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

(19) **World Intellectual Property Organization**
International Bureau

(43) **International Publication Date**
24 October 2019 (24.10.2019)

WIPO | PCT

(10) **International Publication Number**
WO 2019/201232 A1

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

(21) **International Application Number:**
PCT/CN2019/082842

(22) **International Filing Date:**
16 April 2019 (16.04.2019)

(25) **Filing Language:** English

(26) **Publication Language:** English

(30) **Priority Data:**
62/658,316 16 April 2018 (16.04.2018) US

(71) **Applicant: HUAWEI TECHNOLOGIES CO., LTD.** [CN/CN]; Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong 518129 (CN).

(72) **Inventors: MA, Xiang**; Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong 518129 (CN). **CHEN, Jianle**; 10756 Corte De Tiburon, San Diego, CA 92130 (US). **ZHAO, Yin**; Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong 518129 (CN). **YANG, Haitao**; Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong 518129 (CN).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:
— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

(54) **Title:** INTRA-PREDICTION USING CROSS-COMPONENT LINEAR MODEL

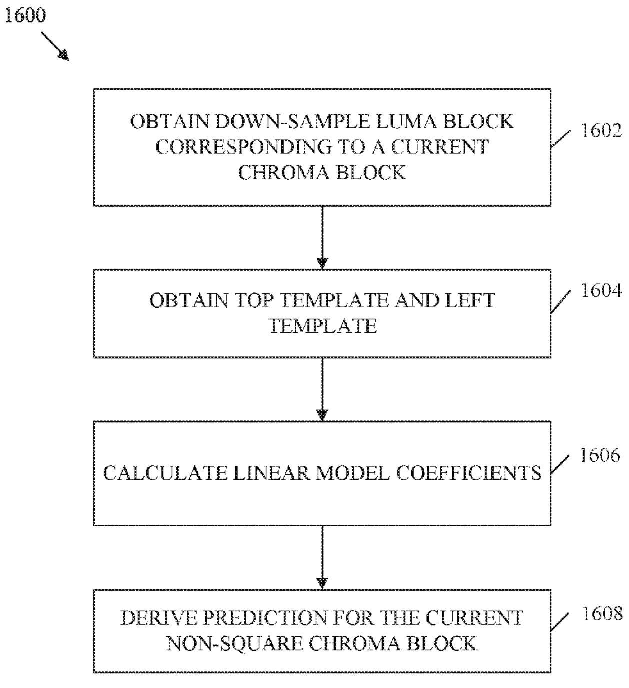

FIG. 16

(57) **Abstract:** A method implemented in an encoder for deriving a prediction for a current chroma block using cross-component intra prediction (CCIP) mode. The method obtains a down-sample luma block corresponding to the current chroma block. The method obtains a top template and a left template. The method extends one of the top template and the left template to be equal in size to a long side size (LSS) such that both the top template and the left template are equal in size to the LSS. The method calculates the linear model coefficients based on a number of reference samples of both the top template and the left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

WO 2019/201232 A1

[Continued on next page]

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

INTRA-PREDICTION USING CROSS-COMPONENT LINEAR MODEL

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This patent application claims the benefit of U.S. Provisional Patent Application No. 62/658,316, filed April 16, 2018 by Xiang Ma, et. al., and titled “Intra Prediction Using Cross-Component Linear Model,” which is hereby incorporated by reference.

TECHNICAL FIELD

[0002] The present disclosure is generally related to video coding, and is specifically related to intra-prediction as part of a video coding mechanism.

BACKGROUND

[0003] The amount of video data needed to depict even a relatively short video can be substantial, which may result in difficulties when the data is to be streamed or otherwise communicated across a communications network with limited bandwidth capacity. Thus, video data is generally compressed before being communicated across modern day telecommunications networks. The size of a video could also be an issue when the video is stored on a storage device because memory resources may be limited. Video compression devices often use software and/or hardware at the source to code the video data prior to transmission or storage, thereby decreasing the quantity of data needed to represent digital video images. The compressed data is then received at the destination by a video decompression device that decodes the video data. With limited network resources and ever increasing demands of higher video quality, improved compression and decompression techniques that improve compression ratio with little to no sacrifice in image quality are desirable.

SUMMARY

[0004] This disclosure describes improved methods and apparatuses for intra prediction coding using cross-component linear model.

[0005] A first aspect relates to a method in an encoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using cross-component intra prediction (CCIP) mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. The method obtains a top template and a left template, wherein one of the top template and the left template is extended to be equal in size to a long side size (LSS) such that both the top template and the left template are equal in size to the LSS, and wherein the LSS is the larger value of the W and the H. The method calculates linear model coefficients of a linear model based on a number of reference samples of (or in) both the top template and the left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0006] It should be noted that the number of reference samples of/in both the top template and the left template, in an example, may mean all the reference samples in the top template and the left template. In another example, the number of reference samples of/in the top template and left template, may mean the partial number of the reference samples in the top template and the left template.

[0007] In another example, all the reference samples in the top template and/or the left template may be used to calculate linear model coefficients of a linear model. In another example, the partial number of the reference samples in the top template and/or the left template may be used to calculate linear model coefficients of a linear model.

[0008] A second aspect relates to a method in an encoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template, wherein the top template is extended to be equal in size to a long side size (LSS) to obtain an extended top template if the top template is not equal in size to the LSS, and wherein the LSS is the larger value of the W and the H. For the CCIP_L mode, the method obtains a left template, wherein the left template is extended to be equal in size to the LSS to obtain an extended left template if the left template is not equal in size to the LSS. For the CCIP_A mode, if the top template is equal in size to the LSS, the method calculates linear model coefficients of a linear model based on the number of reference samples of the top template, otherwise, if the top template is not equal in size to the LSS, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended top template. For the CCIP_L mode, if the left template is equal in size to the LSS, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if the left template is not equal in size to the LSS, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0009] A third aspect relates to a method in an encoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using cross-component intra prediction (CCIP) mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. The method obtains a top template and a left template, wherein a sum of the number of reference samples of the top template and the left template is equal to a smallest integer value (M), wherein M is not smaller than W+H, and wherein M is a value of power of 2. The method calculates the linear model coefficients of a linear model based on a number of reference samples of both the top template and the left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0010] A fourth aspect relates to a method in an encoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template, wherein the top template is extended to be equal in size to a smallest integer value (M) not smaller than W+H to obtain an extended top template. For the CCIP_L mode, the method obtains a left template, wherein the left template is extended to be equal in size to the smallest integer value (M) not smaller than W+H to obtain an extended left template. The method calculates linear model coefficients of a linear model based on a number of reference samples of the extended top template in the CCIP_A mode, or the method calculates linear model coefficients of a linear model based on the number of reference samples of the extended left template in the CCIP_L mode. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0011] A fifth aspect relates to a method implemented in an encoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template of size W, and, if W is less than H, the top template is extended to be double the width (2W) to obtain an extended top template. For the CCIP_L mode, the method obtains a left template of size H, and, if H is less than W, the left template is extended to be double the height (2H) to obtain an extended left template. For the CCIP_A mode, if W is greater than H, the method calculates linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended top template. or For the CCIP_L mode, if H is greater than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0012] A sixth aspect relates to a method implemented in an encoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template of size W, and, if W is less than H, the top template is extended to a minimum value of H and double of W (min (H, 2W)) to obtain an extended top template. For the CCIP_L mode, the method obtains a left template of size H, and, if H is less than W, the left template is extended to a minimum value of W and double of H (min (W, 2H)) to obtain an extended left template. For the CCIP_A mode, if W is greater than H, the method calculates linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended top template. For the CCIP_L mode, if H is greater than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0013] A seventh aspect relates to a method implemented in an encoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template of size W, and, if W is less than H, the top template is extended to a maximum value of H and double of W (max (H, 2W)) to obtain an extended top template. For the CCIP_L mode, the method obtains a left template of size H, and, if H is less than W, the left template is extended to a maximum value of W and double of H (max (W, 2H)) to obtain an extended left template. For the CCIP_A mode, if W is greater than H, the method calculates the linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended top template. For the CCIP_L mode, if H is greater than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0014] An eighth aspect relates to a method implemented in an encoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template of size W, and, if W is less than H, the top template is extended to double the width (2W) to obtain an extended top template, and, if W is greater than H, a first set of samples of size H is discarded from a first end of the top template near a short side and a second set of samples of size H is added to a second end of the top template to obtain a modified top template. For the CCIP_L mode, the method obtains a left template of size H, and, if H is less than W, the left template is extended to double the height (2H) to obtain an extended left template, and, if H is greater than W, a first set of samples of size W is discarded from a first end of the left template near a long side and a second set of samples of size W is added to a second end of the left template to obtain a modified left template. For the CCIP_A mode, if W is less than H, the method calculates linear model coefficients of a linear model based on a number of reference samples of the extended top template, otherwise, if W is greater than H, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the modified top template. For the CCIP_L mode, if H is less than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template, otherwise, if H is greater than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the modified left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0015] A ninth aspect relates to a method in a decoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using cross-component intra prediction (CCIP) mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. The method obtains a top template and a left template, wherein one of the top template and the left template is extended to be equal in size to a long side size (LSS) such that both the top template and the left template are equal in size to the LSS, and wherein the LSS is the larger value of the W and the H. The method calculates linear model coefficients of a linear model based on a number of reference samples of both the top template and the left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0016] A tenth aspect relates to a method in a decoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template, wherein the top template is extended to be equal in size to a long side size (LSS) to obtain an extended top template if the top template is not equal in size to the LSS, and wherein the LSS is the larger value of the W and the H. For the CCIP_L mode, the method obtains a left template, wherein the left template is extended to be equal in size to the LSS to obtain an extended left template if the left template is not equal in size to the LSS. For the CCIP_A mode, if the top template is equal in size to the LSS, the method calculates linear model coefficients of a linear model based on the number of reference samples of the top template, otherwise, if the top template is not equal in size to the LSS, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended top template. For the CCIP_L mode, if the left template is equal in size to the LSS, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if the left template is not equal in size to the LSS, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0017] An eleventh aspect relates to a method in a decoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using cross-component intra prediction (CCIP) mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. The method obtains a top template and a left template, wherein a sum of the number of reference samples of the top template and the left template is equal to a smallest integer value (M), wherein M is not smaller than W+H, and wherein M is a value of power of 2. The method calculates the linear model coefficients of a linear model based on a number of reference samples of both the top template and the left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0018] A twelfth aspect relates to a method in a decoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template, wherein the top template is extended to be equal in size to a smallest integer value (M) not smaller than W+H to obtain an extended top template. For the CCIP_L mode, the method obtains a left template, wherein the left template is extended to be equal in size to the smallest integer value (M) not smaller than W+H to obtain an extended left template. The method calculates linear model coefficients of a linear model based on a number of reference samples of the extended top template in the CCIP_A mode, or the method calculates linear model coefficients of a linear model based on the number of reference samples of the extended left template in the CCIP_L mode. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0019] A thirteenth aspect relates to a method implemented in a decoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template of size W, and, if W is less than H, the top template is extended to be double the width (2W) to obtain an extended top template. For the CCIP_L mode, the method obtains a left template of size H, and, if H is less than W, the left template is extended to be double the height (2H) to obtain an extended left template. For the CCIP_A mode, if W is greater than H, the method calculates linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended top template. Or for the CCIP_L mode, if H is greater than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0020] A fourteenth aspect relates to a method implemented in a decoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template of size W, and, if W is less than H, the top template is extended to a minimum value of H and double of W (min (H, 2W)) to obtain an extended top template. For the CCIP_L mode, the method obtains a left template of size H, and, if H is less than W, the left template is extended to a minimum value of W and double of H (min (W, 2H)) to obtain an extended left template. For the CCIP_A mode, if W is greater than H, the method calculates linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended top template. For the CCIP_L mode, if H is greater than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0021] A fifteenth aspect relates to a method implemented in a decoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template of size W, and, if W is less than H, the top template is extended to a maximum value of H and double of W (max (H, 2W)) to obtain an extended top template. For the CCIP_L mode, the method obtains a left template of size H, and, if H is less than W, the left template is extended to a maximum value of W and double of H (max (W, 2H)) to obtain an extended left template. For the CCIP_A mode, if W is greater than H, the method calculates the linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended top template. For the CCIP_L mode, if H is greater than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0022] An sixteenth aspect relates to a method implemented in a decoder for intra prediction for a current chroma block (such as, that derives a prediction for a current chroma block) using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode. The method obtains a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal. For the CCIP_A mode, the method obtains a top template of size W, and, if W is less than H, the top template is extended to double the width (2W) to obtain an extended top template, and, if W is greater than H, a first set of samples of size H is discarded from a first end of the top template near a short side and a second set of samples of size H is added to a second end of the top template to obtain a modified top template. For the CCIP_L mode, the method obtains a left template of size H, and, if H is less than W, the left template is extended to double the height (2H) to obtain an extended left template, and, if H is greater than W, a first set of samples of size W is discarded from a first end of the left template near a long side and a second set of samples of size W is added to a second end of the left template to obtain a modified left template. For the CCIP_A mode, if W is less than H, the method calculates linear model coefficients of a linear model based on a number of reference samples of the extended top template, otherwise, if W is greater than H, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the modified top template. For the CCIP_L mode, if H is less than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the extended left template, otherwise, if H is greater than W, the method calculates the linear model coefficients of the linear model based on the number of reference samples of the modified left template. The method derives a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

[0023] A seventeenth through twenty-fourth aspect relates to a video encoding device having a processor; and memory coupled to the processor, wherein the processor is configured to execute instructions in the memory to perform the method according to any of the preceding first through eighth aspects.

[0024] A twenty-fifth through thirty-second aspect relates to a video decoding device having a processor; and memory coupled to the processor, wherein the processor is configured to execute instructions in the memory to perform the method according to any of the preceding ninth through sixteenth aspects.

[0025] In a first implementation form according to any of the preceding aspect as such, the top template is top neighboring reconstructed chroma samples and a corresponding down-sampled top neighboring reconstructed luma samples, and the left template is left neighboring reconstructed chroma samples and a corresponding down-sampled left neighboring reconstructed luma samples.

[0026] In a second implementation form according to any of the preceding aspect as such or any preceding implementation form of any preceding aspect, the number of reference samples of both the top template and the left template used in calculating the linear model coefficients is a value of power of 2.

[0027] In a third implementation form according to any of the preceding aspect as such or any preceding implementation form of any preceding aspect, the linear model is as follows:

[0028] $\mathrm{pred_C}(i,j) = \alpha \cdot \mathrm{rec_L}'(i,j) + \beta,$

[0029] where $\mathrm{pred_C}(i,j)$ represents predicted chroma samples of a coding unit (CU) and $\mathrm{rec_L}(i,j)$ represents down-sampled reconstructed luma samples of the same CU., and wherein the parameters α and β are the linear model coefficients.

[0030] In a fourth implementation form according to any of the preceding aspect as such or any preceding implementation form of any preceding aspect, the linear model coefficients, parameters α and β, are calculated by minimizing a regression error between neighboring reconstructed luma and chroma samples around a current luma block and the current non-squared chroma block as follows:

[0031] $$\alpha = \frac{N \cdot \sum(L(n) \cdot C(n)) - \sum L(n) \cdot \sum C(n)}{N \cdot \sum(L(n) \cdot L(n)) - \sum L(n) \cdot \sum L(n)}$$

[0032] $$\beta = \frac{\sum C(n) - \alpha \cdot \sum L(n)}{N}$$

[0033] where $L(n)$ represents down-sampled top and left neighboring reconstructed luma samples, $C(n)$ represents top and left neighboring reconstructed chroma samples, and value of N is equal to the number of reference samples of both the top template and the left template used to derive the linear model coefficients.

[0034] In a fifth implementation form according to any of the preceding aspect as such or any preceding implementation form of any preceding aspect, the number of reference samples of the top template and the left template (or the extended top template or extended left template) refers to some or all of the reference samples in the respective template. For examples, in some implementations, a step size (e.g., step size can be 1 or 2) can be used to reduce the number of samples of the respective template, but still ensure that the number of reference samples used to derive the linear model coefficients of a linear model is a power of 2. The reduction in the number of samples of the respective template may provide the added benefit of reducing the complexity of the preceding aspects.

[0035] A thirty-third aspect relates to a computer-readable storage medium having stored thereon instructions that when executed cause one or more processors configured to code or decode video data is proposed. The instructions cause the one or more processors to perform a method according to the first through the sixteenth aspect or any possible embodiment of the first through the sixteenth aspect.

[0036] In another aspect, an apparatus includes a non-transitory computer readable storage medium encoded with a computer program. The program includes instructions that when executed by the one or more data processing apparatus cause the one or more data processing apparatus to perform operations according to any of the preceding aspects one through sixteenth

[0037] For the purpose of clarity, any one of the foregoing embodiments may be combined with any one or more of the other foregoing embodiments to create a new embodiment within the scope of the present disclosure.

[0038] These and other features will be more clearly understood from the following detailed description taken in conjunction with the accompanying drawings and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0039] For a more complete understanding of this disclosure, reference is now made to the following brief description, taken in connection with the accompanying drawings and detailed description, wherein like reference numerals represent like parts.

[0040] FIG. 1 is a flowchart of an example method of coding a video signal.

[0041] FIG. 2 is a schematic diagram of an example coding and decoding (codec) system for video coding.

[0042] FIG. 3 is a schematic diagram illustrating an example video encoder that may perform cross-component intra-prediction.

[0043] FIG. 4 is a schematic diagram illustrating an example video decoder that may perform cross-component intra-prediction.

[0044] FIG. 5 is a schematic diagram illustrating an example of intra-prediction modes.

[0045] FIG. 6 is a schematic diagram illustrating an example mechanism of performing cross-component intra-prediction.

[0046] FIG. 7 is a schematic diagram illustrating an example mechanism of performing cross-component linear model (CCLM) intra-prediction.

[0047] FIG. 8 is a schematic diagram illustrating a top template and a left template in accordance with an embodiment.

[0048] FIGS. 9-10 are schematic diagrams illustrating an example mechanism of performing multi-directional linear model (MDLM) intra-prediction.

[0049] FIG. 11 illustrates a first embodiment for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure

[0050] FIG. 12 illustrates a second embodiment for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure.

[0051] FIG. 13 illustrates a third embodiment for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure.

[0052] FIG. 14 illustrates a fourth embodiment for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure.

[0053] FIG. 15 illustrates a fifth embodiment for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure

[0054] FIG. 16 is a flowchart of an example method for deriving a prediction for a chroma block using CCIP intra-prediction mode or MDLM intra-prediction mode.

[0055] FIG. 17 is a schematic diagram of an example video coding device.

DETAILED DESCRIPTION

[0056] It should be understood at the outset that although an illustrative implementation of one or more embodiments are provided below, the disclosed systems and/or methods may be implemented using any number of techniques, whether currently known or in existence. The disclosure should in no way be limited to the illustrative implementations, drawings, and techniques illustrated below, including the exemplary designs and implementations illustrated and described herein, but may be modified within the scope of the appended claims along with their full scope of equivalents.

[0057] Video coding includes partitioning video frames into blocks and encoding the blocks via intra-prediction and inter-prediction to compress the size of a video file. Specifically, the video frames can be partitioned into coding units (CUs) that contain luma coding blocks (CBs) and chroma CBs (light values and color values, respectively). Prediction (intra or inter) can then be applied to determine reference blocks for the coding blocks. Prediction results in prediction information that codes the CBs by reference to the reference block(s). Samples (e.g., pixels) of the CBs are compared to samples of the reference block(s) and differences in values are maintained as residual samples. Intra-prediction selects reference blocks in the same frame as the current block, while inter-prediction selects reference blocks in different frames from the current block. In most cases, intra-prediction modes employ reference luma blocks to predict luma blocks and reference chroma blocks to predict chroma blocks. Cross-component modes may also be used. In cross-component intra-prediction, luma reference samples are employed to predict chroma samples of a current block.

