be decoded at a decoder side, and wherein the current block is coded in a non-intra mode;

determining a cross-component model for the current block;

deriving a target cross-component predictor by applying the cross-component model to reconstructed first-colour samples; and

encoding or decoding the second-colour block by using prediction data comprising the target cross-component predictor.

2. The method of Claim 1, wherein the cross-component model for the current block is stored, referenced by following coding blocks, or both.

3. The method of Claim 2, wherein all model parameters of the cross-component model are stored.

4. The method of Claim 2, wherein a subset of model parameters of the cross-component model is stored.

5. The method of Claim 2, wherein if a target following coding block is coded in an intra mode, the target following coding block is allowed to use the cross-component model stored.

6. The method of Claim 2, wherein if a target following coding block is coded in the non-intra mode, the target following coding block is allowed to use the cross-component model stored.

7. The method of Claim 1, wherein the cross-component model is derived from reconstructed first-colour samples and reconstructed second-colour samples.

8. The method of Claim 7, wherein the reconstructed first-colour samples and the reconstructed second-colour samples are from regions indicated by a motion vector or a block vector.

9. The method of Claim 1, wherein the cross-component model is derived based on CCRM (Cross-Component Residual Model).

10. The method of Claim 1, wherein the cross-component model is inherited from a neighbouring block.

11. The method of Claim 10, wherein the neighbouring block corresponds to a spatial neighbouring block or a temporal neighbouring block.

12. The method of Claim 1, wherein different cross-component models associated with blocks coded in different modes are stored in different buffers.

13. The method of Claim 12, wherein a first cross-component model associated with intra-coded blocks and a second cross-component model associated with inter-coded blocks are stored in two different buffers.

14. An apparatus for coding colour pictures using coding tools including one or more cross component models related modes, the apparatus comprising one or more electronic circuits or processors arranged to:

receive input data associated with a current block comprising a first-colour block and a second-colour block, wherein the input data comprise pixel data to be encoded at an encoder side or data associated with the current block to be decoded at a decoder side, and wherein the current block is coded in a non-intra mode;

determine a cross-component model for the current block;

derive a target cross-component predictor by applying the cross-component model to reconstructed first-colour samples; and

encode or decode the second-colour block by using prediction data comprising the target cross-component predictor.

