



(51) International Patent Classification:

H04N 19/537 (2014.01)

H04N 19/55 (2014.01)

H04N 19/573 (2014.01)

(21) International Application Number:

PCT/IB2019/055566

(22) International Filing Date:

01 July 2019 (01.07.2019)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

PCT/CN2018/093943

01 July 2018 (01.07.2018)

CN

(71) Applicants: BEIJING BYTEDANCE NETWORK

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

(72) Inventors: ZHANG, Kai; 12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US). ZHANG, Li; 12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US). LIU, Hongbin; Jinritoutiao Post Office, China Satellite Communications Tower, No.63, Zhichun Road, Haidian District, Beijing 100080 (CN). WANG, Yue; Jinritoutiao Post Office, China Satellite Communications Tower, No.63, Zhichun Road, Haidian District, Beijing 100080 (CN).

(74) Agent: LIU, SHEN & ASSOCIATES; 10th Floor, Building 1, 10 Caihefang Road, Haidian District, Beijing 100080 (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,

(54) Title: IMPROVEMENT OF INTERWEAVED PREDICTION

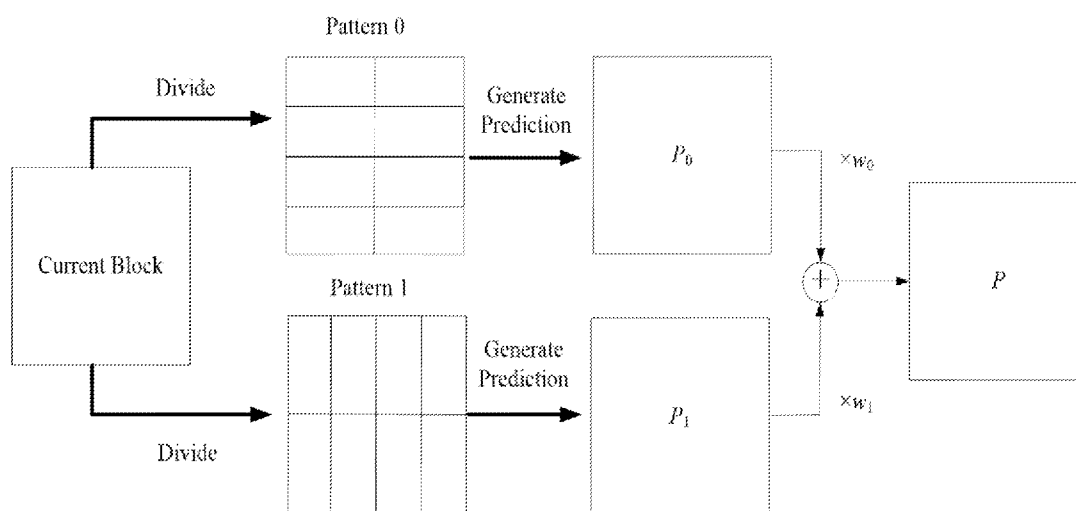


FIG. 13

(57) Abstract: Devices, systems and methods for video processing are described. In a representative aspect, a video processing method is provided to include determining, based on a component type of a current video block, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block; and performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating a predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.



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).

Published:

- *with international search report (Art. 21(3))*
- *in black and white; the international application as filed contained color or greyscale and is available for download from PATENTSCOPE*

IMPROVEMENT OF INTERWEAVED PREDICTION

CROSS-REFERENCE TO RELATED APPLICATION

[0001] Under the applicable patent law and/or rules pursuant to the Paris Convention, this application is made to timely claim the priority to and benefits of International Patent Application No. PCT/CN2018/093943, filed on July 1, 2018. For all purposes under the U.S. law, the entire disclosure of International Patent Application No. PCT/CN2018/093943 is incorporated by reference as part of the disclosure of this application.

TECHNICAL FIELD

[0002] This patent document relates to video coding and decoding techniques, devices and systems.

BACKGROUND

[0003] In spite of the advances in video compression, digital video still accounts for the largest bandwidth use on the internet and other digital communication networks. As the number of connected user devices capable of receiving and displaying video increases, it is expected that the bandwidth demand for digital video usage will continue to grow.

SUMMARY

[0004] This document discloses techniques that can be used in video coding and decoding embodiments to improve performance of sub-block based coding, and in particular, when using affine motion coding mode.