[0058] The present disclosure relates to improvements in the process of performing cross-component intra-prediction. Cross-component intra-prediction may include CCLM mode intra-prediction, MDLM mode intra-prediction, and/or MMLM mode intra-prediction. To perform such intra-prediction, luma samples of the same CU and luma samples and chroma samples from neighboring CBs are used as reference samples to predict chroma samples of a current block. Specifically, disclosed herein are improved mechanisms for deriving a cross-component intra-prediction for the current chroma block by guaranteeing that the number of reference samples used to calculate linear model coefficients of a linear model is a value of power of 2. This increases the efficiency of the component intra-prediction because a decoder can perform a shift operation, which has a lower decoder side cost, as compared to performing a divide operation.

[0059] FIG. 1 is a flowchart of an example operating method 100 of coding a video signal. Specifically, a video signal is encoded at an encoder. The encoding process compresses the video signal by employing various mechanisms to reduce the video file size. A smaller file size allows the compressed video file to be transmitted toward a user, while reducing associated bandwidth overhead. The decoder then decodes the compressed video file to reconstruct the original video signal for display to an end user. The decoding process generally mirrors the encoding process to allow the decoder to consistently reconstruct the video signal.

[0060] At step 101, the video signal is input into the encoder. For example, the video signal may be an uncompressed video file stored in memory. As another example, the video file may be captured by a video capture device, such as a video camera, and encoded to support live streaming of the video. The video file may include both an audio component and a video component. The video component contains a series of image frames that, when viewed in a sequence, gives the visual impression of motion. The frames contain pixels that are expressed in terms of light, referred to herein as luma components (or luma samples), and color, which is referred to as chroma components (or color samples). In some examples, the frames may also contain depth values to support three dimensional viewing.

[0061] At step 103, the video is partitioned into blocks. Partitioning includes subdividing the pixels in each frame into square and/or rectangular blocks for compression. For example, in High Efficiency Video Coding (HEVC) (also known as H.265 and MPEG-H Part 2) the frame can first be divided into coding tree units (CTUs), which are blocks of a predefined size (e.g., sixty four pixels by sixty four pixels). The CTUs contain both luma and chroma samples. Coding trees may be employed to divide the CTUs into blocks and then recursively subdivide the blocks until configurations are achieved that support further encoding. For example, luma components of a frame may be subdivided until the individual blocks contain relatively homogenous lighting values. Further, chroma components of a frame may be subdivided until the individual blocks contain relatively homogenous color values. Accordingly, partitioning mechanisms vary depending on the content of the video frames.

[0062] At step 105, various compression mechanisms are employed to compress the image blocks partitioned at step 103. For example, inter-prediction and/or intra-prediction may be employed. Inter-prediction is designed to take advantage of the fact that objects in a common scene tend to appear in successive frames. Accordingly, a block depicting an object in a reference frame need not be repeatedly described in adjacent frames. Specifically, an object, such as a table, may remain in a constant position over multiple frames. Hence the table is described once and adjacent frames can refer back to the reference frame. Pattern matching mechanisms may be employed to match objects over multiple frames. Further, moving objects may be represented across multiple frames, for example due to object movement or camera movement. As a particular example, a video may show an automobile that moves across the screen over multiple frames. Motion vectors can be employed to describe such movement. A motion vector is a two-dimensional vector that provides an offset from the coordinates of an object in a frame to the coordinates of the object in a reference frame. As such, inter-prediction can encode an image block in a current frame as a set of motion vectors indicating an offset from a corresponding block in a reference frame.

[0063] Intra-prediction encodes blocks in a common frame. Intra-prediction takes advantage of the fact that luma and chroma components tend to cluster in a frame. For example, a patch of green in a portion of a tree tends to be positioned adjacent to similar patches of green. Intra-prediction employs multiple directional prediction modes (e.g., thirty three in HEVC), a planar mode, and a direct current (DC) mode. The directional modes indicate that a current block is similar/the same as samples of a neighbor block in a corresponding direction. Planar mode indicates that a series of blocks along a row/column (e.g., a plane) can be interpolated based on neighbor blocks at the edges of the row. Planar mode, in effect, indicates a smooth transition of light/color across a row/column by employing a relatively constant slope in changing values. DC mode is employed for boundary smoothing and indicates that a block is similar/the same as an average value associated with samples of all the neighbor blocks associated with the angular directions of the directional prediction modes. Accordingly, intra-prediction blocks can represent image blocks as various relational prediction mode values instead of the actual values. Further, inter-prediction blocks can represent image blocks as motion vector values instead of the actual values. In either case, the prediction blocks may not exactly represent the image blocks in some cases. Any differences are stored in residual blocks. Transforms may be applied to the residual blocks to further compress the file.

[0064] At step 107, various filtering techniques may be applied. In HEVC, the filters are applied according to an in-loop filtering scheme. The block based prediction discussed above may result in the creation of blocky images at the decoder. Further, the block based prediction scheme may encode a block and then reconstruct the encoded block for later use as a reference block. The in-loop filtering scheme iteratively applies noise suppression filters, de-blocking filters, adaptive loop filters, and sample adaptive offset (SAO) filters to the blocks/frames. These filters mitigate such blocking artifacts so that the encoded file can be accurately reconstructed. Further, these filters mitigate artifacts in the reconstructed reference blocks so that artifacts are less likely to create additional artifacts in subsequent blocks that are encoded based on the reconstructed reference blocks.

[0065] Once the video signal has been partitioned, compressed, and filtered, the resulting data is encoded in a bitstream at step 109. The bitstream includes the data discussed above as well as any signaling data desired to support proper video signal reconstruction at the decoder. For example, such data may include partition data, prediction data, residual blocks, and various flags providing coding instructions to the decoder. The bitstream may be stored in memory for transmission toward a decoder upon request. The bitstream may also be broadcast and/or multicast toward a plurality of decoders. The creation of the bitstream is an iterative process. Accordingly, steps 101, 103, 105, 107, and 109 may occur continuously and/or simultaneously over many frames and blocks. The order shown in FIG. 1 is presented for clarity and ease of discussion, and is not intended to limit the video coding process to a particular order.

[0066] The decoder receives the bitstream and begins the decoding process at step 111. Specifically, the decoder employs an entropy decoding scheme to convert the bitstream into corresponding syntax and video data. The decoder employs the syntax data from the bitstream to determine the partitions for the frames at step 111. The partitioning should match the results of block partitioning at step 103. Entropy encoding/decoding as employed in step 111 is now described. The encoder makes many choices during the compression process, such as selecting block partitioning schemes from several possible choices based on the spatial positioning of values in the input image(s). Signaling the exact choices may employ a large number of bins. As used herein, a bin is a binary value that is treated as a variable (e.g., a bit value that may vary depending on context). Entropy coding allows the encoder to discard any options that are clearly not viable for a particular case, leaving a set of allowable options. Each allowable option is then assigned a code word. The length of the code word is based on the number of allowable options (e.g., one bin for two options, two bins for three to four options, etc.) The encoder then encodes the code word for the selected option. This scheme reduces the size of the code words as the code words are as big as desired to uniquely indicate a selection from a small sub-set of allowable options as opposed to uniquely indicating the selection from a potentially large set of all possible options. The decoder then decodes the selection by determining the set of allowable options in a similar manner to the encoder. By determining the set of allowable options, the decoder can read the code word and determine the selection made by the encoder.

[0067] At step 113, the decoder performs block decoding. Specifically, the decoder employs reverse transforms to generate residual blocks. Then the decoder employs the residual blocks and corresponding prediction blocks to reconstruct the image blocks according to the partitioning. The prediction blocks may include both intra-prediction blocks and inter-prediction blocks as generated at the encoder at step 105. The reconstructed image blocks are then positioned into frames of a reconstructed video signal according to the partitioning data determined at step 111. Syntax for step 113 may also be signaled in the bitstream via entropy coding as discussed above.

[0068] At step 115, filtering is performed on the frames of the reconstructed video signal in a manner similar to step 107 at the encoder. For example, noise suppression filters, de-blocking filters, adaptive loop filters, and SAO filters may be applied to the frames to remove blocking artifacts. Once the frames are filtered, the video signal can be output to a display at step 117 for viewing by an end user.

[0069] FIG. 2 is a schematic diagram of an example coding and decoding (codec) system 200 for video coding. Specifically, codec system 200 provides functionality to support the implementation of operating method 100. Codec system 200 is generalized to depict components employed in both an encoder and a decoder. Codec system 200 receives and partitions a video signal as discussed with respect to steps 101 and 103 in operating method 100, which results in a partitioned video signal 201. Codec system 200 then compresses the partitioned video signal 201 into a coded bitstream when acting as an encoder as discussed with respect to steps 105, 107, and 109 in method 100. When acting as a decoder codec system 200 generates an output video signal from the bitstream as discussed with respect to steps 111, 113, 115, and 117 in operating method 100. The codec system 200 includes a general coder control component 211, a transform scaling and quantization component 213, an intra-picture estimation component 215, an intra-picture prediction component 217, a motion compensation component 219, a motion estimation component 221, a scaling and inverse transform component 229, a filter control analysis component 227, an in-loop filters component 225, a decoded picture buffer component 223, and a header formatting and context adaptive binary arithmetic coding (CABAC) component 231. Such components are coupled as shown. In FIG. 2, black lines indicate movement of data to be encoded/decoded while dashed lines indicate movement of control data that controls the operation of other components. The components of codec system 200 may all be present in the encoder. The decoder may include a subset of the components of codec system 200. For example, the decoder may include the intra-picture prediction component 217, the motion compensation component 219, the scaling and inverse transform component 229, the in-loop filters component 225, and the decoded picture buffer component 223. These components are now described.

[0070] The partitioned video signal 201 is a captured video sequence that has been partitioned into blocks of pixels by a coding tree. A coding tree employs various split modes to subdivide a block of pixels into smaller blocks of pixels. These blocks can then be further subdivided into smaller blocks. The blocks may be referred to as nodes on the coding tree. Larger parent nodes are split into smaller child nodes. The number of times a node is subdivided is referred to as the depth of the node/coding tree. The divided blocks can be included in coding units (CUs) in some cases. For example, a CU can be a sub-portion of a CTU that contains a luma block, red difference chroma (Cr) block(s), and a blue difference chroma (Cb) block(s) along with corresponding syntax instructions for the CU. The split modes may include a binary tree (BT), triple tree (TT), and a quad tree (QT) employed to partition a node into two, three, or four child nodes, respectively, of varying shapes depending on the split modes employed. The partitioned video signal 201 is forwarded to the general coder control component 211, the transform scaling and quantization component 213, the intra-picture estimation component 215, the filter control analysis component 227, and the motion estimation component 221 for compression.

[0071] The general coder control component 211 is configured to make decisions related to coding of the images of the video sequence into the bitstream according to application constraints. For example, the general coder control component 211 manages optimization of bitrate/bitstream size versus reconstruction quality. Such decisions may be made based on storage space/bandwidth availability and image resolution requests. The general coder control component 211 also manages buffer utilization in light of transmission speed to mitigate buffer underrun and overrun issues. To manage these issues, the general coder control component 211 manages partitioning, prediction, and filtering by the other components. For example, the general coder control component 211 may dynamically increase compression complexity to increase resolution and increase bandwidth usage or decrease compression complexity to decrease resolution and bandwidth usage. Hence, the general coder control component 211 controls the other components of codec system 200 to balance video signal reconstruction quality with bit rate concerns. The general coder control component 211 creates control data, which controls the operation of the other components. The control data is also forwarded to the header formatting and CABAC component 231 to be encoded in the bitstream to signal parameters for decoding at the decoder.

[0072] The partitioned video signal 201 is also sent to the motion estimation component 221 and the motion compensation component 219 for inter-prediction. A frame or slice of the partitioned video signal 201 may be divided into multiple video blocks. Motion estimation component 221 and the motion compensation component 219 perform inter-predictive coding of the received video block relative to one or more blocks in one or more reference frames to provide temporal prediction. Codec system 200 may perform multiple coding passes, e.g., to select an appropriate coding mode for each block of video data.

[0073] Motion estimation component 221 and motion compensation component 219 may be highly integrated, but are illustrated separately for conceptual purposes. Motion estimation, performed by motion estimation component 221, is the process of generating motion vectors, which estimate motion for video blocks. A motion vector, for example, may indicate the displacement of a coded object relative to a predictive block. A predictive block is a block that is found to closely match the block to be coded, in terms of pixel difference. A predictive block may also be referred to as a reference block. Such pixel difference may be determined by sum of absolute difference (SAD), sum of square difference (SSD), or other difference metrics. HEVC employs several coded objects including a CTU, coding tree blocks (CTBs), and CUs. For example, a CTU can be divided into CTBs, which can then be divided into CBs for inclusion in CUs. A CU can be encoded as a prediction unit (PU) containing prediction data and/or one or more transform unit (TUs) containing transformed residual data for the CU. The motion estimation component 221 generates motion vectors, PUs, and TUs by using a rate-distortion analysis as part of a rate distortion optimization process. For example, the motion estimation component 221 may determine multiple reference blocks, multiple motion vectors, etc. for a current block/frame, and may select the reference blocks, motion vectors, etc. having the best rate-distortion characteristics. The best rate-distortion characteristics balance both quality of video reconstruction (e.g., amount of data loss by compression) with coding efficiency (e.g., size of the final encoding).

[0074] In some examples, codec system 200 may calculate values for sub-integer pixel positions of reference pictures stored in decoded picture buffer component 223. For example, video codec system 200 may interpolate values of one-quarter pixel positions, one-eighth pixel positions, or other fractional pixel positions of the reference picture. Therefore, motion estimation component 221 may perform a motion search relative to the full pixel positions and fractional pixel positions and output a motion vector with fractional pixel precision. The motion estimation component 221 calculates a motion vector for a PU of a video block in an inter-coded slice by comparing the position of the PU to the position of a predictive block of a reference picture. Motion estimation component 221 outputs the calculated motion vector as motion data to header formatting and CABAC component 231 for encoding and motion to the motion compensation component 219.

[0075] Motion compensation, performed by motion compensation component 219, may involve fetching or generating the predictive block based on the motion vector determined by motion estimation component 221. Again, motion estimation component 221 and motion compensation component 219 may be functionally integrated, in some examples. Upon receiving the motion vector for the PU of the current video block, motion compensation component 219 may locate the predictive block to which the motion vector points. A residual video block is then formed by subtracting pixel values of the predictive block from the pixel values of the current video block being coded, forming pixel difference values. In general, motion estimation component 221 performs motion estimation relative to luma components, and motion compensation component 219 uses motion vectors calculated based on the luma components for both chroma components and luma components. The predictive block and residual block are forwarded to transform scaling and quantization component 213.

[0076] The partitioned video signal 201 is also sent to intra-picture estimation component 215 and intra-picture prediction component 217. As with motion estimation component 221 and motion compensation component 219, intra-picture estimation component 215 and intra-picture prediction component 217 may be highly integrated, but are illustrated separately for conceptual purposes. The intra-picture estimation component 215 and intra-picture prediction component 217 intra-predict a current block relative to blocks in a current frame, as an alternative to the inter-prediction performed by motion estimation component 221 and motion compensation component 219 between frames, as described above. In particular, the intra-picture estimation component 215 determines an intra-prediction mode to use to encode a current block. In some examples, intra-picture estimation component 215 selects an appropriate intra-prediction mode to encode a current block from multiple tested intra-prediction modes. The selected intra-prediction modes are then forwarded to the header formatting and CABAC component 231 for encoding.

[0077] For example, the intra-picture estimation component 215 calculates rate-distortion values using a rate-distortion analysis for the various tested intra-prediction modes, and selects the intra-prediction mode having the best rate-distortion characteristics among the tested modes. Rate-distortion analysis generally determines an amount of distortion (or error) between an encoded block and an original unencoded block that was encoded to produce the encoded block, as well as a bitrate (e.g., a number of bits) used to produce the encoded block. The intra-picture estimation component 215 calculates ratios from the distortions and rates for the various encoded blocks to determine which intra-prediction mode exhibits the best rate-distortion value for the block. In addition, intra-picture estimation component 215 may be configured to code depth blocks of a depth map using a depth modeling mode (DMM) based on rate-distortion optimization (RDO).

[0078] The intra-picture prediction component 217 may generate a residual block from the predictive block based on the selected intra-prediction modes determined by intra-picture estimation component 215 when implemented on an encoder or read the residual block from the bitstream when implemented on a decoder. The residual block includes the difference in values between the predictive block and the original block, represented as a matrix. The residual block is then forwarded to the transform scaling and quantization component 213. The intra-picture estimation component 215 and the intra-picture prediction component 217 may operate on both luma and chroma components.

[0079] The transform scaling and quantization component 213 is configured to further compress the residual block. The transform scaling and quantization component 213 applies a transform, such as a discrete cosine transform (DCT), a discrete sine transform (DST), or a conceptually similar transform, to the residual block, producing a video block comprising residual transform coefficient values. Wavelet transforms, integer transforms, sub-band transforms or other types of transforms could also be used. The transform may convert the residual information from a pixel value domain to a transform domain, such as a frequency domain. The transform scaling and quantization component 213 is also configured to scale the transformed residual information, for example based on frequency. Such scaling involves applying a scale factor to the residual information so that different frequency information is quantized at different granularities, which may affect final visual quality of the reconstructed video. The transform scaling and quantization component 213 is also configured to quantize the transform coefficients to further reduce bit rate. The quantization process may reduce the bit depth associated with some or all of the coefficients. The degree of quantization may be modified by adjusting a quantization parameter. In some examples, the transform scaling and quantization component 213 may then perform a scan of the matrix including the quantized transform coefficients. The quantized transform coefficients are forwarded to the header formatting and CABAC component 231 to be encoded in the bitstream.

[0080] The scaling and inverse transform component 229 applies a reverse operation of the transform scaling and quantization component 213 to support motion estimation. The scaling and inverse transform component 229 applies inverse scaling, transformation, and/or quantization to reconstruct the residual block in the pixel domain, e.g., for later use as a reference block which may become a predictive block for another current block. The motion estimation component 221 and/or motion compensation component 219 may calculate a reference block by adding the residual block back to a corresponding predictive block for use in motion estimation of a later block/frame. Filters are applied to the reconstructed reference blocks to mitigate artifacts created during scaling, quantization, and transform. Such artifacts could otherwise cause inaccurate prediction (and create additional artifacts) when subsequent blocks are predicted.

[0081] The filter control analysis component 227 and the in-loop filters component 225 apply the filters to the residual blocks and/or to reconstructed image blocks. For example, the transformed residual block from the scaling and inverse transform component 229 may be combined with a corresponding prediction block from intra-picture prediction component 217 and/or motion compensation component 219 to reconstruct the original image block. The filters may then be applied to the reconstructed image block. In some examples, the filters may instead be applied to the residual blocks. As with other components in FIG. 2, the filter control analysis component 227 and the in-loop filters component 225 are highly integrated and may be implemented together, but are depicted separately for conceptual purposes. Filters applied to the reconstructed reference blocks are applied to particular spatial regions and include multiple parameters to adjust how such filters are applied. The filter control analysis component 227 analyzes the reconstructed reference blocks to determine where such filters should be applied and sets corresponding parameters. Such data is forwarded to the header formatting and CABAC component 231 as filter control data for encoding. The in-loop filters component 225 applies such filters based on the filter control data. The filters may include a deblocking filter, a noise suppression filter, a SAO filter, and an adaptive loop filter. Such filters may be applied in the spatial/pixel domain (e.g., on a reconstructed pixel block) or in the frequency domain, depending on the example.

[0082] When operating as an encoder, the filtered reconstructed image block, residual block, and/or prediction block are stored in the decoded picture buffer component 223 for later use in motion estimation as discussed above. When operating as a decoder, the decoded picture buffer component 223 stores and forwards the reconstructed and filtered blocks toward a display as part of an output video signal. The decoded picture buffer component 223 may be any memory device capable of storing prediction blocks, residual blocks, and/or reconstructed image blocks.