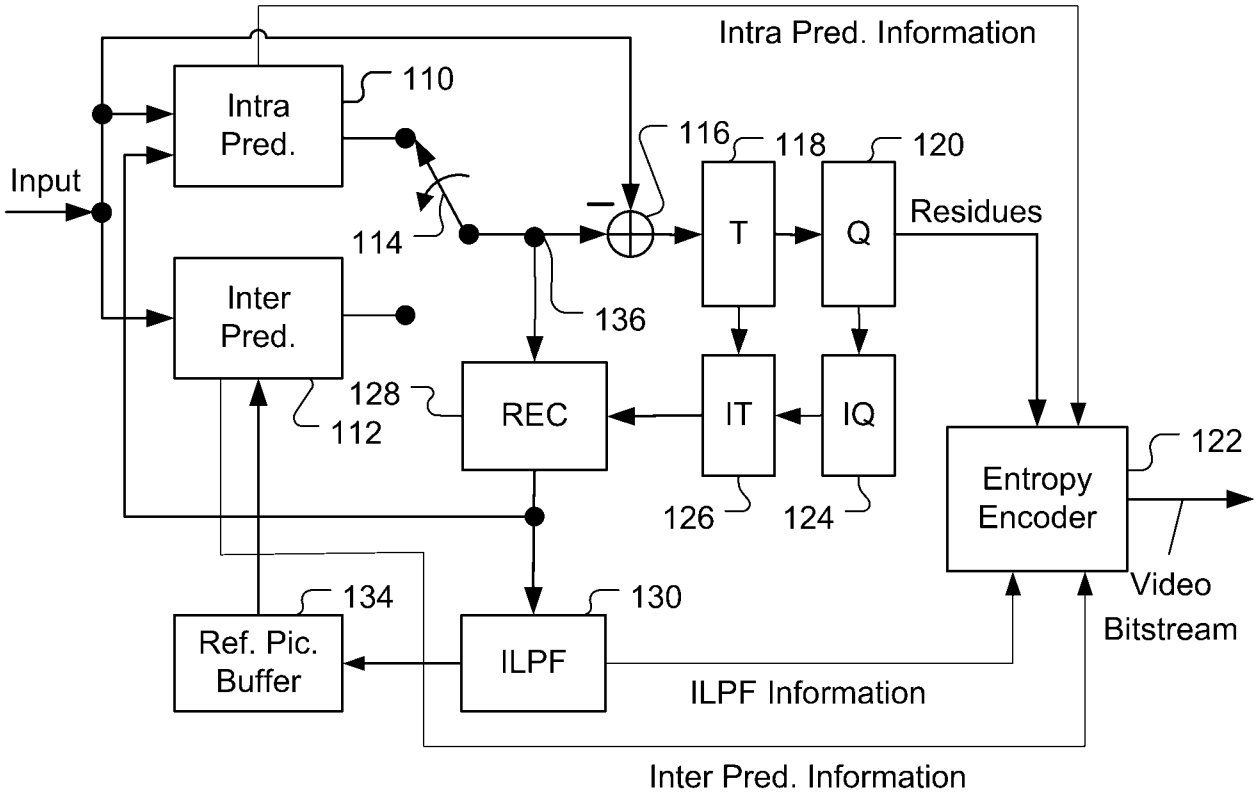

Intra Pred. Information
Intra Pred.
110
Input
116
118
120
T
Q
Residues
114
Inter Pred.
136
128
112
REC
IT
IQ
122
Entropy Encoder
126
124
134
130
Ref. Pic. Buffer
ILPF
Video Bitstream
ILPF Information
Inter Pred. Information

Fig. 1A

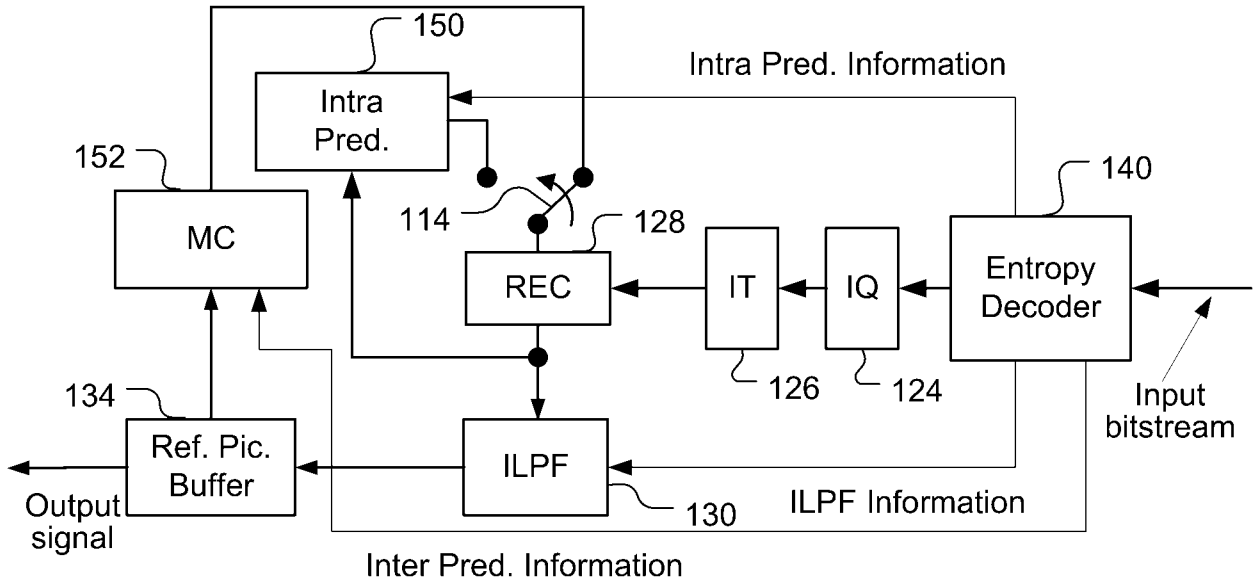

150
Intra Pred. Information
Intra Pred.
152
140
MC
114
128
REC
IT
IQ
Entropy Decoder
126
124
134
Input bitstream
Ref. Pic. Buffer
ILPF
Output signal
130
ILPF Information
Inter Pred. Information