[0083] The header formatting and CABAC component 231 receives the data from the various components of codec system 200 and encodes such data into a coded bitstream for transmission toward a decoder. Specifically, the header formatting and CABAC component 231 generates various headers to encode control data, such as general control data and filter control data. Further, prediction data, including intra-prediction and motion data, as well as residual data in the form of quantized transform coefficient data are all encoded in the bitstream. The final bitstream includes all information desired by the decoder to reconstruct the original partitioned video signal 201. Such information may also include intra-prediction mode index tables (also referred to as code word mapping tables), definitions of encoding contexts for various blocks, indications of most probable intra-prediction modes, an indication of partition information, etc. Such data may be encoded be employing entropy coding. For example, the information may be encoded by employing context adaptive variable length coding (CAVLC), CABAC, syntax-based context-adaptive binary arithmetic coding (SBAC), probability interval partitioning entropy (PIPE) coding, or another entropy coding technique. Following the entropy coding, the coded bitstream may be transmitted to another device (e.g., a video decoder) or archived for later transmission or retrieval.

[0084] FIG. 3 is a block diagram illustrating an example video encoder 300 that may perform cross-component intra-prediction. Video encoder 300 may be employed to implement the encoding functions of codec system 200 and/or implement steps 101, 103, 105, 107, and/or 109 of operating method 100. Encoder 300 partitions an input video signal, resulting in a partitioned video signal 301, which is substantially similar to the partitioned video signal 201. The partitioned video signal 301 is then compressed and encoded into a bitstream by components of encoder 300.

[0085] Specifically, the partitioned video signal 301 is forwarded to an intra-picture prediction component 317 for intra-prediction. The intra-picture prediction component 317 may be substantially similar to intra-picture estimation component 215 and intra-picture prediction component 217. The partitioned video signal 301 is also forwarded to a motion compensation component 321 for inter-prediction based on reference blocks in a decoded picture buffer component 323. The motion compensation component 321 may be substantially similar to motion estimation component 221 and motion compensation component 219. The prediction blocks and residual blocks from the intra-picture prediction component 317 and the motion compensation component 321 are forwarded to a transform and quantization component 313 for transformation and quantization of the residual blocks. The transform and quantization component 313 may be substantially similar to the transform scaling and quantization component 213. The transformed and quantized residual blocks and the corresponding prediction blocks (along with associated control data) are forwarded to an entropy coding component 331 for coding into a bitstream. The entropy coding component 331 may be substantially similar to the header formatting and CABAC component 231.

[0086] The transformed and quantized residual blocks and/or the corresponding prediction blocks are also forwarded from the transform and quantization component 313 to an inverse transform and quantization component 329 for reconstruction into reference blocks for use by the motion compensation component 321. The inverse transform and quantization component 329 may be substantially similar to the scaling and inverse transform component 229. In-loop filters in an in-loop filters component 325 are also applied to the residual blocks and/or reconstructed reference blocks, depending on the example. The in-loop filters component 325 may be substantially similar to the filter control analysis component 227 and the in-loop filters component 225. The in-loop filters component 325 may include multiple filters as discussed with respect to in-loop filters component 225. The filtered blocks are then stored in a decoded picture buffer component 323 for use as reference blocks by the motion compensation component 321. The decoded picture buffer component 323 may be substantially similar to the decoded picture buffer component 223.

[0087] For example, the intra-picture prediction component 317 can be configured to perform cross-component intra-prediction. In cross-component intra-prediction, the chroma components for a current block are predicted based in part on the luma components of neighboring blocks. In an encoder 300, the neighboring blocks may be encoded and then later reconstructed to act as reference blocks for further blocks. Hence, the chroma components of the current block are predicted based on reconstructed neighboring luma samples from reconstructed neighboring blocks. Multiple cross-component mechanisms are discussed in greater detail below. Regardless of the cross-component mechanisms used, the neighboring luma components are downsampled. This is because luma blocks are generally at four times the resolution of chroma blocks, and hence contain four times the number of samples of the chroma blocks. Downsampling is the process of reducing the resolution of a group of pixel samples. Downsampling allows the number of reference luma samples to match the number of chroma samples to provide for an accurate comparison when performing cross-component intra-prediction.

[0088] FIG. 4 is a block diagram illustrating an example video decoder 400 that may perform cross-component intra-prediction. Video decoder 400 may be employed to implement the decoding functions of codec system 200 and/or implement steps 111, 113, 115, and/or 117 of operating method 100. Decoder 400 receives a bitstream, for example from an encoder 300, and generates a reconstructed output video signal based on the bitstream for display to an end user.

[0089] The bitstream is received by an entropy decoding component 433. The entropy decoding component 433 is configured to implement an entropy decoding scheme, such as CAVLC, CABAC, SBAC, PIPE coding, or other entropy coding techniques. For example, the entropy decoding component 433 may employ header information to provide a context to interpret additional data encoded as code words in the bitstream. The decoded information includes any desired information to decode the video signal, such as general control data, filter control data, partition information, motion data, prediction data, and quantized transform coefficients from residual blocks. The quantized transform coefficients are forwarded to an inverse transform and quantization component 429 for reconstruction into residual blocks. The inverse transform and quantization component 429 may be similar to inverse transform and quantization component 329.

[0090] The reconstructed residual blocks and/or prediction blocks are forwarded to intra-picture prediction component 417 for reconstruction into image blocks based on intra-prediction operations. The intra-picture prediction component 417 may be similar to intra-picture estimation component 215 and an intra-picture prediction component 217. Specifically, the intra-picture prediction component 417 employs prediction modes to locate a reference block in the frame and applies a residual block to the result to reconstruct intra-predicted image blocks. The reconstructed intra-predicted image blocks and/or the residual blocks and corresponding inter-prediction data are forwarded to a decoded picture buffer component 423 via an in-loop filters component 425, which may be substantially similar to decoded picture buffer component 223 and in-loop filters component 225, respectively. The in-loop filters component 425 filters the reconstructed image blocks, residual blocks, and/or prediction blocks, and such information is stored in the decoded picture buffer component 423. Reconstructed image blocks from decoded picture buffer component 423 are forwarded to a motion compensation component 421 for inter-prediction. The motion compensation component 421 may be substantially similar to motion estimation component 221 and/or motion compensation component 219. Specifically, the motion compensation component 421 employs motion vectors from a reference block to generate a prediction block and applies a residual block to the result to reconstruct an image block. The resulting reconstructed blocks may also be forwarded via the in-loop filters component 425 to the decoded picture buffer component 423. The decoded picture buffer component 423 continues to store additional reconstructed image blocks, which can be reconstructed into frames via the partition information. Such frames may also be placed in a sequence. The sequence is output toward a display as a reconstructed output video signal.

[0091] As with encoder 300, the intra-picture prediction component 417 of the decoder 400 can be configured to perform cross-component intra-prediction. In cross-component intra-prediction, the chroma components for a current block are predicted based on the luma components of neighboring blocks. In a decoder 400, the neighboring blocks may be reconstructed and then used as reference blocks for further blocks (e.g., a current block). Hence, the chroma components of the current block are in part predicted based on reconstructed neighboring luma samples from reconstructed neighboring blocks.

[0092] FIG. 5 is a schematic diagram illustrating an example of intra-prediction modes 500 as employed in video coding. For example, intra-prediction modes 500 may be employed by steps 105 and 113 of method 100, intra-picture estimation component 215 and an intra-picture prediction component 217 of codec system 200, intra-picture prediction component 317 of encoder 300, and/or intra-picture prediction component 417 of decoder 400.

[0093] Intra-prediction involves matching one or more samples of a current block in a CU to one or more reference samples of one or more neighboring blocks. The current block can then be represented as a selected prediction mode index and a residual block, which is much smaller than representing all of the sample values contained in the current block. Intra-prediction can be used when there is no available reference frame, or when inter-predication coding is not used for the current block, slice, and/or frame. The reference samples for intra-prediction are generally derived from neighboring blocks in the same frame. Advanced Video Coding (AVC), also known as H.264, and H.265/HEVC both employ a reference line of boundary samples of adjacent blocks as reference samples for intra-prediction. In order to cover different textures or structural characteristics many different intra-prediction modes 500 are employed. In each mode, a different prediction signal derivation method is used. For example, H.265/HEVC supports a total of thirty five single component intra-prediction modes 500 that spatially correlate a current block of samples to one or more reference samples of the same type (e.g., luma samples predicted by reference luma samples and chroma samples predicted by reference chroma samples). Specifically, as shown in FIG. 5, intra-prediction modes 500 include thirty-three directional prediction modes indexed as modes two through thirty four, a DC mode indexed as mode one, and a planar mode indexed as mode zero.

[0094] During encoding, the encoder matches the luma/chroma values of a current block in a CU with the luma/chroma values of corresponding reference samples of a reference line across the edges of neighboring blocks. When the best match is found with one of the reference lines, the encoder selects one of the directional intra-prediction modes 500 that points to the best matching reference line (e.g., smallest difference in sample value(s)). For clarity of discussion, acronyms are employed below to reference particular directional intra-prediction modes 500. DirS denotes the starting directional intra-prediction mode when counting clockwise from the bottom left (e.g., mode two in HEVC). DirE denotes the ending directional intra-prediction mode when counting clockwise from the bottom left (e.g., mode thirty four in HEVC). DirD denotes the middle directional intra-coding mode when counting clockwise from the bottom left (e.g., mode eighteen in HEVC). DirH denotes a horizontal intra-prediction mode (e.g., mode ten in HEVC). DirV denotes a vertical intra-prediction mode (e.g., mode twenty six in HEVC).

[0095] DC mode acts as a smoothing function and derives a prediction value of a block in the CU as an average value of all the reference samples of the reference line traversing the neighboring blocks. Planar mode returns a prediction value that indicates a smooth transition (e.g., constant slope of values) between samples at the bottom and top left or top left and top right of the reference line of reference samples.

[0096] For planar, DC, and prediction modes from DirH to DirV, the samples of both the top row of the reference line and the left column of the reference line are used as reference samples. For prediction modes with prediction directions from DirS to DirH (including DirS and DirH), the reference samples of the neighboring blocks on the left column of the reference line are used as reference samples. For prediction modes with prediction directions from DirV to DirE (including DirV and DirE), the reference samples of the neighboring blocks on the top row of the reference line are used as reference samples. Accordingly, intra-prediction modes 500 can be used to indicate a spatial and/or directional relationship between samples and one or more reference samples.

[0097] FIG. 6 is a schematic diagram illustrating an example mechanism of performing cross-component intra-prediction 600. Cross-component intra-prediction 600 may be performed by an intra-picture estimation component 215 and/or an intra-picture prediction component 217 of a codec system 200, an intra-picture prediction component 317 of an encoder 300, and/or an intra-picture prediction component 417 of a decoder 400. Specifically, cross-component intra-prediction 600 can be employed during block compression at step 105 of method 100 and during block decoding at step 113 of method 100 at an encoder and a decoder, respectively. Cross-component intra-prediction 600 operates in a manner that is substantially similar to intra-prediction modes 500. However, cross-component intra-prediction 600 uses luma reference samples to predict chroma samples in order to reduce cross-component redundancy.

[0098] Cross-component intra-prediction 600 operates on a current block 601 based on neighboring blocks 607 as well as other blocks in the same CU. Specifically, a CU contains a luma block and two chroma blocks. In this context, the current block 601 is a chroma block that contains chroma samples. The chroma samples may include red difference chroma (Cr) samples, blue difference chroma (Cb) samples, or combinations thereof. The neighboring blocks 607 contain luma samples and chroma samples. A neighboring block 607 is a block of samples that is directly adjacent to the current block 601. Cross-component intra-prediction 600 operates in part based on neighboring reference samples 603 in the neighboring blocks 607. Specifically, chroma samples of the current block 601 can be predicted by a combination of luma samples of the same CU as the current block as well as set(s) of luma samples and chroma samples acting as neighboring reference samples 603 in the neighboring blocks 607. Neighboring reference samples 603 are taken from rows of the neighboring blocks 607 that are directly adjacent to the current block 601. Neighboring reference samples 603 are also taken from columns of the neighboring blocks 607 that are directly adjacent to the current block 601.

[0099] Several mechanisms may be employed to predict chroma samples of the current block 601 based on the neighboring luma samples and neighboring chroma samples of the neighboring blocks 607 as discussed in greater detail below. In general, a linear model is created to correlate the chroma samples of the current block 601 to the luma samples of the same CU and neighboring luma and neighboring chroma samples acting as reference samples 603. The model parameters may be calculated by minimizing regression error between the luma and chroma samples acting as neighboring reference samples 603 of the neighboring block(s) 607. Another approach involves calculating model parameters based on the minimum and maximum luma values from the neighboring reference samples 603. Such computation mechanisms are discussed in detail below.

[00100] FIG. 7 is a schematic diagram illustrating an example mechanism of performing CCLM intra-prediction 700. CCLM intra-prediction 700 is a type of cross-component intra-prediction 600. Hence, CCLM intra-prediction 700 may be performed by an intra-picture estimation component 215 and/or an intra-picture prediction component 217 of a codec system 200, an intra-picture prediction component 317 of an encoder 300, and/or an intra-picture prediction component 417 of a decoder 400. Specifically, CCLM intra-prediction 700 can be employed during block compression at step 105 of method 100 and during block decoding at step 113 of method 100 at an encoder and a decoder, respectively.

[00101] CCLM intra-prediction 700 predicts chroma samples 703 in a chroma block 701. The chroma samples 703 appear at integer positions shown as intersecting lines. The prediction is based in part on neighboring reference samples, which are depicted as black circles. Unlike with intra-prediction modes 500, the chroma samples 703 are not predicted solely based on the neighboring chroma reference samples 705, which are denoted as reconstructed chroma samples (Rec'C). The chroma samples 703 are also predicted based on luma reference samples 713 and neighboring luma reference samples 715. Specifically, a CU contains a luma block 711 and two chroma blocks 701. A model is generated that correlates the chroma samples 703 and the luma reference samples 713 in the same CU. Linear coefficients for the model are determined by comparing the neighboring luma reference samples 715 to the neighboring chroma reference samples 705.

[00102] The luma reference samples 713 are derived from the luma block 711 in the same CU as the chroma block 701. The neighboring luma reference samples 715 are selected from luma samples of neighboring blocks adjacent to the luma block 711. For instance, the neighboring luma reference samples 715 in the top neighboring samples section 717 are selected from the corresponding positions in the neighboring block above the luma block 711, and the neighboring luma reference samples 715 in the left neighboring samples section 716 are selected from the corresponding positions in the neighboring block to the left of the luma block 711. As the luma reference samples 713 are reconstructed, the luma reference samples 713 are denoted as reconstructed luma samples (Rec'L). Similarly, the neighboring chroma reference samples 705 are selected from luma samples of neighboring blocks adjacent to the chroma block 701. For instance, the neighboring chroma reference samples 705 in the top neighboring samples section 707 are selected from the corresponding positions in the neighboring chroma block above the chroma block 701, and the neighboring chroma reference samples 705 in the left neighboring samples section 706 are selected from the corresponding positions in the neighboring chroma block to the left of the chroma block 701. As the neighboring chroma reference samples 705 are reconstructed, the neighboring chroma reference samples 705 are denoted as reconstructed chroma samples (Rec'C).

[00103] As shown, the luma block 711 contains four times the samples as the chroma block 701. Specifically, the chroma block 701 contains N number of samples by N number of samples while the luma block 711 contains 2N number of samples by 2N number of samples. Hence, the luma block 711 is four times the resolution of the chroma block 701. For the prediction to operate on the luma reference samples 713 and the neighboring luma reference samples 715, the luma reference samples 713 and the neighboring luma reference samples 715 are downsampled to provide an accurate comparison with the neighboring chroma reference samples 705 and the chroma samples 703. Downsampling is the process of reducing the resolution of a group of sample values. For example, when YUV4:2:0 format is used, the luma samples may be downsampled by a factor of four (e.g., width by two, and height by two). YUV is a color encoding system that employs a color space in terms of luma components Y and two chrominance components U and V.

[00104] Once the neighboring luma reference samples 715 and the luma reference samples 713 are downsampled, a model can be generated to predict the chroma samples 703 of the chroma block 701. Specifically, in CCLM intra-prediction 700, a prediction for chroma samples 703 of the chroma block 701 can be determined according to the linear model described by equation 1:

[00105] $$\mathrm{pred_C(i,j)} = \alpha \cdot \mathrm{rec_L{}'(i,j)} + \beta \tag{1}$$

[00106] where $\mathrm{pred_C(i,j)}$ are the prediction chroma samples 703 of chroma block 701 at a height i and a width j, $\mathrm{rec_L{}'(i,j)}$ are the reconstructed downsampled luma reference samples 713 of the same CU, and α and β are linear coefficients determined by comparing the neighboring downsampled luma reference samples 715 and the chroma reference samples 705.

[00107] In one example, α and β are determined by minimizing the regression error between the downsampled neighboring luma reference samples 715 and the chroma reference samples 705. This can be done according to equations 2 and 3:

[00108] $$\alpha = \frac{\mathrm{N}\cdot\sum(\mathrm{L(n)}\cdot\mathrm{C(n)}) - \sum\mathrm{L(n)}\cdot\sum\mathrm{C(n)}}{\mathrm{N}\cdot\sum(\mathrm{L(n)}\cdot\mathrm{L(n)}) - \sum\mathrm{L(n)}\cdot\sum\mathrm{L(n)}} \tag{2}$$

[00109] $$\beta = \frac{\sum\mathrm{C(n)} - \alpha\cdot\sum\mathrm{L(n)}}{\mathrm{N}} \tag{3}$$

[00110] where $\mathrm{L(n)}$ represents the downsampled top and left neighboring reconstructed luma samples (e.g., downsampled neighboring luma reference samples 715), $\mathrm{C(n)}$ represents the top and left neighboring reconstructed chroma samples (e.g., the neighboring chroma reference samples 705), and value of N is equal to the number of samples of both the top template and the left template, as described below, that are used to derive the linear model coefficients of the current chroma coding block (e.g., chroma block 701).

[00111] FIG. 8 is a schematic diagram illustrating a top template and a left template in accordance with an embodiment. In this disclosure, the term template is used to denote the neighboring reconstructed chroma samples and the corresponding down-sampled neighboring reconstructed luma samples. For instance, as shown in FIG. 8, a top template means the top neighboring reconstructed chroma samples and the corresponding down-sampled top neighboring reconstructed luma samples. A left template means the left neighboring reconstructed chroma samples and the corresponding down-sampled left neighboring reconstructed luma samples. For example, in the depicted embodiment, luma' block 830 has a top template 810 and a left template 820. Luma' means the down-sampled luma part which has a same spatial resolution with chroma part. Chroma block 860 has a top template 840 and a left template 850. For the luma' block 830' and chroma block 860 in Fig. 8, because the width is larger than height, the top template 810 and top template 840 can also be described as the template at long side, and the left template 820 and the left template 850 can also be described as the template at short side. The top template, for example, may be a top neighboring sample line. In an example, the line of neighboring samples (such as a top neighboring sample line) located above of the block (such as luma' block or chroma block), and horizontal position (such as x of (x, y1)) of the left-most neighboring sample is equal to the horizontal position (such as x of (x, y2)) of the top-left predicted sample of the block (such as luma' block or chroma block). The left template, for example, may be a left neighboring sample column. In an example, the column of neighboring samples (such as a left neighboring sample column) located to the left of the block (such as luma' block or chroma block), and vertical position (such as y of (x1, y)) of the top-most neighboring sample is equal to the vertical position (such as y of (x2, y)) of the top-left predicted sample of the block (such as luma' block or chroma block).

[00112] FIGS. 9-10 are schematic diagrams illustrating an example mechanism of performing MDLM intra-prediction. MDLM intra-prediction operates in a manner similar to CCLM intra-prediction 700, but employs different templates. Specifically, MDLM intra-prediction uses both a cross-component linear model prediction (CCIP)_A mode 900 and a CCIP_L mode 1000 when determining linear model coefficients α and β. For example, MDLM intra-prediction may calculate linear model coefficients α and β using CCIP_A mode 900 and CCIP_L mode 1000. In CCIP_A mode 900, only the top templates are used to calculate the linear model coefficients. To get more samples, the top templates are extended to (W+H), usually. Similarly, In CCIP_L mode, only the left templates are used to calculate the linear model coefficients. To get more samples, the left templates are extended to (H+W), usually. MDLM intra-prediction may then select CCIP_A mode 900 or CCIP_L mode 1000 depending on the result that provides the greatest coding efficiency (e.g., the least residual samples). In another example, MDLM intra-prediction may use both CCIP_A mode 900 and CCIP_L mode 1000 to determine linear model coefficients α and β.

[00113] As such, MDLM intra-prediction using CCIP_A mode 900 and/or CCIP_L mode 1000 is a type of cross-component intra-prediction 600. Accordingly, MDLM intra-prediction using CCIP_A mode 900 and/or CCIP_L mode 1000 may be performed by an intra-picture estimation component 215 and/or an intra-picture prediction component 217 of a codec system 200, an intra-picture prediction component 317 of an encoder 300, and/or an intra-picture prediction component 417 of a decoder 400. Specifically, MDLM intra-prediction using CCIP_A mode 900 and/or CCIP_L mode 1000 can be employed during block compression at step 105 of method 100 and during block decoding at step 113 of method 100 at an encoder and a decoder, respectively.

[00114] CCIP_A mode 900 generates a model to predict chroma samples 903 in a current block 901 based on luma reference samples 913 in a luma block 911, where the luma block 911 is in the same CU as the current block 901, in a manner similar to CCLM intra-prediction 700. Specifically, a linear model is generated according to equation 1 above. Further, a template 917 and a corresponding template 907 are employed to select neighboring luma reference samples 915 and neighboring chroma reference samples 905 in a manner similar to CCLM intra-prediction 700. The difference is that the templates 907 and 917 are shaped differently. As with CCLM intra-prediction 700, CCIP_A mode 900 downsamples the neighboring luma reference samples 915 and the luma reference samples 913. CCIP_A mode 900 uses the luma reference samples 913 to determine Rec'L in equation 1. CCIP_A mode 900 then uses downsampled neighboring luma reference samples 915 and neighboring chroma reference samples 905 to determine linear model coefficients α and β. The linear model coefficients α and β can then be employed to complete the linear model according to equation 1 to obtain the prediction for the chroma samples 903.

[00115] The templates 907 and 917 are applied to neighboring chroma blocks and luma blocks, respectively, that are above and directly adjacent to the current block 901 and the corresponding luma block 911, respectively. The templates 907 and 917 do not obtain samples to the left of the current block 901 and the corresponding luma block 911. As templates 907 and 917 are only applied above the current block 901 and the luma block 911, the templates

907 and 917 are extended horizontally past the current block 901 and the corresponding luma block 911, respectively, to obtain more samples. This may allow templates 907 and 917 to obtain an equivalent number of neighboring chroma reference samples 905 and neighboring luma reference samples 915, respectively, to the numbers used by the CCLM intra-prediction 700 when determining linear model coefficients α and β.

[00116] CCIP_L mode 1000 generates a model to predict chroma samples 1003 in a current block 1001 based on luma reference samples 1013 in a luma block 1011, where the luma block 1011 is in the same CU as the current block 1001, in a manner similar to CCLM intra-prediction 700. Specifically, a linear model is generated according to equation 1 above. Further, a template 1016 and a corresponding template 1006 are employed to select neighboring luma reference samples 1015 and neighboring chroma reference samples 1005 in a manner similar to CCLM intra-prediction 700. The difference is that the templates 1006 and 1016 are shaped differently. As with CCLM intra-prediction 700, CCIP_L mode 1000 downsamples the neighboring luma reference samples 1015 and the luma reference samples 1013. CCIP_A mode 1000 uses the luma reference samples 1013 to determine Rec'L in equation 1. CCIP_A mode 1000 then uses downsampled neighboring luma reference samples 1015 and neighboring chroma reference samples 1005 to determine linear model coefficients α and β. The linear model coefficients α and β can then be employed to complete the linear model according to equation 1 to obtain the prediction for the chroma samples 1003.

[00117] The templates 1006 and 1016 are applied to neighboring chroma blocks and luma blocks, respectively, that are to the left of and directly adjacent to the current block 1001 and the corresponding luma block 1011, respectively. The templates 1006 and 1016 do not obtain samples above the current block 1001 and the corresponding luma block 1011. As templates 1006 and 1016 are only applied to the left of the current block 1001 and the luma block 1011, the templates 1006 and 1016 are extended vertically below the current block 1001 and the corresponding luma block 1011, respectively, to obtain more samples. This may allow templates 1006 and 1016 to obtain an equivalent number of neighboring chroma reference samples 1005 and neighboring luma reference samples 1015, respectively, to the numbers used by the CCLM intra-prediction 700 when determining linear model coefficients α and β.

[00118] CCIP mode and MDLM (consisting of CCIP_A mode and CCIP_L mode) can be used together, or, separately. For example, in some embodiments, only CCIP is used in a codec, or only MDLM is used in a codec, or both CCIP and MDLM are used in a codec. In the last case, the three modes (CCIP, CCIP_A, and CCIP_L) are added as three additional chroma intra prediction modes. At the encoder side, three more rate-distortion (RD) cost checks for the chroma components are added for selecting the chroma intra prediction mode. It should be noted that in existing coding block partition methods, the size of height or width is a usually a value of power of 2, even for a non-square block, but the sum of the height and the width of the block may not be a value of power of 2.

[00119] Despite the compression advantage that has been shown for the CCIP and MDLM modes, some areas can be optimized to achieve low decoder side cost, or can be improved to achieve higher coding gain. One such area is the selection of reference samples. Currently, the number of reference samples to calculate the linear model coefficients may not be a power of 2 for a non-square block. This causes inefficiencies in a decoder since a decoder prefers to perform a shift operation to a divide operation because a shift operation has a lower decoder side cost as compared with a divide operation. A shift operation takes two integers as operands and returns the result of shifting the bits of the left operand by the number of positions specified by the right operand. For example, the left shift operator shifts bits to the left, and the right shift operator shifts bits to the right. The idea is based on the fact that every number can be represented in binary form. In binary arithmetic, division by two can be performed by a bit shift operation that shifts the number one place to the right. For example, 1101001 in binary (the decimal number 105), shifted one place to the right, is 110100 (the decimal number 52): the lowest order bit is removed. Similarly, division by any power of two (2k) may be performed by right-shifting k positions. Because bit shifts are often much faster operations than division, replacing a division by a shift in this way can improve a decoder's performance. Thus, it would be beneficial to that the number of reference samples is a value of power of 2.

[00120] Accordingly, disclosed herein are several embodiments for ensuring that the number of reference samples used to derive the linear model coefficients is a value of power of 2. In particular, the disclosed embodiments apply to a chroma block having a width (W) and a height (H) that are not equal (i.e., a non-square block). For a square block, existing methods can be used. In the following example embodiments, the chroma block will have a height less than the width. However, it should be noted that the disclosed embodiments apply equally to a chroma block having a height greater than the width.

[00121] In the various embodiments disclosed herein, the number of reference samples of the top template and the left template (or the extended top template or extended left template) refers to some or all of the reference samples in the respective template. For examples, in some implementations, a step size (e.g., step size can be 1 or 2) can be used to reduce the number of samples of the respective template, but still ensure that the number of reference samples used to derive the linear model coefficients of a linear model is a power of 2. The reduction in the number of samples of the respective template may provide the added benefit of reducing the complexity of the preceding aspects.

[00122] FIG. 11 illustrates a first embodiment 1100 for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure. The first embodiment can be used for both CCIP intra-prediction mode and MDLM intra-prediction mode. Hence, the first embodiment may be performed by an intra-picture estimation component 215 and/or an intra-picture prediction component 217 of a codec system 200, an intra-picture prediction component 317 of an encoder 300, and/or an intra-picture prediction component 417 of a decoder 400. Specifically, MMLM intra-prediction can be employed during block compression at step 105 of method 100 and during block decoding at step 113 of method 100 at an encoder and a decoder, respectively.

[00123] In the first embodiment, the number of reference samples of both top template and the left template is equal to the long side size (LSS). For the short side, the template is extended to equal to the LSS. In the depicted embodiment, the LSS is W and the short side size (SSS) is H. As shown in FIG. 11, the top template (A) 1102 is on the LSS and the left template (B) 1140 is on the SSS. In this embodiment, the left template (B) 1140 will be extended from H to W, which means, the reference samples of the left template portion (C) 1106 will also be used. Thus, for CCIP mode, the top template (A) 1102, the left template (B) 1140, and the left template portion (C) 1106 will be used to calculate the linear model coefficients, making the number of reference samples equal to 2W.

[00124] As for MDLM mode, for CCIP_A mode, the top template (A) 1102 will be used to calculate the linear model coefficients, and the number of reference samples is W. For CCIP_L mode, the left template (B) 1140 and the left template portion (C) 1106 will be used to calculate the linear model coefficients, and the number of reference samples is W.

[00125] By using the proposed first embodiment, even for a non-square block, the number of reference samples to calculate the linear model coefficients is a value of power of 2, for both CCIP mode and MDLM mode (CCIP_A mode and CCIP_L mode). Another benefit of this method is that the template size of top boundary or left boundary is the same for CCIP, CCIP_A, and CCIP_L modes.

[00126] FIG. 12 illustrates a second embodiment 1200 for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure. The second embodiment can be used for both CCIP intra-prediction mode and MDLM intra-prediction mode. Hence, the second embodiment may be performed by an intra-picture estimation component 215 and/or an intra-picture prediction component 217 of a codec system 200, an intra-picture prediction component 317 of an encoder 300, and/or an intra-picture prediction component 417 of a decoder 400. Specifically, MMLM intra-prediction can be employed during block compression at step 105 of method 100 and during block decoding at step 113 of method 100 at an encoder and a decoder, respectively.

[00127] In the second embodiment, the number of reference samples that will be used to calculate linear model coefficients of a linear model for a current chroma block is equal to M. M is define as the smallest integer value that is not smaller than (W+H), and M is a value of power of 2. In this embodiment, the additional samples (i.e., the difference between M and (W+H)), will be resampled from the existing (W+H) samples using equal step size S, where S is an integer, and is calculated as S= (N)/(M-N), where N=(W+H).

[00128] For example, using the current chroma block illustrated in FIG. 12, the LSS is W, and the SSS is H. Therefore, for CCIP mode, the top template (A) 1202 and the left template (B) 1206 will be used to calculate the linear model coefficients. The number of existing reference samples is N, which is not power of 2 for a non-square block. In this embodiment, another (M-W-H) reference samples that will be re-sampled from the existing N samples. For examples, the array containing samples of the left template (B) 1206 and the top template (A) 1202 is:

[00129] $\Omega = (\omega_0, \omega_1, \omega_2, \omega_3, \omega_4, \ldots \omega_{N-1}) = (b_0, b_1, b_2, b_3, \ldots b_{H-1}, a_0, a_1, a_2, \ldots a_{W-1})$

[00130] The re-sampling samples are $(\omega_0, \omega_S, \omega_{2S}, \omega_{3S}, \ldots \omega_{(M-N-1)S})$.

[00131] In an embodiment, the order to read the samples of the left template (B) 1206 and the top template (A) 1202 is from the bottom of the left template (B) 1206 to top of the left template (B) 1206, and then from the left of the top template (A) 1202 to right of the top template (A) 1202.

[00132] As for MDLM mode, for CCIP_A mode, the top template (A) 1202 and a top template (D) 1204 will be used to calculate the linear model coefficients. The number of reference samples is W+H. Another (M-N) samples will be re-sampled from top template (A) 1202 and the top template (D) 1204. In an embodiment, the resampling method is similar to the method described above for CCIP mode. In an embodiment, the order to read the samples of the top template (A) 1202 and the top template (D) 1204 is from the left of the top template (A) 1202 to right of the top template (A) 1202, and then from the left of the top template (D) 1204 to right of the top template (D) 1204.

[00133] For CCIP_L mode, the left template (B) 1206 and the left template (C) 1208 will be used to calculate the linear model coefficients. The number of reference samples is H+W. An additional (M-N) samples will be re-sampled from the left template (B) 1206 and the left template (C) 1208. In an embodiment, the resampling method is similar to the method described above for CCIP mode. In an embodiment, the order to read the samples of the left template (B) 1206 and the left template (C) 1208 is from the top of the left template (B) 1206 to bottom of the left template (B) 1206, and then from the top of the left template (C) 1208 to bottom of the left template (C) 1208.

[00134] By using the proposed second embodiment, even for a non-square block, the number of reference samples to calculate the linear model coefficients is a value of power of 2, for both CCIP mode and MDLM mode.

[00135] FIG. 13 illustrates a third embodiment 1300 for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure. The third embodiment can only be used for MDLM intra-prediction mode. Hence, the third embodiment may be performed by an intra-picture estimation component 215 and/or an intra-picture prediction component 217 of a codec system 200, an intra-picture prediction component 317 of an encoder 300, and/or an intra-picture prediction component 417 of a decoder 400. Specifically, MMLM intra-prediction can be employed during block compression at step 105 of method 100 and during block decoding at step 113 of method 100 at an encoder and a decoder, respectively.

[00136] In the third embodiment, for MDLM, the number of reference samples that will be used to calculate linear model coefficients of a linear model for a current chroma block is as follows: the number of reference samples for long side is equal to the LSS, while the number of reference samples for short side is equal to double of SSS (2*SSS). For example, in reference to FIG. 13, the LSS is equal to W, and the SSS is equal to H. Thus, in the third embodiment, for CCIP_A mode, the top template (A) 1302 will be used to calculate the linear model coefficients, and the number of reference samples is W. For CCIP_L mode, the left template (B) 1304 and the left template (C) 1306 will be used to calculate the linear model coefficients, and the number of reference samples is 2H.

[00137] FIG. 14 illustrates a fourth embodiment 1400 for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure. The fourth embodiment can only be used for MDLM intra-prediction mode. Hence, the fourth embodiment may be performed by an intra-picture estimation component 215 and/or an intra-picture prediction component 217 of a codec system

200, an intra-picture prediction component 317 of an encoder 300, and/or an intra-picture prediction component 417 of a decoder 400. Specifically, MMLM intra-prediction can be employed during block compression at step 105 of method 100 and during block decoding at step 113 of method 100 at an encoder and a decoder, respectively.

[00138] In the fourth embodiment, for MDLM, the number of reference samples that will be used to calculate linear model coefficients of a linear model for a current chroma block is as follows: the number of reference samples for long side is equal to the LSS, while the number of reference samples for short side is equal to the minimum value of LSS and double of SSS (($M = \min(W, 2H)$). In an alternative embodiment, the number of reference samples for long side is equal to the LSS, while the number of reference samples for short side is equal to the maximum value of LSS and double of SSS ($M = \max(W, 2H)$). In these embodiments, the order to read the samples is: for left template, from the top of the left template (B) 1404 to bottom of the left template (B) 1404 and then from the top of the left template (C) 1406 to bottom of the left template (C) 1406, and for the top template (A) 1402, from left to right.

[00139] By using the third and fourth embodiment, even for a non-square block, the number of reference samples to calculate the linear model coefficients is a value of power of 2, for MDLM intra-prediction mode.

[00140] FIG. 15 illustrates a fifth embodiment 1500 for selecting a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block in accordance with the present disclosure. The fifth embodiment can only be used for MDLM intra-prediction mode. Hence, the fifth embodiment may be performed by an intra-picture estimation component 215 and/or an intra-picture prediction component 217 of a codec system 200, an intra-picture prediction component 317 of an encoder 300, and/or an intra-picture prediction component 417 of a decoder 400. Specifically, MMLM intra-prediction can be employed during block compression at step 105 of method 100 and during block decoding at step 113 of method 100 at an encoder and a decoder, respectively.

[00141] In the fifth embodiment, for MDLM, the number of reference samples that will be used to calculate linear model coefficients of a linear model for a current chroma block is as follows: the number of reference samples for long side is equal to the LSS, while the number of reference samples for short side is equal to the short side of any of the embodiments described above (first, second, third, or fourth embodiment). The difference in the fifth embodiment from say the third embodiment is that the positions of selected samples are different. For example, the samples that are at the long side and near the short side are discarded, and the number of the discarded samples is equal to the SSS. The fifth embodiment is illustrated in FIG. 15, where the top template section (A1) 1502 of size H (size of the SSS) is discarded from the original top template comprising of the top template section (A1) 1502 and the top template section (A2) 1504. A top template section (A3) 1506 having size equal to the SSS (i.e., the H in the depicted embodiments) is added to the remaining portion of the top template section (i.e., the top template section (A2) 1504). The number of reference samples of the top template section (A2) 1504 and the top template section (A3) 1506 is then used to calculate the linear model coefficients.

[00142] By using the proposed fifth method, even for a non-square block, the number of reference samples to calculate the linear model coefficients is a value of power of 2, for MDLM mode.

[00143] FIG. 16 is a flowchart of an example method 1600 for deriving a prediction for a chroma block using CCIP intra-prediction mode or MDLM intra-prediction mode. The method 1600 can be may be performed by an intra-picture estimation component 215 and/or an intra-picture prediction component 217 of a codec system 200 as well as an intra-picture prediction component 417 of a decoder 400. Specifically, the method 1600 can be employed during block decoding at step 113 of method 100 at a decoder.

[00144] The method 1600 begins at step 1602, by obtaining the corresponding down-sampled luma samples to a current chroma block as described above. The current chroma block has a W and an H that are not equal.

[00145] The method 1600, at step 1604, obtains a top template and a left template. The top template is top neighboring reconstructed chroma samples and a corresponding down-sampled top neighboring reconstructed luma samples. The left template is left neighboring reconstructed chroma samples and a corresponding down-sampled left neighboring reconstructed luma samples.

[00146] In accordance with the various embodiments described herein, the number of selected reference samples of the top template and/or left template that is used to calculate linear model coefficients of a linear model for the current chroma block is guaranteed to be a value of power of 2.

[00147] For example, in the first embodiment described herein, for CCIP intra-prediction mode, one of the top template or the left template is extended to be equal in size to a LSS such that both the top template and the left template are equal in size to the LSS. As described above, the LSS is the larger value of the W and the H. For MDLM intra-prediction mode, if CCIP_A intra-prediction mode is used, the top template is extended to be equal in size to a LSS to increase the number of reference samples if the top template is not equal in size to the LSS. If

CCIP_L intra-prediction mode is used, the left template is extended to be equal in size to the LSS to increase the number of reference samples if the left template is not equal in size to the LSS.

[00148] In the second embodiment described herein, for CCIP intra-prediction mode, a sum of the number of reference samples of the top template and the left template is equal to a smallest integer value (M) not smaller than W+H, and wherein M is a value of power of 2. For MDLM intra-prediction mode, if CCIP_A intra-prediction mode is used, the top template is extended to be equal in size to a smallest integer value (M) not smaller than W+H. If CCIP_L intra-prediction mode is used, the left template is extended to be equal in size to the smallest integer value (M) not smaller than W+H.

[00149] In the third embodiment described herein, for MDLM intra-prediction mode only, if CCIP_A intra-prediction mode is used, a top template of size W is obtained, and the top template is extended to double the width (2W) if W is less than H (i.e., W is the SSS). If CCIP_L intra-prediction mode is used, a left template of size H is obtained, and the left template is extended to double the height (2H) if H is less than W (i.e., H is the SSS).

[00150] In the fourth embodiment described herein, for MDLM intra-prediction mode only, if CCIP_A intra-prediction mode is used, a top template of size W is obtained, and the top template is extended to a minimum value of H and double of W (min (H, 2W)) if W is less than H. If CCIP_L intra-prediction mode is used, a left template of size H is obtained, and the left template is extended to a minimum value of W and double of H (min (W, 2H)) if H is less than W. In an alternative embodiment, for MDLM intra-prediction mode only, if CCIP_A intra-prediction mode is used, a top template of size W is obtained, and the top template is extended to a maximum value of H and double of W (max (H, 2W)) if W is less than H. If CCIP_L intra-prediction mode is used, a left template of size H is obtained, and the left template is extended to a maximum value of W and double of H (max (W, 2H)) if H is less than W.

[00151] In the fifth embodiment described herein, for MDLM intra-prediction mode only, if CCIP_A intra-prediction mode is used, a top template of size W is obtained, and, if W is less than H, the top template is extended to double the width (2W). However, if W is greater than H, a first set of samples of size H from a first end of the top template near a short side is discarding and a second set of samples of size H is added to a second end of the top template. If CCIP_L intra-prediction mode is used, a left template of size H is obtained, and, if H is less than W, the left template is extended to double the height (2H). However, if H is greater than W, a first set of samples of size W from a first end of the left template near a long side is discarded and a second set of samples of size W is added to a second end of the left template.