Fig. 1B

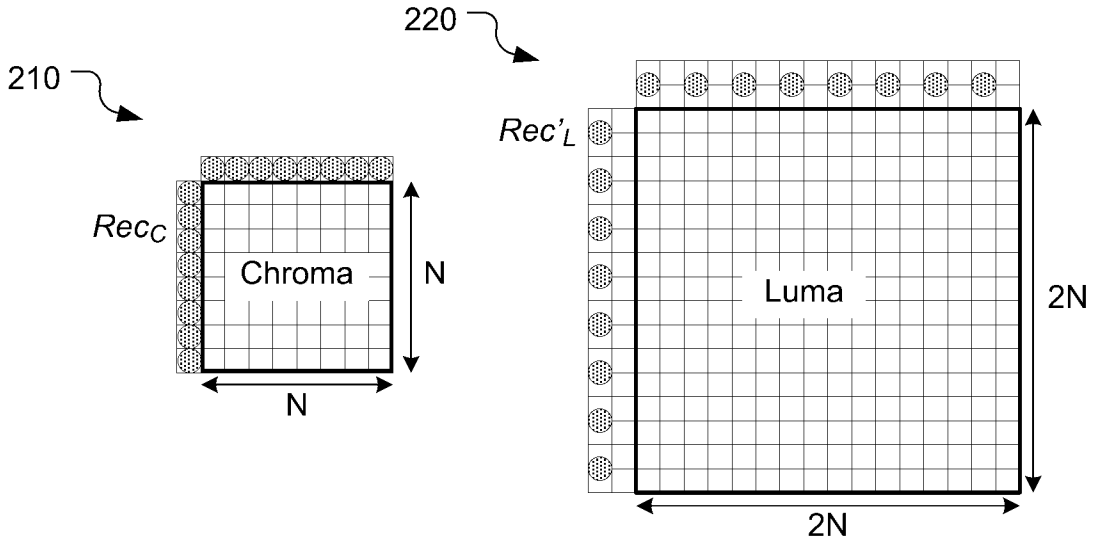

220
210
Rec'L
RecC
Chroma
N
N
Luma
2N
2N

Fig. 2

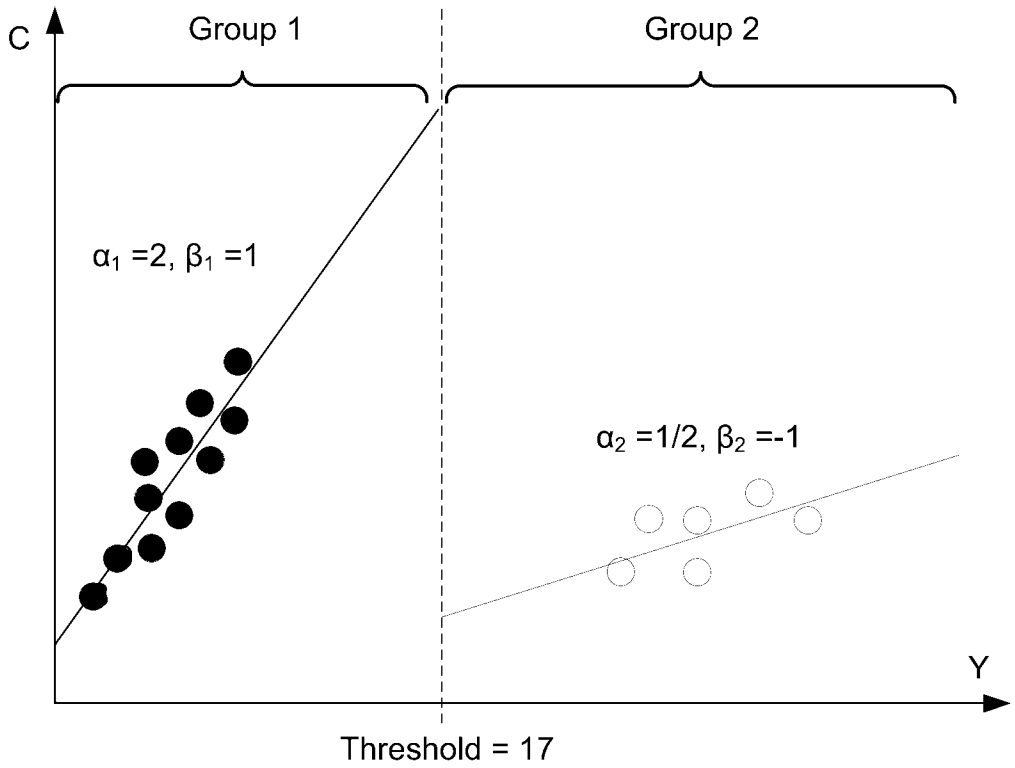

C
Group 1
Group 2
α1 =2, β1 =1
α2 =1/2, β2 =-1
Y
Threshold = 17

Fig. 3

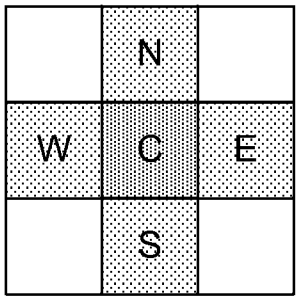