[00152] Once one of the above embodiments is used for selecting a number of reference samples, the method 1600, at step 1606, calculates the linear model coefficients of a linear model based on the selected reference samples. In an embodiment, the linear model is as follows:

[00153] $\mathrm{pred_C}(i, j) = \alpha \cdot \mathrm{rec_L}'(i, j) + \beta$,

[00154] where $\mathrm{pred_C}(i, j)$ represents predicted chroma samples of a coding unit (CU) and $\mathrm{rec_L}(i, j)$ represents down-sampled reconstructed luma samples of the same CU., and wherein the parameters α and β are the linear model coefficients. The linear model coefficients (parameters α and β) are calculated by minimizing a regression error between neighboring reconstructed luma and chroma samples around a current luma block and the current non-squared chroma block as follows:

[00155] $\alpha = \frac{N \cdot \sum(L(n) \cdot C(n)) - \sum L(n) \cdot \sum C(n)}{N \cdot \sum(L(n) \cdot L(n)) - \sum L(n) \cdot \sum L(n)}$

[00156] $\beta = \frac{\sum C(n) - \alpha \cdot \sum L(n)}{N}$

[00157] where $L(n)$ represents down-sampled top and left neighboring reconstructed luma samples, $C(n)$ represents top and left neighboring reconstructed chroma samples, and value of N is equal to the sum of the W and the H of the non-squared current chroma coding block.

[00158] At step 1608, the method 1600 derives a prediction for the current chroma block by using the calculated linear model coefficients of the linear model.

[00159] FIG. 17 is a schematic diagram of an example video coding device 1700. The video coding device 1700 is suitable for implementing the disclosed examples/embodiments as described herein. The video coding device 1700 comprises downstream ports 1720, upstream ports 1750, and/or transceiver units (Tx/Rx) 1710, including transmitters and/or receivers for communicating data upstream and/or downstream over a network. The video coding device 1700 also includes a processor 1730 including a logic unit and/or central processing unit (CPU) to process the data and a memory 1732 for storing the data. The video coding device 1700 may also comprise optical-to-electrical (OE) components, electrical-to-optical (EO) components, and/or wireless communication components coupled to the upstream ports 1750 and/or downstream ports 1720 for communication of data via optical or wireless communication networks. The video coding device 1700 may also include input and/or output (I/O) devices 1760 for communicating data to and from a user. The I/O devices 1760 may include output devices such as a display for displaying video data, speakers for outputting audio data, etc. The

I/O devices 1760 may also include input devices, such as a keyboard, mouse, trackball, etc., and/or corresponding interfaces for interacting with such output devices.

[00160] The processor 1730 is implemented by hardware and software. The processor 1730 may be implemented as one or more CPU chips, cores (e.g., as a multi-core processor), field-programmable gate arrays (FPGAs), application specific integrated circuits (ASICs), and digital signal processors (DSPs). The processor 1730 is in communication with the downstream ports 1720, Tx/Rx 1710, upstream ports 1750, and memory 1732. The processor 1730 comprises a coding module 1714. The coding module 1714 implements the disclosed embodiments described above, such as methods 100, 1500, and 1600 and/or mechanisms 1300 and/1400 as part of intra-prediction 600, CCLM intra-prediction 700, mechanism 800, MDLM intra-prediction using CCIP_A mode 900 and CCIP_L mode 1000, and/or MMLM intra-prediction as depicted in graph 1100, as well as any other method/mechanism described herein. Further, the coding module 1714 may implement a codec system 200, an encoder 300, and/or a decoder 400. For example, the coding module 1714 can be employed to perform cross-component intra-prediction to code chroma samples based on downsampled luma samples from the same CU and based on a comparison of neighboring chroma reference samples and downsampled neighboring luma reference samples. This may be accomplished according to equation 1 and equations 2-3 and/or according to equation 1. Specifically, the coding module 1714 can be configured in according to various embodiments to select a number of reference samples to calculate linear model coefficients of a linear model for a current chroma block such that the number of selected reference samples is a value of power of 2. As such, coding module 1714 improves the functionality of the video coding device 1700 as well as addresses problems that are specific to the video coding arts. Further, coding module 1714 effects a transformation of the video coding device 1700 to a different state. Alternatively, the coding module 1714 can be implemented as instructions stored in the memory 1732 and executed by the processor 1730 (e.g., as a computer program product stored on a non-transitory medium).

[00161] The memory 1732 comprises one or more memory types such as disks, tape drives, solid-state drives, read only memory (ROM), random access memory (RAM), flash memory, ternary content-addressable memory (TCAM), static random-access memory (SRAM), etc. The memory 1732 may be used as an over-flow data storage device, to store programs when such programs are selected for execution, and to store instructions and data that are read during program execution.

[00162] While several embodiments have been provided in the present disclosure, it may be understood that the disclosed systems and methods might be embodied in many other specific forms without departing from the spirit or scope of the present disclosure. The present examples are to be considered as illustrative and not restrictive, and the intention is not to be limited to the details given herein. For example, the various elements or components may be combined or integrated in another system or certain features may be omitted, or not implemented.

[00163] In addition, techniques, systems, subsystems, and methods described and illustrated in the various embodiments as discrete or separate may be combined or integrated with other systems, components, techniques, or methods without departing from the scope of the present disclosure. Other examples of changes, substitutions, and alterations are ascertainable by one skilled in the art and may be made without departing from the spirit and scope disclosed herein.

CLAIMS

What is claimed is:

1. A method implemented in an encoder for intra prediction for a current chroma block using cross-component intra prediction (CCIP) mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

obtaining a top template and a left template, wherein one of the top template and the left template is extended to be equal in size to a long side size (LSS) such that both the top template and the left template are equal in size to the LSS, and wherein the LSS is the larger value of the W and the H;

calculating linear model coefficients of a linear model based on a number of reference samples of both the top template and the left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

2. A method implemented in an encoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template, wherein the top template is extended to be equal in size to a long side size (LSS) to obtain an extended top template if the top template is not equal in size to the LSS, and wherein the LSS is the larger value of the W and the H;

for the CCIP_L mode, obtaining a left template, wherein the left template is extended to be equal in size to the LSS to obtain an extended left template if the left template is not equal in size to the LSS;

for the CCIP_A mode, if the top template is equal in size to the LSS, calculating linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if the top template is not equal in size to the LSS, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended top template;

for the CCIP_L mode, if the left template is equal in size to the LSS, calculating the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if the left template is not equal in size to the LSS, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

3. A method implemented in an encoder for intra prediction for a current chroma block using cross-component intra prediction (CCIP) mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

obtaining a top template and a left template, wherein a sum of the number of reference samples of the top template and the left template is equal to a smallest integer value (M), wherein M is not smaller than W+H, and wherein M is a value of power of 2;

calculating the linear model coefficients of a linear model based on a number of reference samples of both the top template and the left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

4. A method implemented in an encoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template, wherein the top template is extended to be equal in size to a smallest integer value (M) not smaller than W+H to obtain an extended top template;

for the CCIP_L mode, obtaining a left template, wherein the left template is extended to be equal in size to the smallest integer value (M) not smaller than W+H to obtain an extended left template;

calculating linear model coefficients of a linear model based on a number of reference samples of the extended top template in the CCIP_A mode, and calculating linear model coefficients of a linear model based on the number of reference samples of the extended left template in the CCIP_L mode; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

5. A method implemented in an encoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template of size W, and, if W is less than H, the top template is extended to be double the width (2W) to obtain an extended top template;

for the CCIP_L mode, obtaining a left template of size H, and, if H is less than W, the left template is extended to be double the height (2H) to obtain an extended left template;

for the CCIP_A mode, if W is greater than H, calculating linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended top template; or

for the CCIP_L mode, if H is greater than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

6. A method implemented in an encoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template of size W, and, if W is less than H, the top template is extended to a minimum value of H and double of W (min (H, 2W)) to obtain an extended top template;

for the CCIP_L mode, obtaining a left template of size H, and, if H is less than W, the left template is extended to a minimum value of W and double of H (min (W, 2H)) to obtain an extended left template;

for the CCIP_A mode, if W is greater than H, calculating linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended top template;

for the CCIP_L mode, if H is greater than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

7. A method implemented in an encoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template of size W, and, if W is less than H, the top template is extended to a maximum value of H and double of W (max (H, 2W)) to obtain an extended top template;

for the CCIP_L mode, obtaining a left template of size H, and, if H is less than W, the left template is extended to a maximum value of W and double of H (max (W, 2H)) to obtain an extended left template;

for the CCIP_A mode, if W is greater than H, calculating the linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended top template;

for the CCIP_L mode, if H is greater than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

8. A method implemented in an encoder for deriving a prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template of size W, and, if W is less than H, the top template is extended to double the width (2W) to obtain an extended top template, and, if W is greater than H, a first set of samples of size H is discarded from a first end of the top template near a short side and a second set of samples of size H is added to a second end of the top template to obtain a modified top template;

for the CCIP_L mode, obtaining a left template of size H, and, if H is less than W, the left template is extended to double the height (2H) to obtain an extended left template, and, if H is greater than W, a first set of samples of size W is discarded from a first end of the left template near a long side and a second set of samples of size W is added to a second end of the left template to obtain a modified left template;

for the CCIP_A mode, if W is less than H, calculating linear model coefficients of a linear model based on a number of reference samples of the extended top template, otherwise, if W is greater than H, calculating the linear model coefficients of the linear model based on the number of reference samples of the modified top template;

for the CCIP_L mode, if H is less than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template, otherwise, if H is greater than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the modified left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

9. A method implemented in a decoder for intra prediction for a current chroma block using cross-component intra prediction (CCIP) mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

obtaining a top template and a left template, wherein one of the top template and the left template is extended to be equal in size to a long side size (LSS) such that both the top template and the left template are equal in size to the LSS, and wherein the LSS is the larger value of the W and the H;

calculating linear model coefficients of a linear model based on a number of reference samples of both the top template and the left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

10. A method implemented in a decoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template, wherein the top template is extended to be equal in size to a long side size (LSS) to obtain an extended top template if the top template is not equal in size to the LSS, and wherein the LSS is the larger value of the W and the H;

for the CCIP_L mode, obtaining a left template, wherein the left template is extended to be equal in size to the LSS to obtain an extended left template if the left template is not equal in size to the LSS;

for the CCIP_A mode, if the top template is equal in size to the LSS, calculating linear model coefficients of a linear model based on the number of reference samples of the top template, otherwise, if the top template is not equal in size to the LSS, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended top template;

for the CCIP_L mode, if the left template is equal in size to the LSS, calculating the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if the left template is not equal in size to the LSS, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

11. A method implemented in a decoder for intra prediction for a current chroma block using cross-component intra prediction (CCIP) mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

obtaining a top template and a left template, wherein a sum of the number of reference samples of the top template and the left template is equal to a smallest integer value (M), wherein M is not smaller than W+H, and wherein M is a value of power of 2;

calculating the linear model coefficients of a linear model based on a number of reference samples of both the top template and the left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

12. A method implemented in a decoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template, wherein the top template is extended to be equal in size to a smallest integer value (M) not smaller than W+H to obtain an extended top template;

for the CCIP_L mode, obtaining a left template, wherein the left template is extended to be equal in size to the smallest integer value (M) not smaller than W+H to obtain an extended left template;

calculating linear model coefficients of a linear model based on a number of reference samples of the extended top template in the CCIP_A mode, and calculating linear model coefficients of a linear model based on the number of reference samples of the extended left template in the CCIP_L mode; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

13. A method implemented in a decoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template of size W, and, if W is less than H, the top template is extended to be double the width (2W) to obtain an extended top template;

for the CCIP_L mode, obtaining a left template of size H, and, if H is less than W, the left template is extended to be double the height (2H) to obtain an extended left template;

for the CCIP_A mode, if W is greater than H, calculating linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended top template; or

for the CCIP_L mode, if H is greater than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

14. A method implemented in a decoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template of size W, and, if W is less than H, the top template is extended to a minimum value of H and double of W (min (H, 2W)) to obtain an extended top template;

for the CCIP_L mode, obtaining a left template of size H, and, if H is less than W, the left template is extended to a minimum value of W and double of H (min (W, 2H)) to obtain an extended left template;

for the CCIP_A mode, if W is greater than H, calculating linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended top template;

for the CCIP_L mode, if H is greater than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

15. A method implemented in a decoder for intra prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template of size W, and, if W is less than H, the top template is extended to a maximum value of H and double of W (max (H, 2W)) to obtain an extended top template;

for the CCIP_L mode, obtaining a left template of size H, and, if H is less than W, the left template is extended to a maximum value of W and double of H (max (W, 2H)) to obtain an extended left template;

for the CCIP_A mode, if W is greater than H, calculating the linear model coefficients of a linear model based on a number of reference samples of the top template, otherwise, if W is less than H, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended top template;

for the CCIP_L mode, if H is greater than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the left template, otherwise, if H is less than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

16. A method implemented in a decoder for deriving a prediction for a current chroma block using one of a multi-directional linear model (MDLM) mode comprising a CCIP_A mode and a CCIP_L mode, the method comprising:

obtaining a down-sampled luma block corresponding to the current chroma block, the current chroma block having a width (W) and a height (H) that are not equal;

for the CCIP_A mode, obtaining a top template of size W, and, if W is less than H, the top template is extended to double the width (2W) to obtain an extended top template, and, if W is greater than H, a first set of samples of size H is discarded from a first end of the top template near a short side and a second set of samples of size H is added to a second end of the top template to obtain a modified top template;

for the CCIP_L mode, obtaining a left template of size H, and, if H is less than W, the left template is extended to double the height (2H) to obtain an extended left template, and, if H is greater than W, a first set of samples of size W is discarded from a first end of the left template near a long side and a second set of samples of size W is added to a second end of the left template to obtain a modified left template;

for the CCIP_A mode, if W is less than H, calculating linear model coefficients of a linear model based on a number of reference samples of the extended top template, otherwise, if W is greater than H, calculating the linear model coefficients of the linear model based on the number of reference samples of the modified top template;

for the CCIP_L mode, if H is less than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the extended left template, otherwise, if H is greater than W, calculating the linear model coefficients of the linear model based on the number of reference samples of the modified left template; and

deriving a prediction for the current chroma block using the calculated linear model coefficients of the linear model.

17. The method of according to any of the preceding claims, wherein the top template is top neighboring reconstructed chroma samples and corresponding down-sampled top neighboring reconstructed luma samples, and wherein the left template is left neighboring reconstructed chroma samples and corresponding down-sampled left neighboring reconstructed luma samples.

18. The method of according to any of the preceding claims, wherein the number of reference samples of both the top template and the left template used in calculating the linear model coefficients is a value of power of 2.

19. The method of according to any of the preceding claims, wherein the linear model is as follows:

$$pred_C(i,j) = \alpha \cdot rec_L'(i,j) + \beta,$$

where $pred_C(i,j)$ represents predicted chroma samples of a coding unit (CU) and $rec_L(i,j)$ represents down-sampled reconstructed luma samples of the same CU, and wherein the parameters α and β are the linear model coefficients of the linear model.

20. The method of according to any of the preceding claims, wherein the linear model coefficients, are calculated by minimizing a regression error between neighboring reconstructed

luma and chroma samples around a current luma block and the current non-squared chroma block as follows:

$$\alpha = \frac{N \cdot \sum(L(n) \cdot C(n)) - \sum L(n) \cdot \sum C(n)}{N \cdot \sum(L(n) \cdot L(n)) - \sum L(n) \cdot \sum L(n)}$$

$$\beta = \frac{\sum C(n) - \alpha \cdot \sum L(n)}{N}$$

where $L(n)$ represents down-sampled top and left neighboring reconstructed luma samples, $C(n)$ represents top and left neighboring reconstructed chroma samples, and value of N is equal to the number of reference samples of both the top template and the left template used to derive the linear model coefficients.

21. The method according to any of claims 3 and 11, and any preceding dependent claims of claims 3 and 11, where N=(W+H), and wherein an additional (M-N) number of samples is re-sampled from an existing N samples.

22. The method according to any of claims 3 and 11, and any preceding dependent claims of claims 3 and 11, wherein resampling is performed from a bottom of an existing left template to a top of the existing left template, and then from a left of an existing top template to a right of the existing top template.

23. The method according to any of claims 4 and 12, and any preceding dependent claims of claims 4 and 12, where N=(W+H), and wherein an additional (M-N) number of samples is re-sampled from the extended top template in CCIP_A mode, and wherein the additional (M-N) number of samples is re-sampled from the extended left template in CCIP_L mode.

24. The method according to any of claims 4 and 12, and any preceding dependent claims of claims 4 and 12, wherein resampling is performed from a bottom of the extended left template to a top of the extended left template in CCIP_L mode, and then from a left of an extended top template to a right of the extended top template in CCIP_A mode.

25. The method according to any of claims 3, 4, 11, and 12, and any preceding dependent claims of claims 3, 4, 11, and 12, wherein re-sampling from the existing N samples is performed in equal step size S, where S is an integer that is equaled to N divided by (M-N).

26. A video encoding device comprising:

a processor; and memory coupled to the processor, the processor configured to execute instructions in the memory to perform the method according to any of claims 1-8 and 17-25.

27. A video decoding device comprising:

a processor; and memory coupled to the processor, the processor configured to execute instructions in the memory to perform the method according to any of claims 9-16 and 17-25.

28. A non-transitory computer readable medium comprising a computer program product for use by a video coding device, the computer program product comprising computer executable instructions stored on the non-transitory computer readable medium such that when executed by a processor cause the video coding device to perform the method of any of claims 1-25.

29. An encoder comprising processing circuitry for carrying out the method according to any one of claims 1-8 and 17-25.

30. A decoder comprising processing circuitry for carrying out the method according to any one of claims 9-16 and 17-25.

31. An electronic device for coding video data, the electronic device comprising one or more processing units configured to perform any one of claims 1-25.

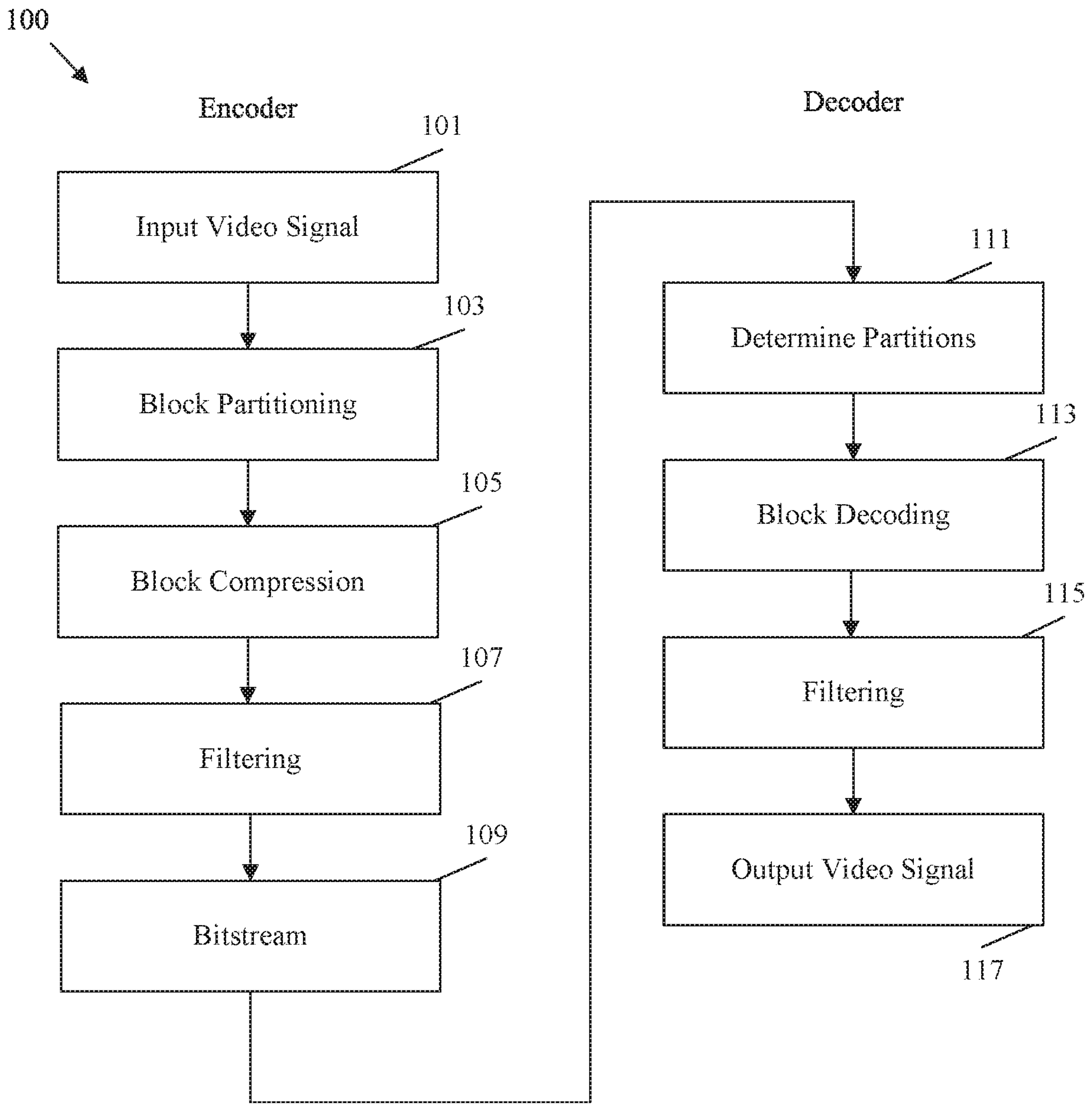
100
Encoder
Decoder
101
Input Video Signal
103
Block Partitioning
105
Block Compression
107
Filtering
109
Bitstream
111
Determine Partitions
113
Block Decoding
115
Filtering
Output Video Signal
117

FIG. 1

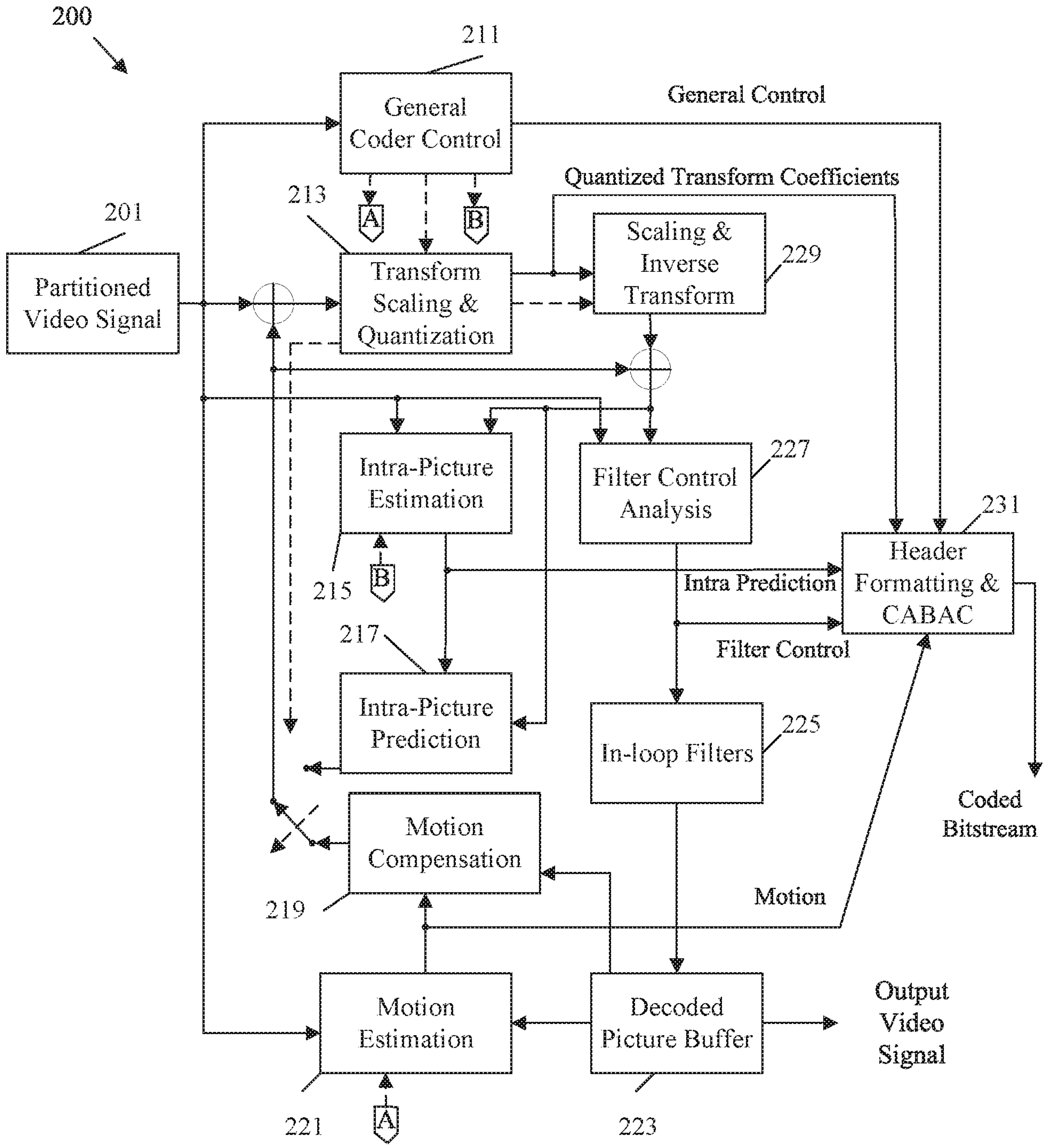
200
211
General Coder Control
General Control
Quantized Transform Coefficients
213
A
B
201
Partitioned Video Signal
Transform Scaling & Quantization
Scaling & Inverse Transform
229
227
Intra-Picture Estimation
Filter Control Analysis
231
Header Formatting & CABAC
215
B
Intra Prediction
217
Filter Control
Intra-Picture Prediction
In-loop Filters
225
Coded Bitstream
Motion Compensation
219
Motion
Motion Estimation
Decoded Picture Buffer
Output Video Signal
221
A
223

FIG. 2

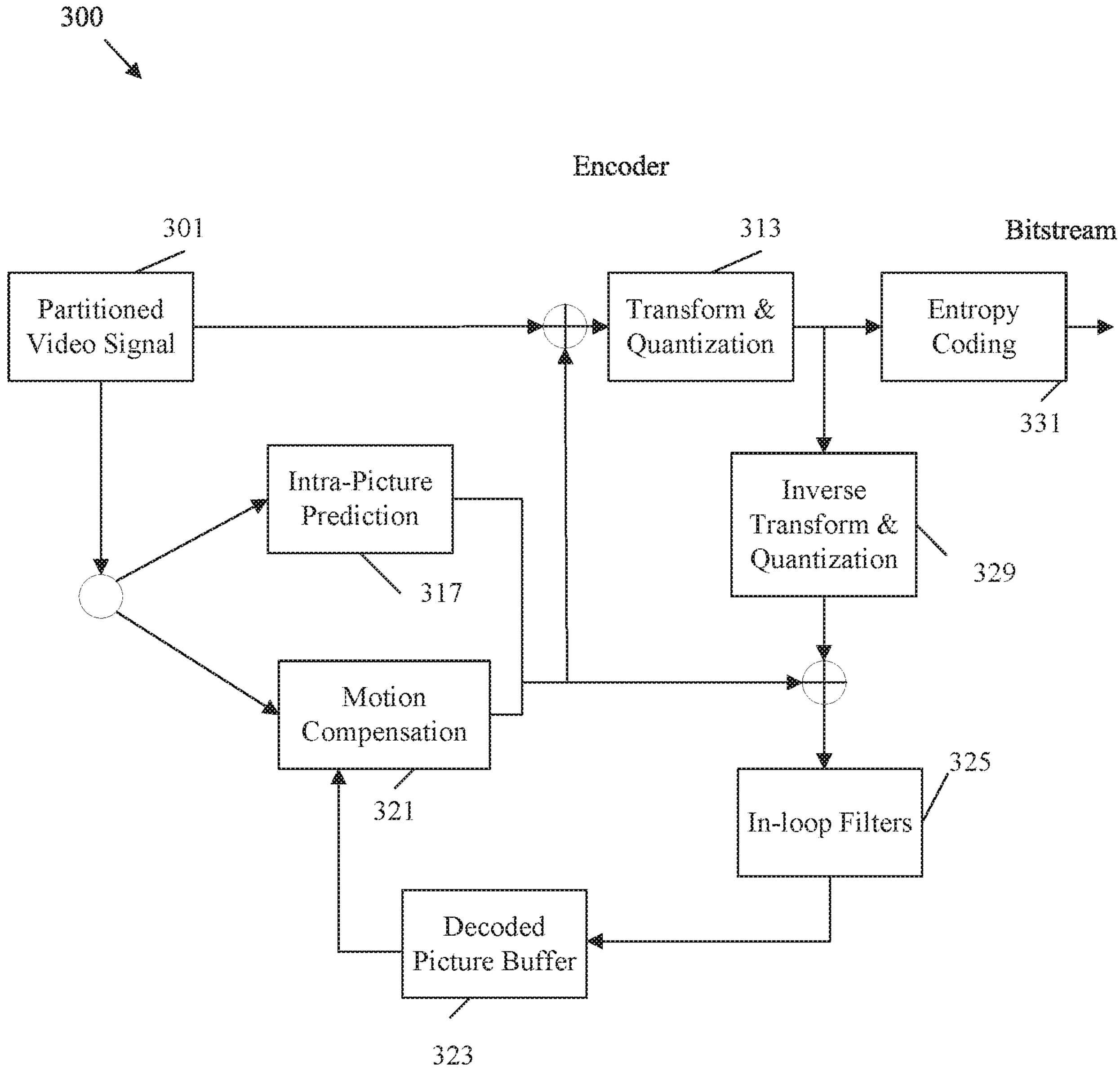
300
Encoder
301
313
Bitstream
Partitioned Video Signal
Transform & Quantization
Entropy Coding
331
Intra-Picture Prediction
Inverse Transform & Quantization
317
329
Motion Compensation
325
321
In-loop Filters
Decoded Picture Buffer
323

FIG. 3

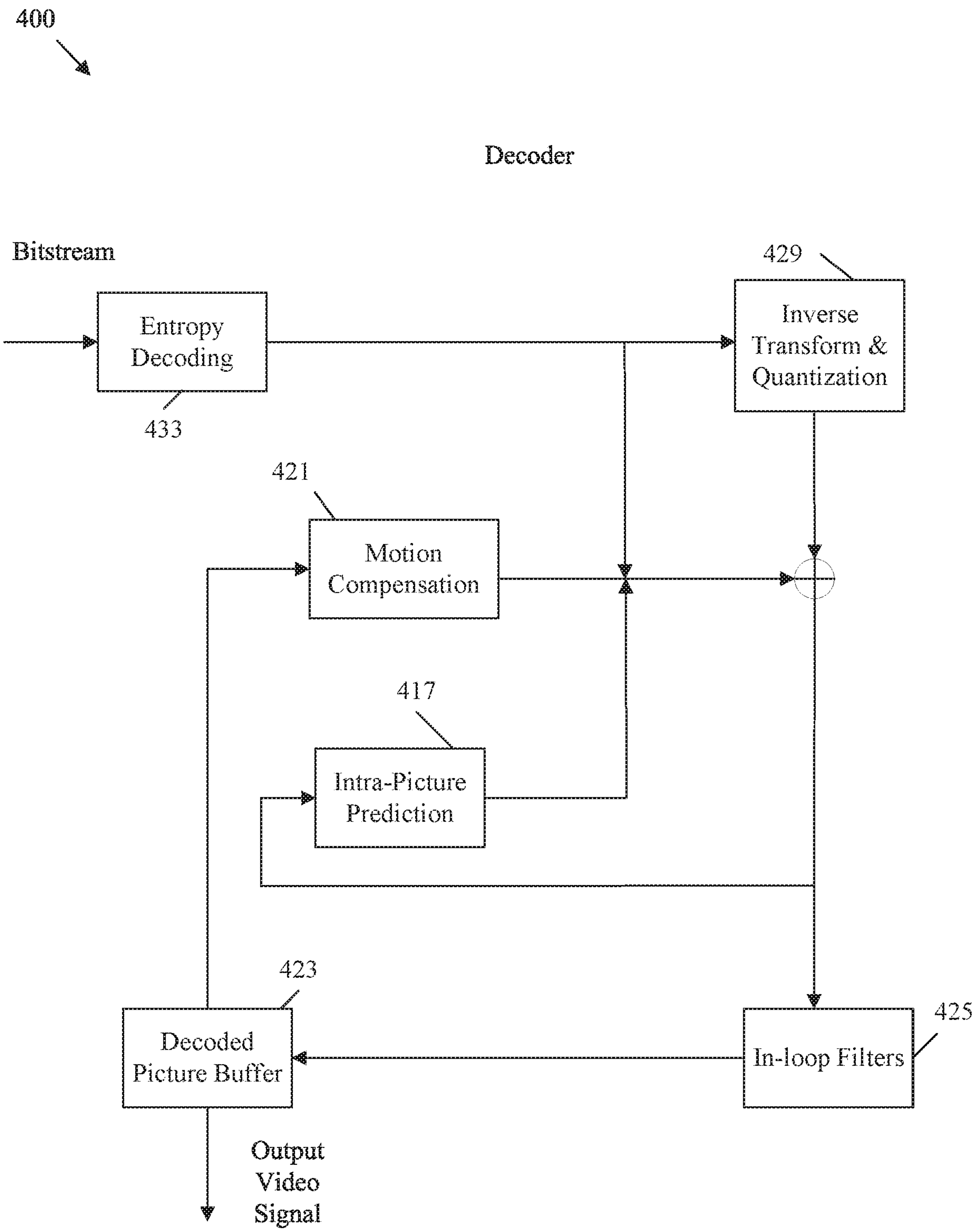
400
Decoder
Bitstream
Entropy Decoding
433
429
Inverse Transform & Quantization
421
Motion Compensation
417
Intra-Picture Prediction
423
Decoded Picture Buffer
425
In-loop Filters
Output Video Signal

FIG. 4

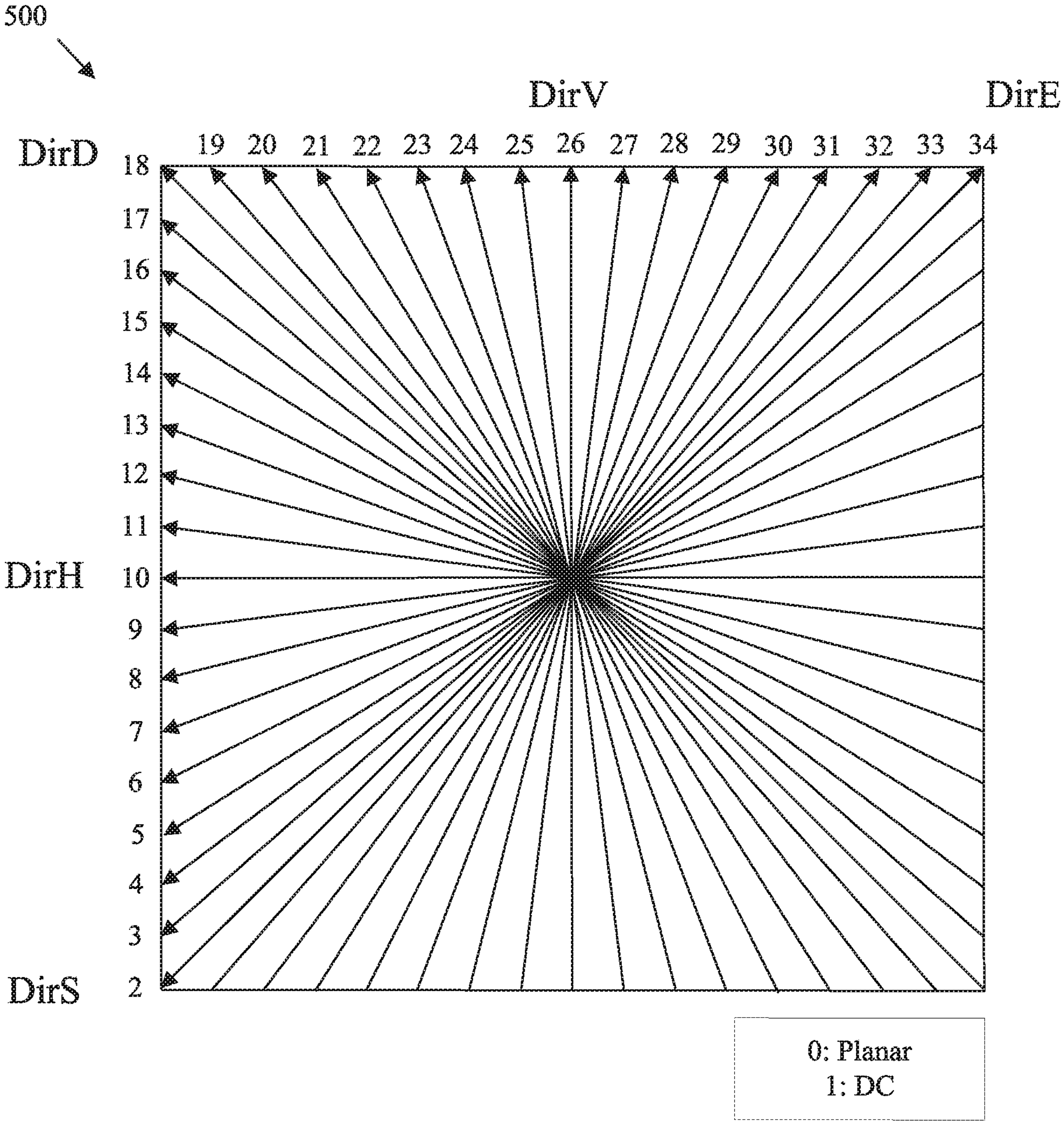
500
DirV
DirE
DirD
DirH
DirS
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
0: Planar
1: DC

FIG. 5

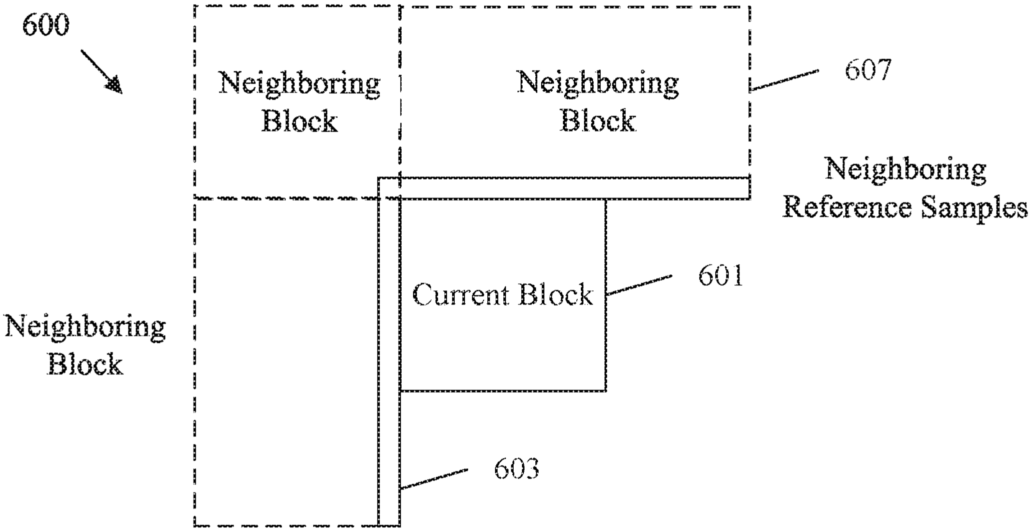
600
Neighboring Block
Neighboring Block
607
Neighboring Reference Samples
Current Block
601
Neighboring Block
603