Fig. 4

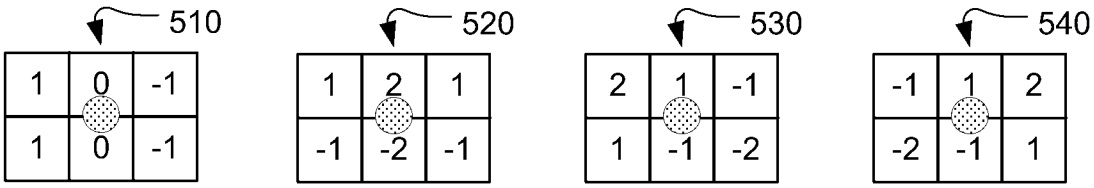


Fig. 5

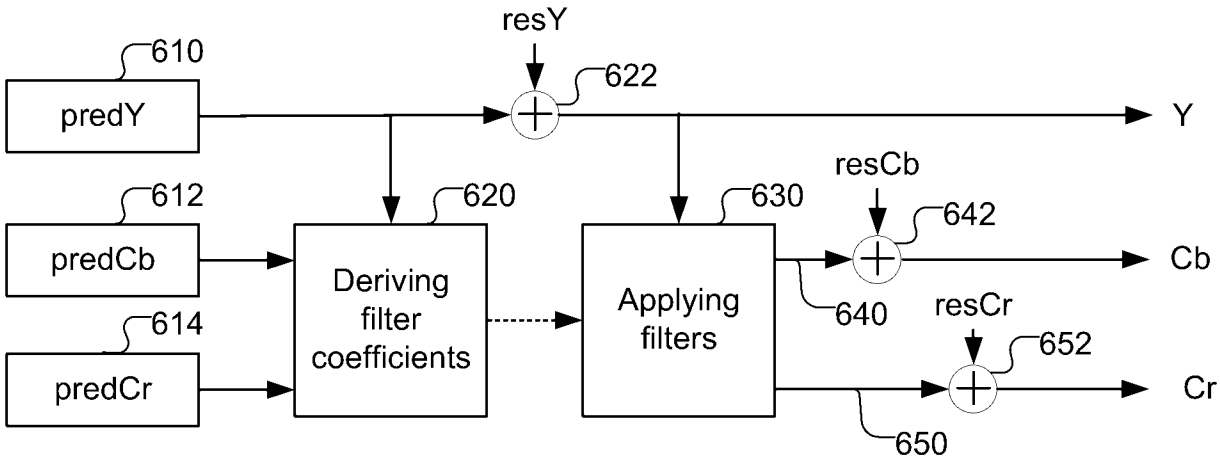


Fig. 6

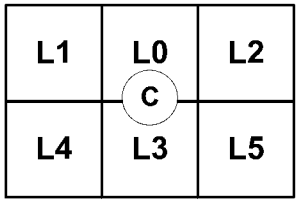


Fig. 7

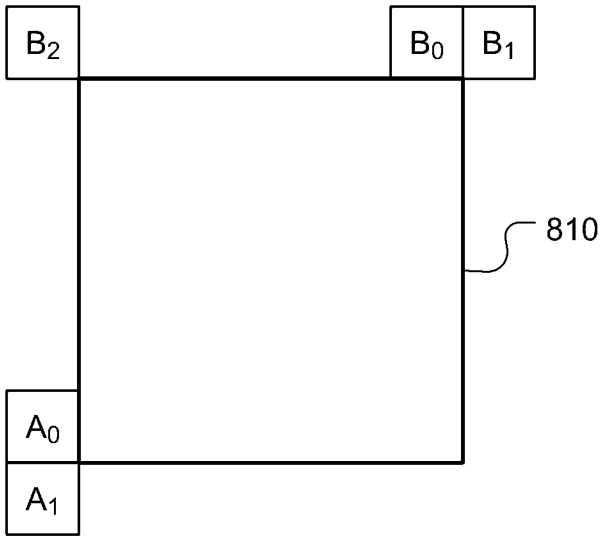


Fig. 8

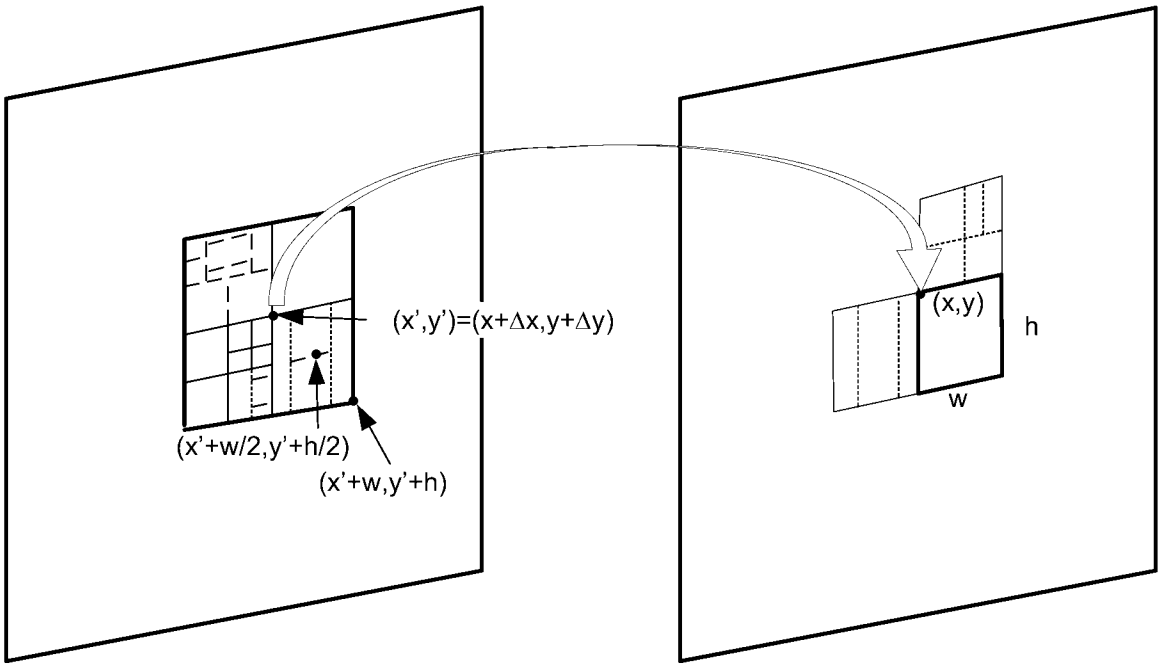


Fig. 9

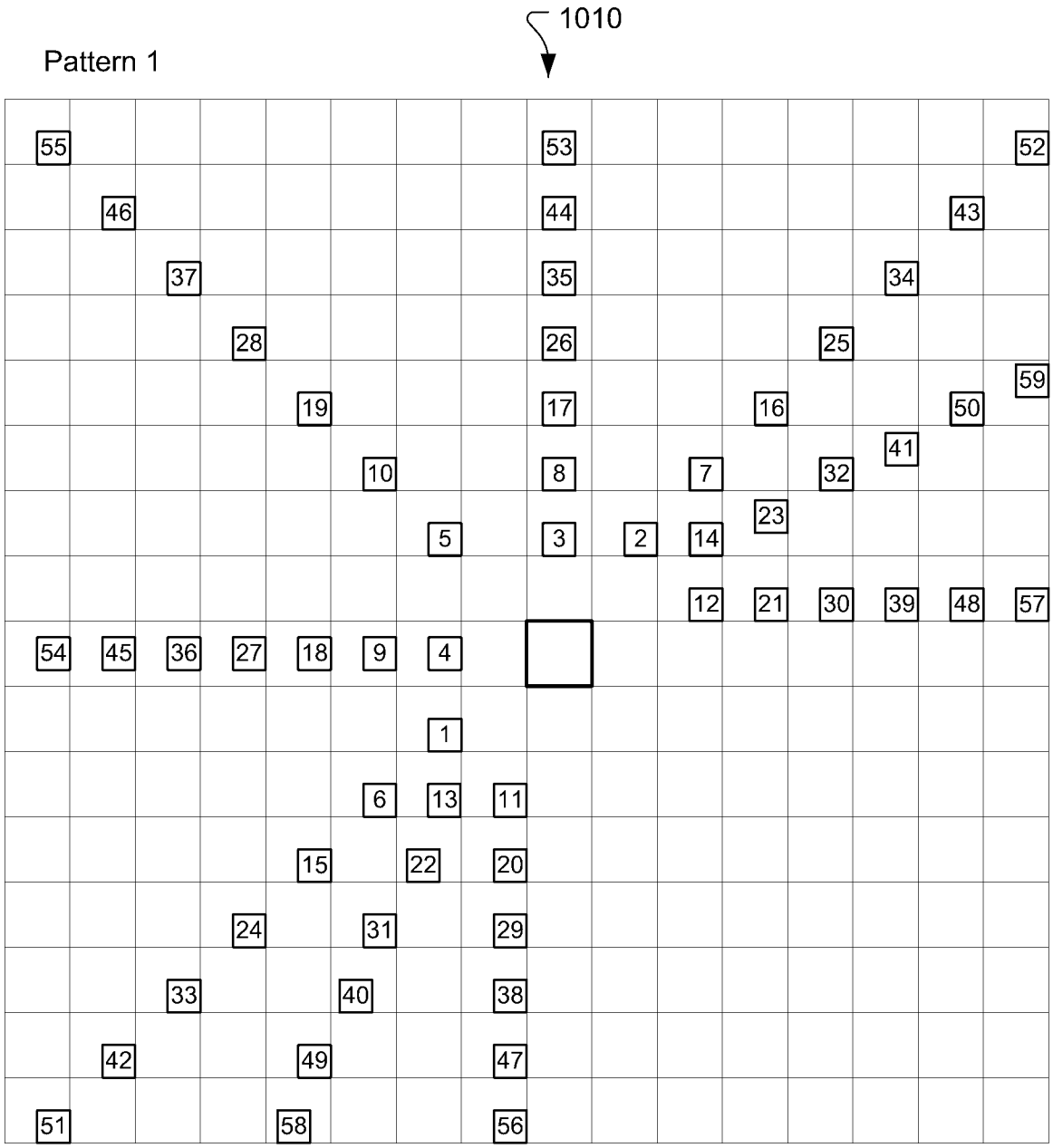

Pattern 1
1010
55
53
52
46
44
43
37
35
34
28
26
25
59
19
17
16
50
41
10
8
7
32
23
5
3
2
14
12
21
30
39
48
57
54
45
36
27
18
9
4
1
6
13
11
15
22
20
24
31
29
33
40
38
42
49
47
51
58
56