FIG. 6

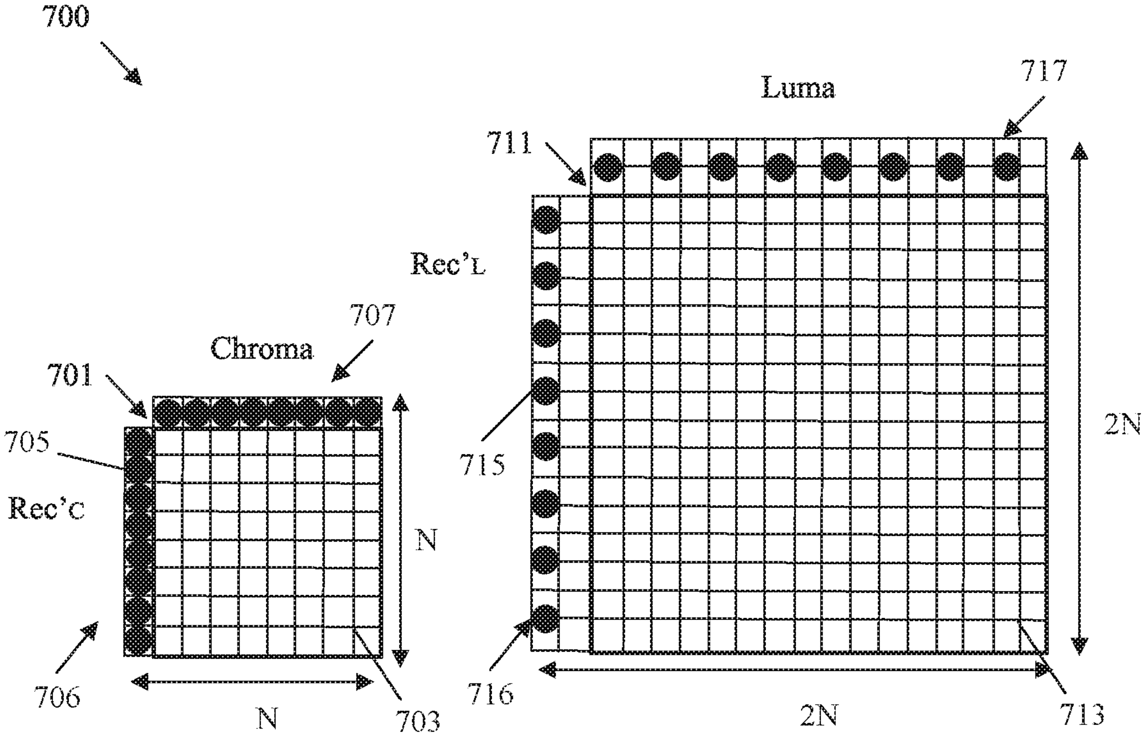
700
Luma
717
711
Rec'L
Chroma
707
701
705
Rec'C
N
715
2N
706
N
703
716
2N
713

FIG. 7

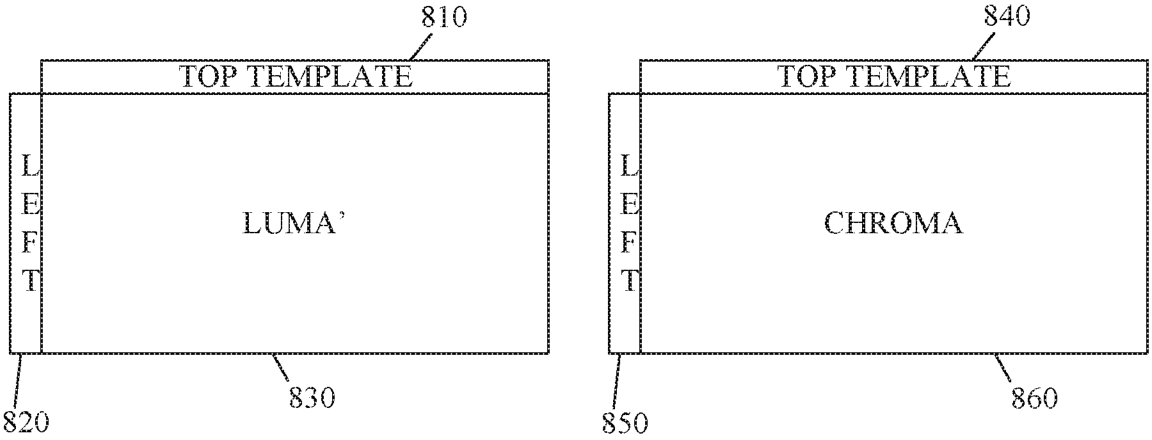
810
TOP TEMPLATE
LEFT
LUMA'
820
830
840
TOP TEMPLATE
LEFT
CHROMA
850
860

FIG. 8

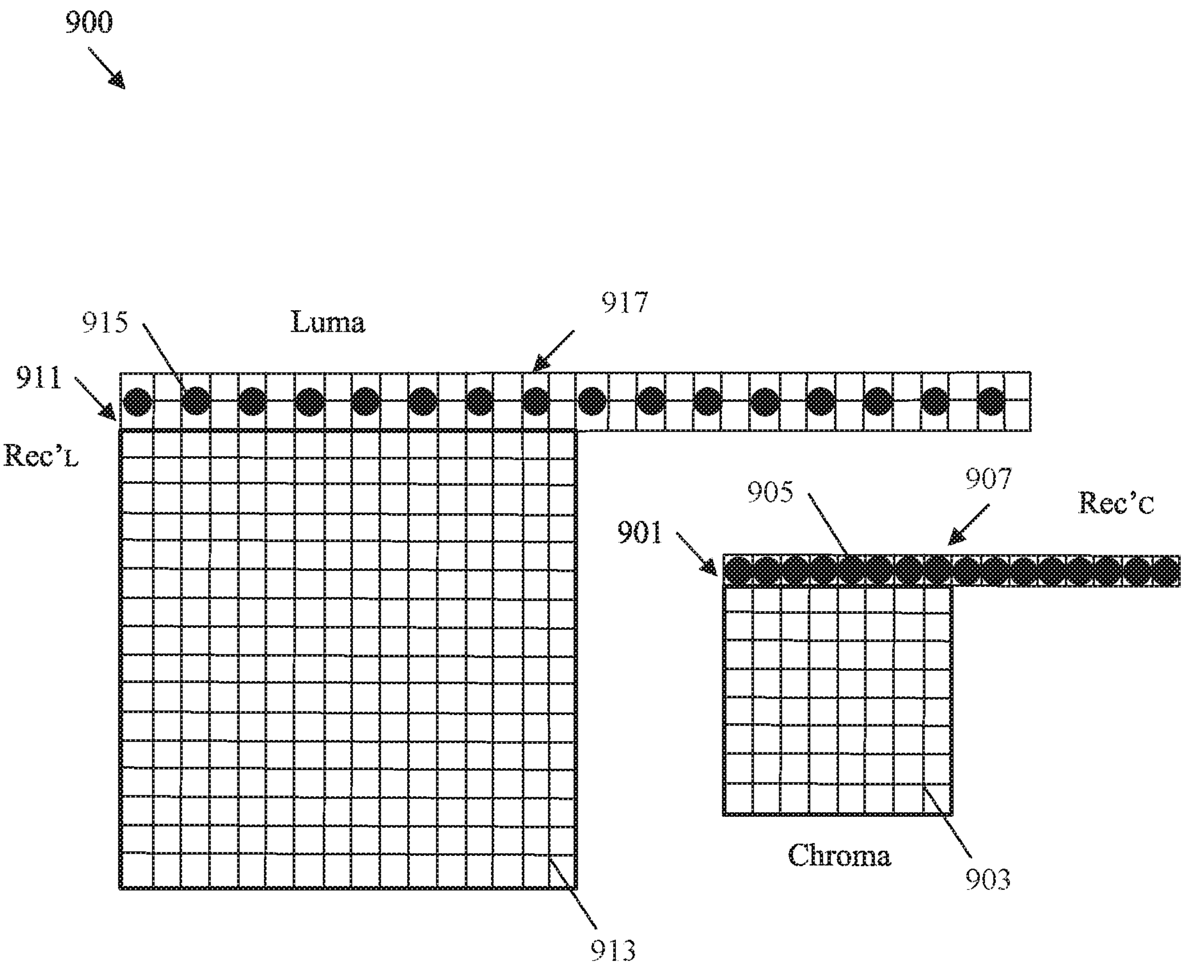

FIG. 9

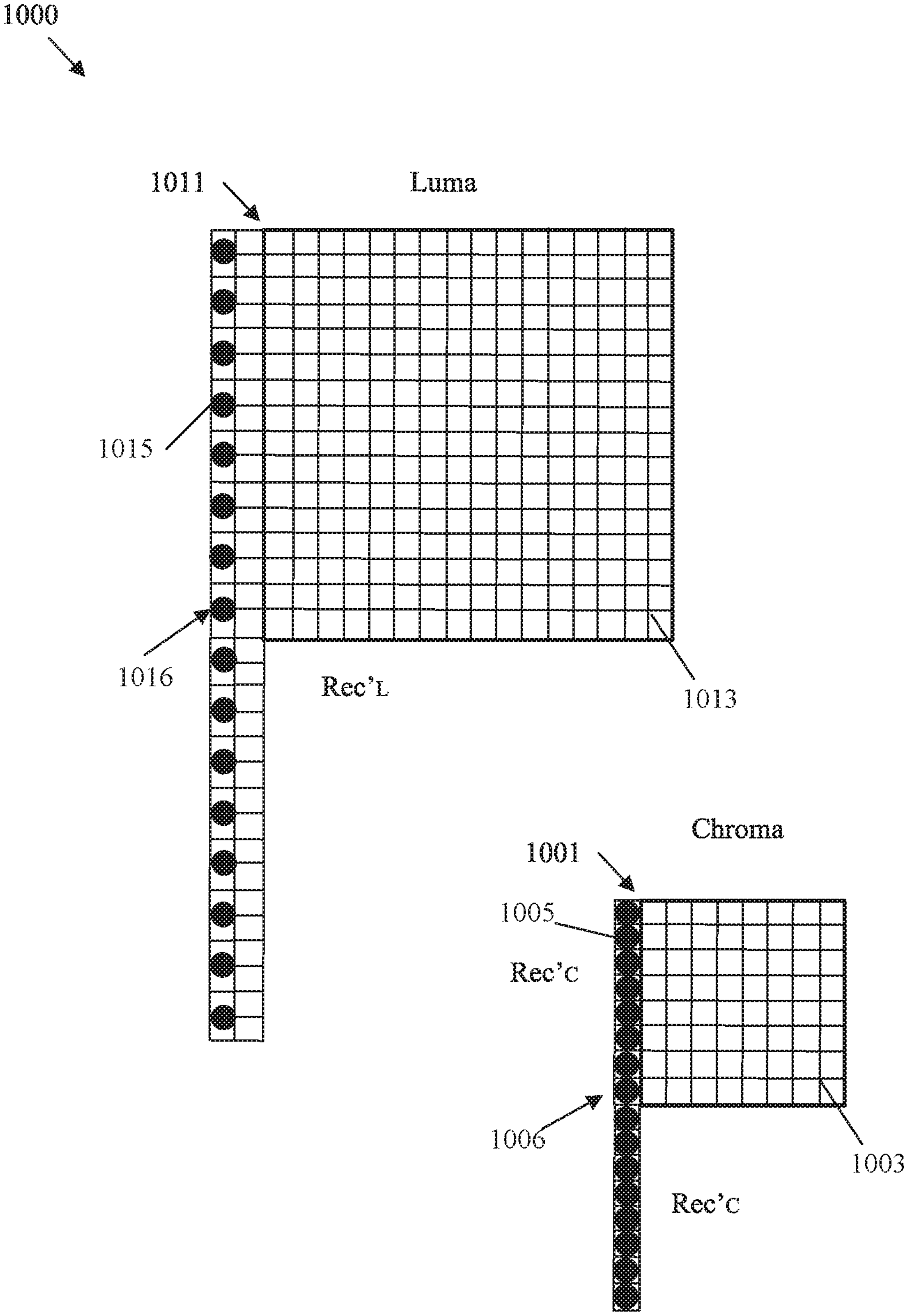
1000
1011
Luma
1015
1016
Rec'L
1013
Chroma
1001
1005
Rec'C
1006
1003
Rec'C