Fig. 10A

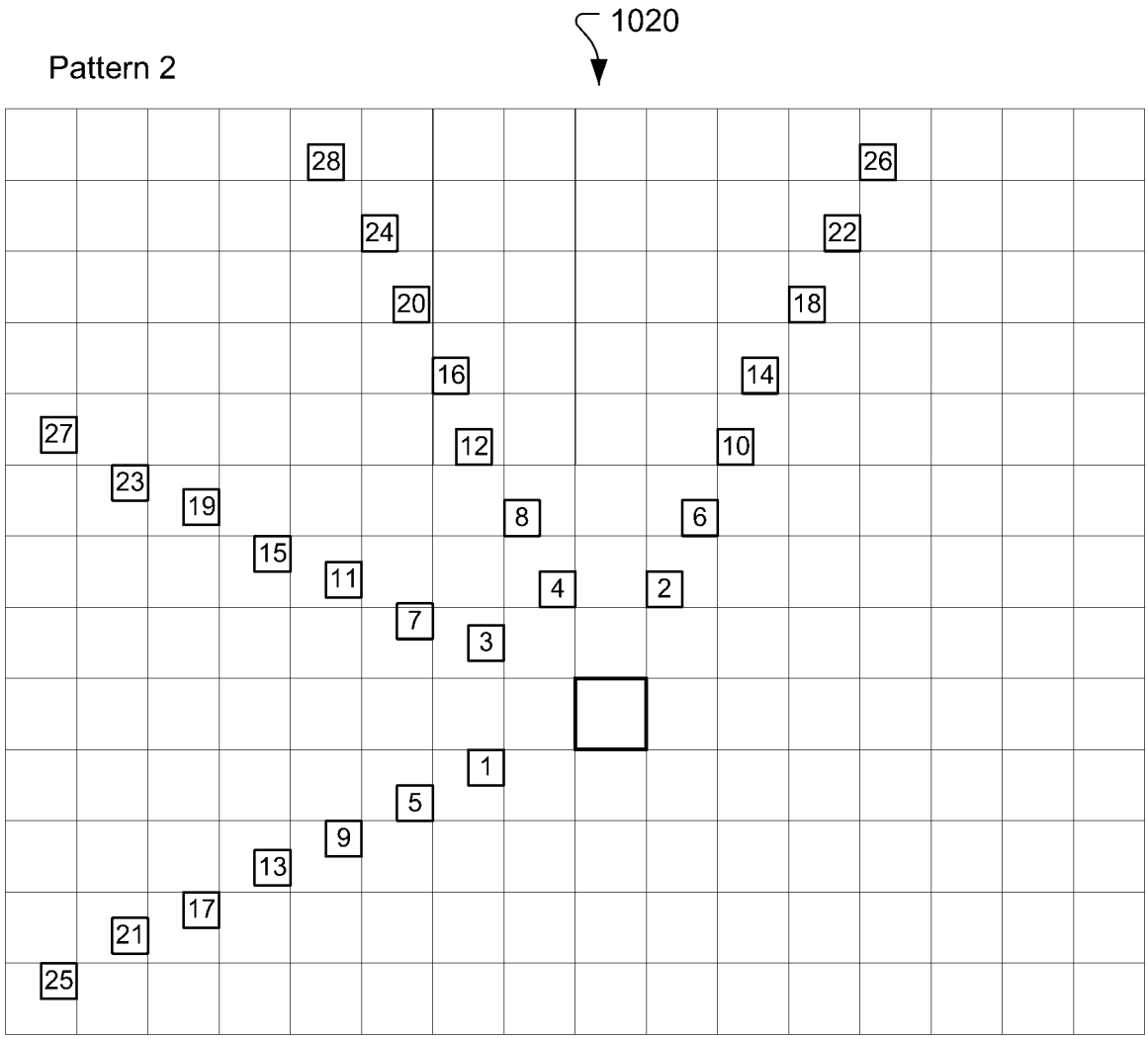

Pattern 2
1020
28
26
24
22
20
18
16
14
27
12
10
23
19
8
6
15
11
4
2
7
3
1
5
9
13
17
21
25

Fig. 10B

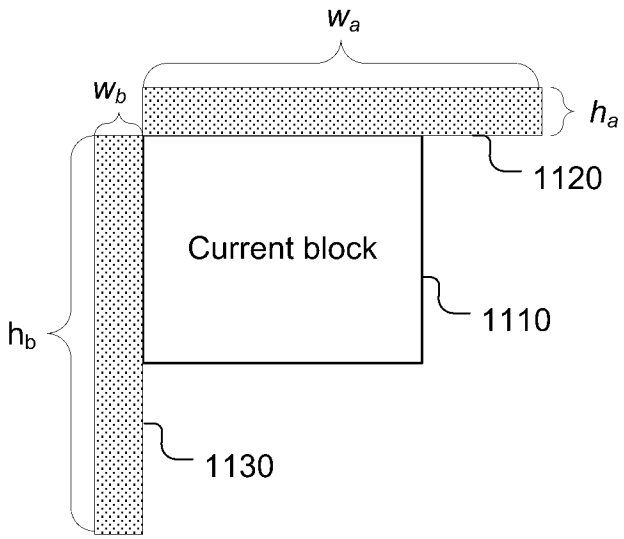


Fig. 11

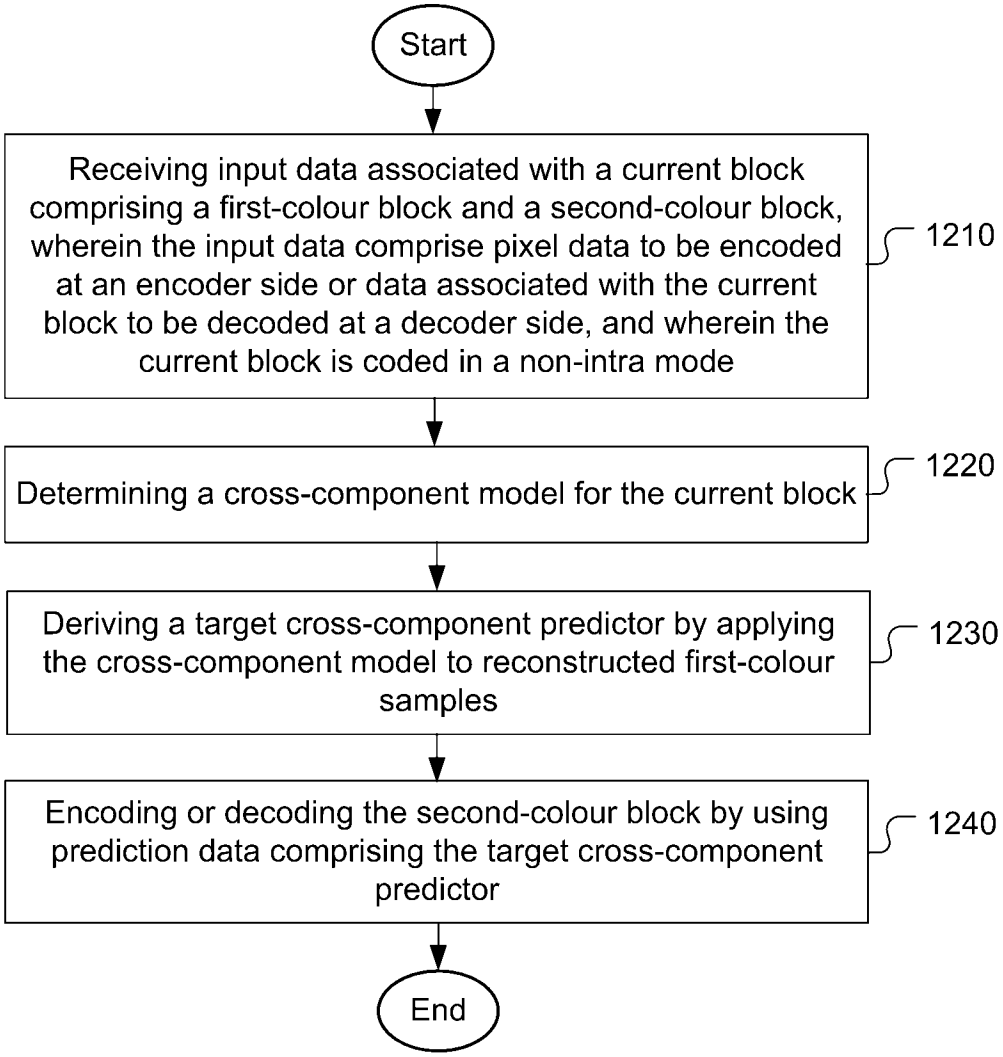

Start
Receiving input data associated with a current block comprising a first-colour block and a second-colour block, wherein the input data comprise pixel data to be encoded at an encoder side or data associated with the current block to be decoded at a decoder side, and wherein the current block is coded in a non-intra mode
1210
Determining a cross-component model for the current block
1220
Deriving a target cross-component predictor by applying the cross-component model to reconstructed first-colour samples
1230
Encoding or decoding the second-colour block by using prediction data comprising the target cross-component predictor
1240
End

Fig. 12

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2024/097398

A. CLASSIFICATION OF SUBJECT MATTER

H04N19/186(2014.01)i; H04N19/50(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,DWPI,JVET,JCTVC:CCLM, CCP, CCRM, cross 1w component, predict, reconstruct, colo?r, second, chroma, residual, edcod+, cod+

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2021297680 A1 (TELEFONAKTIEBOLAGET LM ERICSSON(PUBL)) 23 September 2021 (2021-09-23) description paragraph 109	1-14
Y	US 2022014772 A1 (GUANGDONG OPPO MOBILE TELECOMMUNICATIONS CORP., LTD.) 13 January 2022 (2022-01-13) description paragraphs 23, 100-101, FIG5	1-14
A	US 2016219283 A1 (QUALCOMM INCORPORATED) 28 July 2016 (2016-07-28) The whole document	1-14
A	WO 2019069950 A1 (SHARP KABUSHIKI KAISHA et al.) 11 April 2019 (2019-04-11) The whole document	1-14
A	WO 2022191553 A1 (HYUNDAI MOTOR COMPANY et al.) 15 September 2022 (2022-09-15) The whole document	1-14