FIG. 10

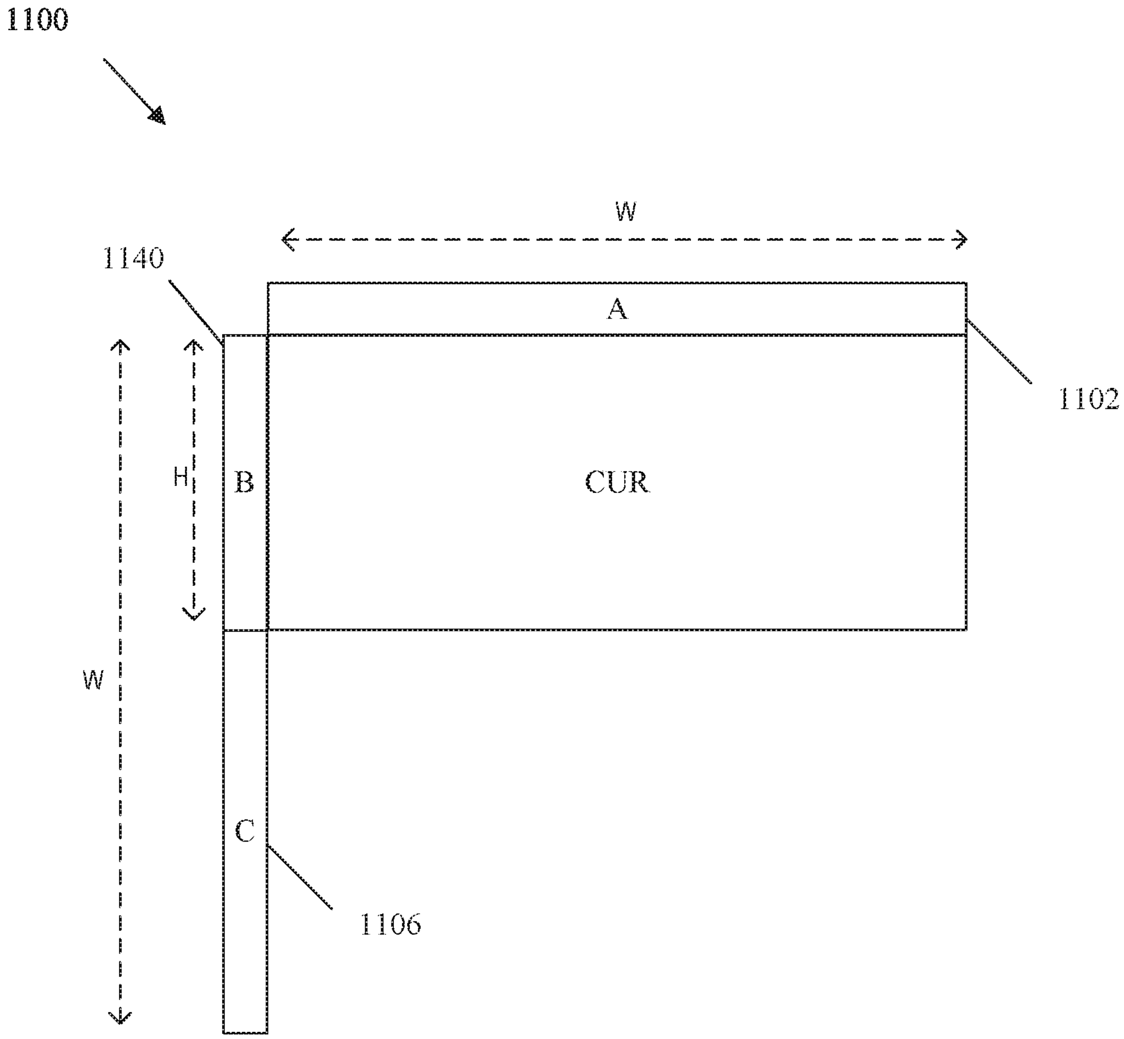

FIG. 11

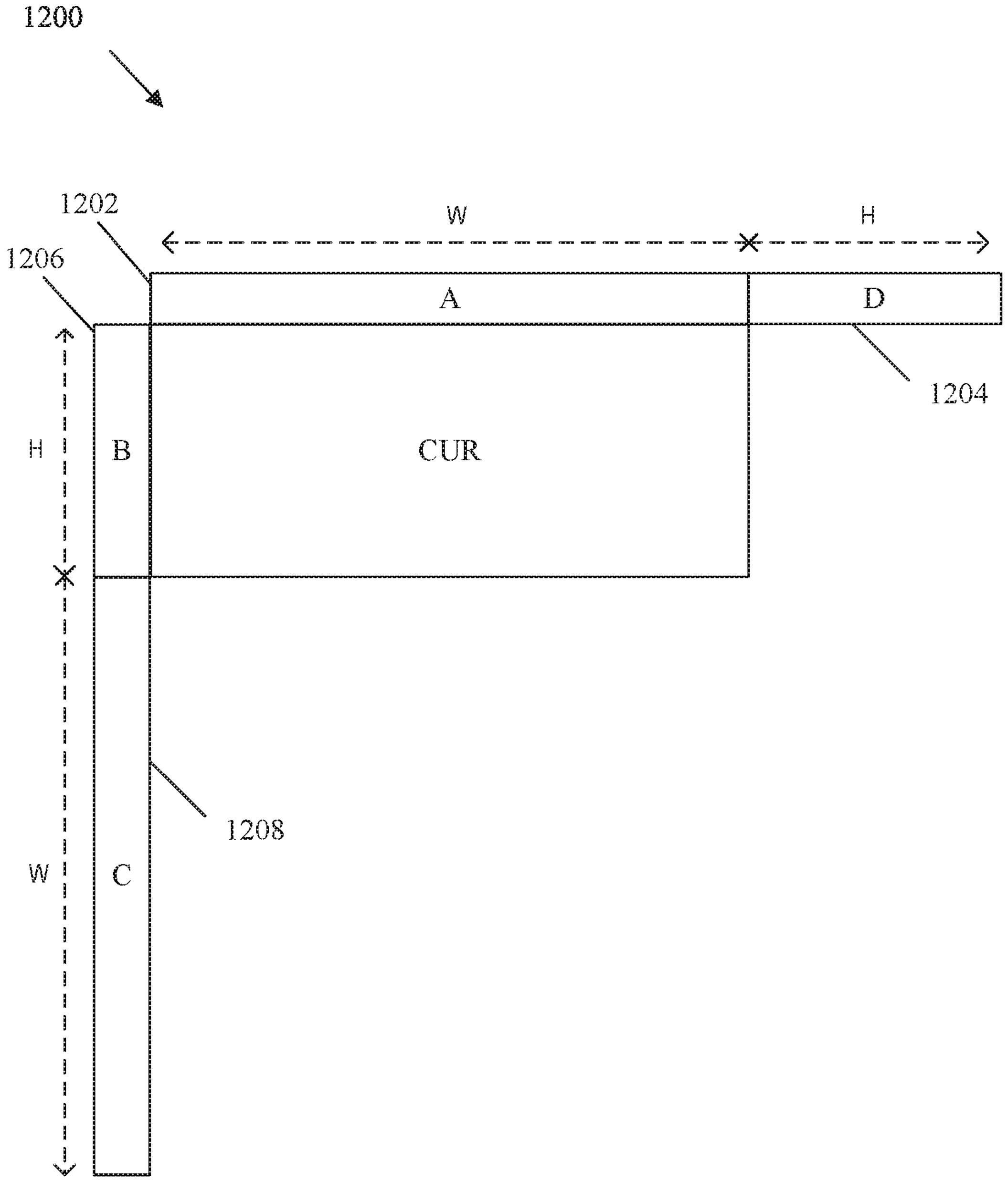

FIG. 12

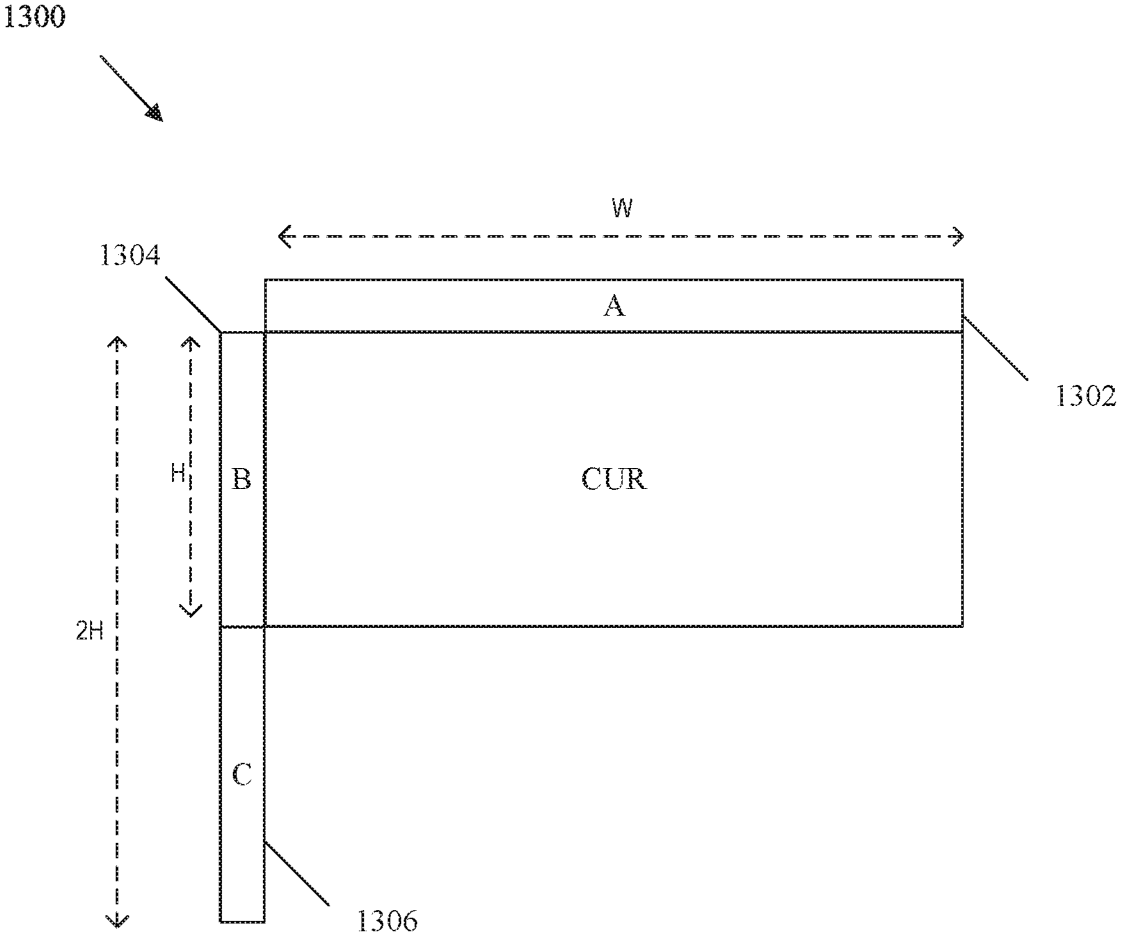

FIG. 13

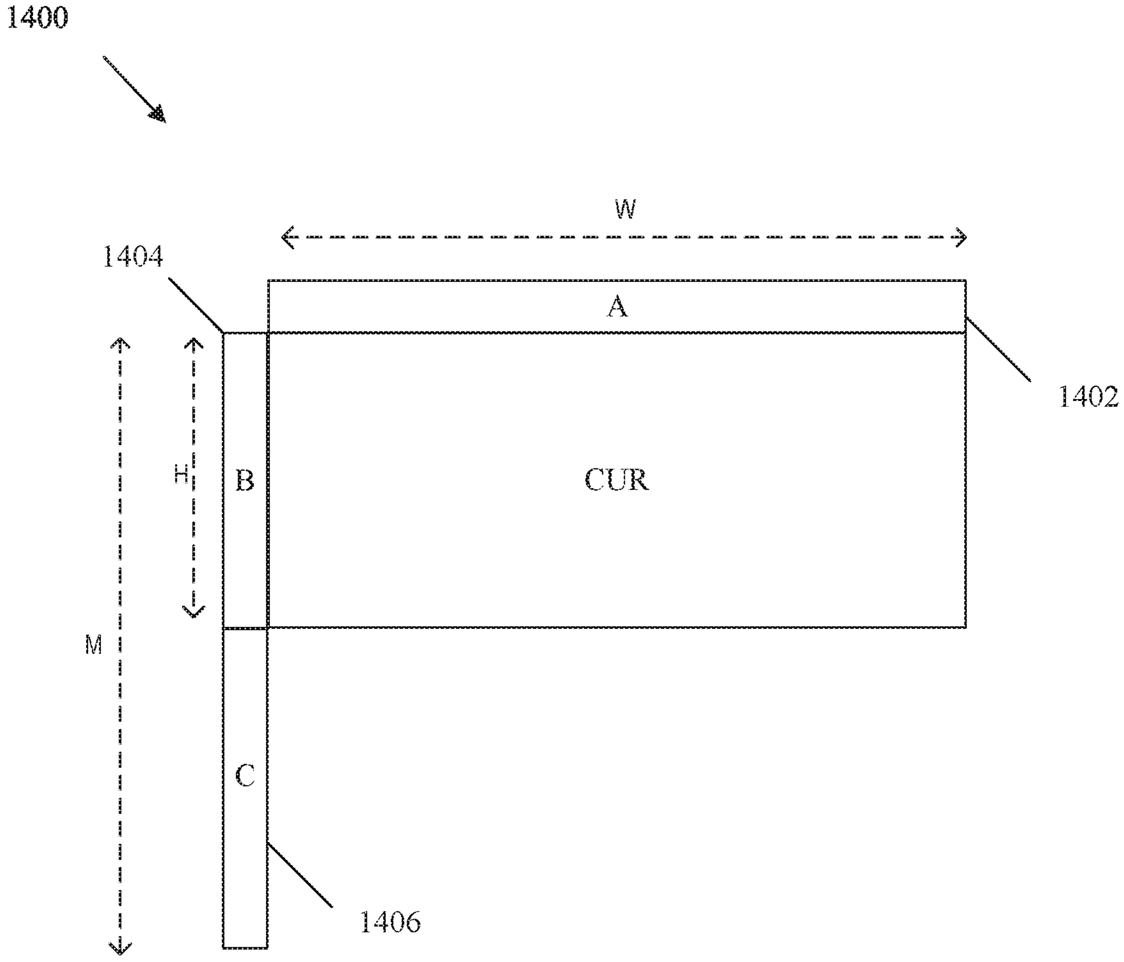

FIG. 14

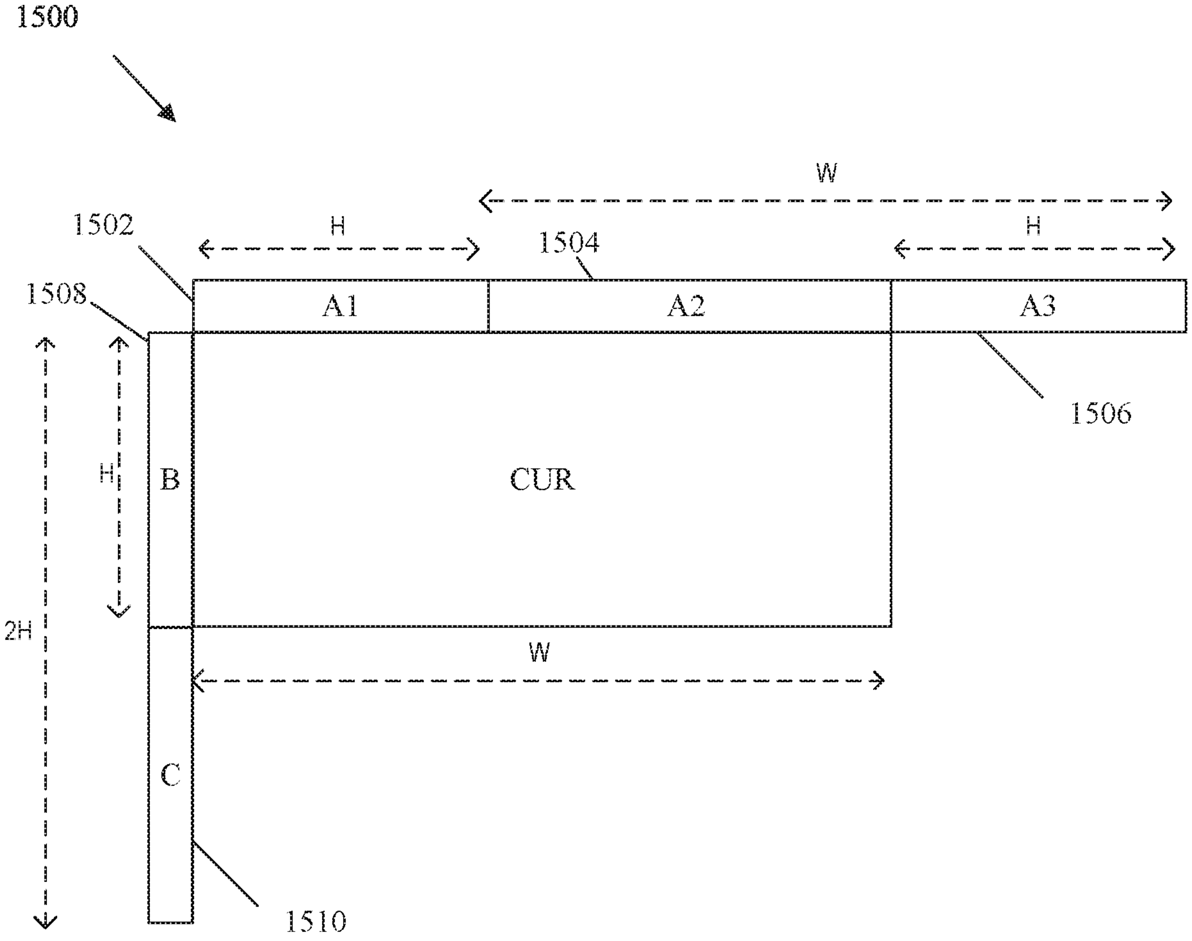

FIG. 15

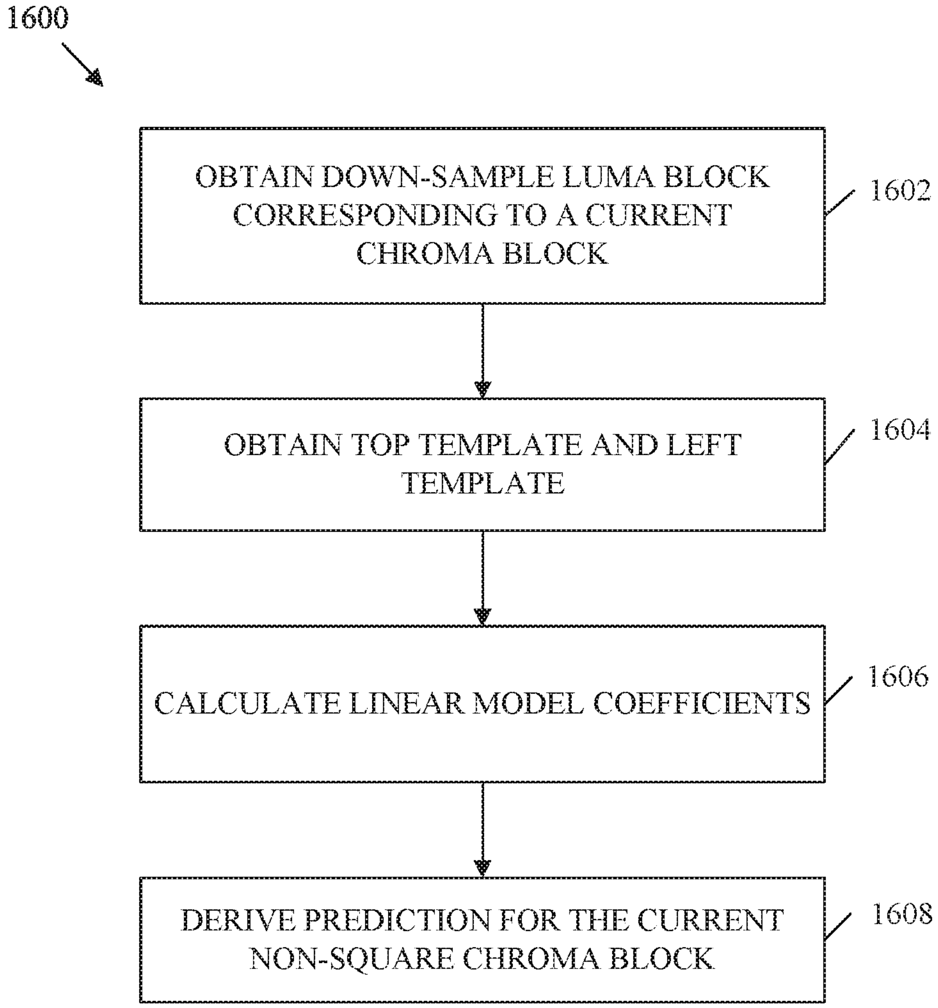
1600
OBTAIN DOWN-SAMPLE LUMA BLOCK CORRESPONDING TO A CURRENT CHROMA BLOCK
1602
OBTAIN TOP TEMPLATE AND LEFT TEMPLATE
1604
CALCULATE LINEAR MODEL COEFFICIENTS
1606
DERIVE PREDICTION FOR THE CURRENT NON-SQUARE CHROMA BLOCK
1608

FIG. 16

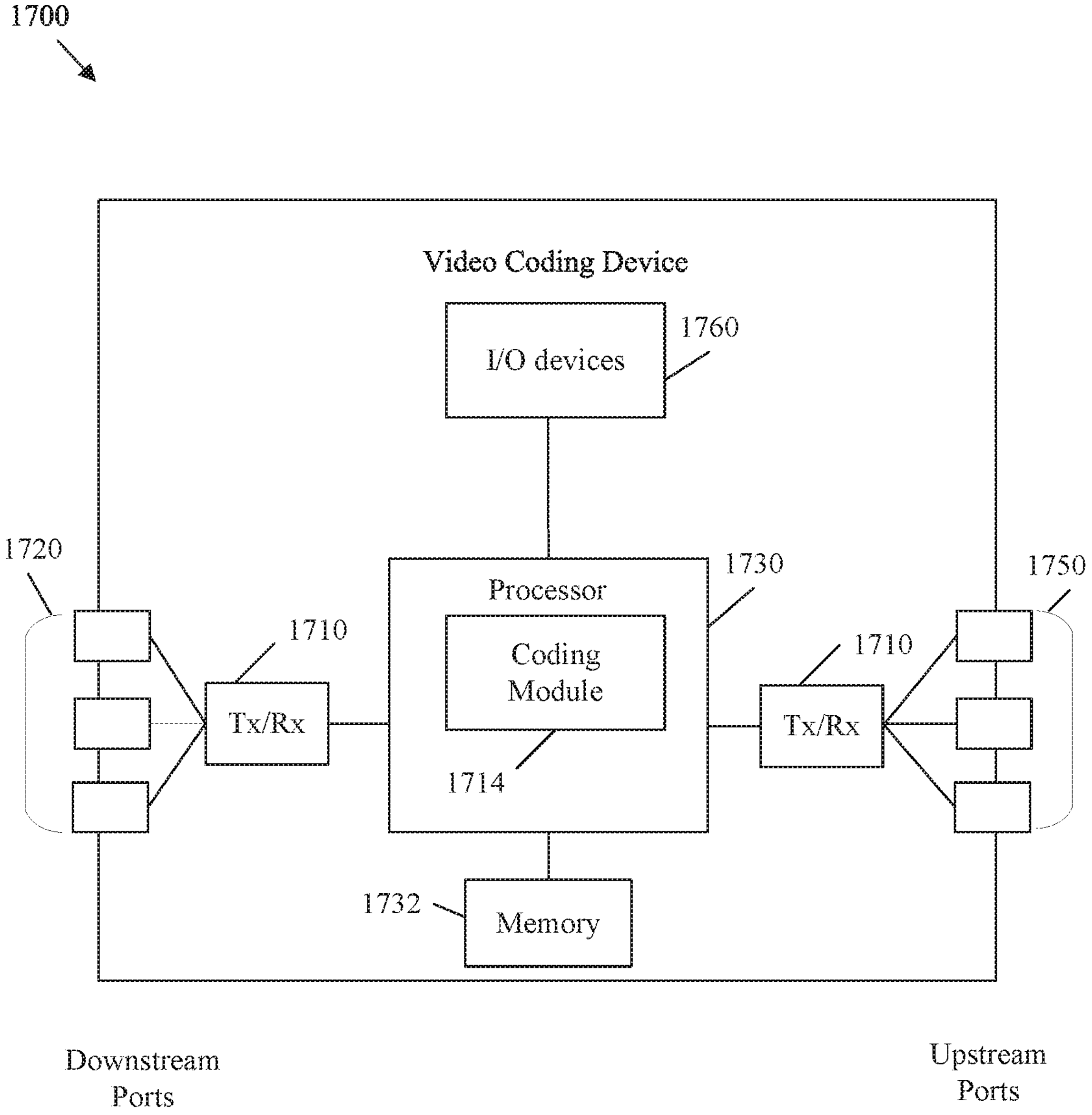
1700
Video Coding Device
I/O devices
1760
1720
1710
Tx/Rx
Processor
1730
Coding Module
1714
1710
Tx/Rx
1750
1732
Memory
Downstream Ports
Upstream Ports

FIG. 17

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2019/082842

A. CLASSIFICATION OF SUBJECT MATTER

H04N 19/50(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)

H04N

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

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

WPI, EPODOC, CNPAT, CNKI, IEEE: encode, decode, intra, prediction, chroma, luma, luminance, block, region, area, top, left, template, width, height, linear, model, coefficient, extend

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 9288500 B2 (TEXAS INSTRUMENTS INCORPORATED) 15 March 2016 (2016-03-15) claim 1, description, column 9 line 57 – column 11 line 53	1-31
A	US 9693070 B2 (TEXAS INSTRUMENTS INCORPORATED) 27 June 2017 (2017-06-27) the whole document	1-31
A	US 2014086502 A1 (GUO, MEI ET AL.) 27 March 2014 (2014-03-27) the whole document	1-31
A	US 2014355667 A1 (MEDIATEK SINGAPORE PTE. LTD.) 04 December 2014 (2014-12-04) the whole document	1-31
A	US 2017295365 A1 (TEXAS INSTRUMENTS INCORPORATED) 12 October 2017 (2017-10-12) the whole document	1-31

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search

04 July 2019

Date of mailing of the international search report

18 July 2019

Name and mailing address of the ISA/CN

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

Facsimile No. **(86-10)62019451**

Authorized officer

DAI,Lei

Telephone No. **86-(10)-53961402**

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2019/082842

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
US 9288500 B2	15 March 2016	US 2017359597 A1	14 December 2017
		US 2012287995 A1	15 November 2012
		US 9723327 B2	01 August 2017
		US 2016198190 A1	07 July 2016
US 9693070 B2	27 June 2017	US 2012328013 A1	27 December 2012
		US 2017295365 A1	12 October 2017
US 2014086502 A1	27 March 2014	US 9565428 B2	07 February 2017
		WO 2012175003 A1	27 December 2012
		CN 103688533 A	26 March 2014
		CN 103688533 B	23 December 2015
US 2014355667 A1	04 December 2014	AU 2012364505 A1	10 July 2014
		EP 2801197 A1	12 November 2014
		AU 2012364505 B2	29 October 2015
		US 9560359 B2	31 January 2017
		KR 101708985 B1	21 February 2017
		EP 2801197 B1	15 November 2017
		WO 2013102418 A1	11 July 2013
		WO 2013102293 A1	11 July 2013
		KR 20140110015 A	16 September 2014
		CN 103918269 A	09 July 2014
		CN 103918269 B	01 August 2017
US 2017295365 A1	12 October 2017	US 9693070 B2	27 June 2017
		US 2012328013 A1	27 December 2012