☑ 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	Date of mailing of the international search report
13 August 2024	**28 August 2024**
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 **ZHANG,LuWen** Telephone No. **(+86) 010-53961724**

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2024/097398

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	ZHANG, Kai et al. "Enhanced Cross-component Linear Model Intra-prediction (JVET-D0110)" *Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 4th Meeting: Chengdu, CN, 15–21 October 2016*, 17 October 2016 (2016-10-17), The whole document	1-14

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2024/097398

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2021297680	A1	23 September 2021	WO	2018070914	A1	19 April 2018
				EP	3526968	A1	21 August 2019
US	2022014772	A1	13 January 2022	KR	20210139328	A	22 November 2021
				JP	2024095842	A	10 July 2024
				JP	2022528635	A	15 June 2022
				WO	2020192085	A1	01 October 2020
				EP	3944621	A1	26 January 2022
US	2016219283	A1	28 July 2016	CA	2972501	A1	04 August 2016
				TW	201640895	A	16 November 2016
				TN	2017000284	A1	16 January 2019
				JP	2018509035	A	29 March 2018
				EP	3092803	A1	16 November 2016
				EA	201791476	A1	30 November 2017
				KR	20170107448	A	25 September 2017
				WO	2016123219	A1	04 August 2016
WO	2019069950	A1	11 April 2019	US	2020260096	A1	13 August 2020
				US	2021409745	A1	30 December 2021
				US	2023021748	A1	26 January 2023
				JP	2021010046	A	28 January 2021
WO	2022191553	A1	15 September 2022	US	2023421752	A1	28 December 2023