[0005] In one example aspect, a video processing method is provided to include determining, based on a component type of a current video block, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block; and performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, wherein the applying the interweaved prediction includes dividing a portion of the current

video block into at least one sub-block using more than one dividing patterns and generating a predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[0006] In another example aspect, a video processing method is provided to include

5 determining, based on a prediction direction of a current video block, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block; and performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, wherein the applying the interweaved prediction includes
10 dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[0007] In another example aspect, a video processing method is provided to include

15 determining, based on a low delay mode of a current picture, whether an interweaved prediction mode is applicable to a conversion between the current video block in the current picture and a bitstream representation of the current video block; and performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, and wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than
20 one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[0008] In another example aspect, a video processing method is provided to include

determining, based on a usage of a current picture including a current video block as a reference, whether an interweaved prediction mode is applicable to a conversion between the current video
25 block and a bitstream representation of the current video block; and performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, and wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block patterns using more than one dividing patterns and generating predictor for the current
30 video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[0009] In another example aspect, a video processing method is provided to include performing, selectively based on a video condition, interweaved prediction-based coding of one or more components of the video from a luma component, a first chroma component, and a second chroma component of a video frame, wherein the performing of the interweaved prediction comprises determining a prediction block for a current block of a component of video by: selecting a set of pixels of the component of the video frame to form a block; partitioning the block into a first set of sub-blocks according to a first pattern; generating a first intermediate prediction block based on the first set of sub-blocks; partitioning the block into a second set of sub-blocks according to a second pattern, wherein at least one sub-block in the second set is not in the first set; generating a second intermediate prediction block based on the second set of sub-blocks; and determining a prediction block based on the first intermediate prediction block and the second intermediate prediction block.

[0010] In yet another example aspect, a video encoder device that implements a video encoding method described herein is disclosed.

[0011] In yet another representative aspect, the various techniques described herein may be embodied as a computer program product stored on a non-transitory computer readable media. The computer program product includes program code for carrying out the methods described herein.

[0012] In yet another representative aspect, a video decoder apparatus may implement a method as described herein.

[0013] The details of one or more implementations are set forth in the accompanying attachments, the drawings, and the description below. Other features will be apparent from the description and drawings, and from the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] FIG. 1 shows an example of sub-block based prediction.

[0015] FIG. 2 illustrates an example of simplified affine motion model.

[0016] FIG.3 shows an example of affine motion vector field (MVF) per sub-block.

[0017] FIG. 4 shows an example of motion vector prediction (MVP) for AF_INTER mode.

[0018] FIG. 5A and FIG. 5B depict examples of Candidates for AF_MERGE.encoding mode.

[0019] FIG. 6 shows an example process of advanced temporal motion vector predictor (ATMVP) motion prediction for a coding unit (CU).

[0020] FIG. 7 shows an example of one CU with four sub-blocks (A-D) and its neighboring blocks (a-d).

5 [0021] FIG. 8 shows an example of optical flow trajectory in video coding.

[0022] FIGS. 9A and 9B show an example of bi-directional optical (BIO) coding technique without block extension. FIG. 9A shows an example of access positions outside of the block and FIG. 9B shows an example of padding used in order to avoid extra memory access and calculation.

10 [0023] FIG. 10 shows an example of bilateral matching.

[0024] FIG. 11 shows an example of template matching.

[0025] FIG. 12 shows an example of Unilateral motion estimation (ME) in frame rate upconversion (FRUC).

[0026] FIG. 13 illustrate an example implementation of interweaved prediction.

15 [0027] FIGS. 14A to 14C show examples of partial interweaved prediction. Dotted lines represent the first dividing pattern; solid lines represent the second; bold lines represent the region to apply the interweaved prediction. Outside the region, the interweaved prediction is not applied.

[0028] FIG. 15 shows examples of weighting values in a sub-block. Exemplary weighting values {W_a, W_b} are {3, 1}, {7, 1}, {5, 3}, {13, 3}, etc.

[0029]

[0030] FIG. 16 shows an example of interweaved prediction with two dividing patterns in accordance with the disclosed technology.

25 [0031] FIG. 17A shows an example dividing pattern in which block is divided into 4×4 sub-blocks in accordance with the disclosed technology.

[0032] FIG. 17B shows an example dividing pattern in which a block is divided into 8×8 sub-blocks in accordance with the disclosed technology.

[0033] FIG. 17C shows an example dividing pattern in which a block is divided into 4×8 sub-blocks in accordance with the disclosed technology.

30 [0034] FIG. 17D shows an example dividing pattern in which a block is divided into 8×4 sub-blocks in accordance with the disclosed technology.

[0035] FIG. 17E shows an example dividing pattern in which a block is divided into non-uniform sub-blocks in accordance with the disclosed technology.

[0036] FIG. 17F shows another example dividing pattern in which a block is divided into non-uniform sub-blocks in accordance with the disclosed technology.

5 [0037] FIG. 17G shows yet another example dividing pattern in which a block is divided into non-uniform sub-blocks in accordance with the disclosed technology.

[0038] FIG. 18 is a block diagram of an example of a hardware platform for implementing a video processing method described in the present document.

10 [0039] FIG. 19 is a flowchart for an example method for video processing described in the present document.

[0040] FIG. 20 is a flowchart for another example method for video processing described in the present document.

[0041] FIG. 21 is a flowchart for another example method for video processing described in the present document.

15 [0042] FIG. 22 is a flowchart for another example method for video processing described in the present document.

[0043] FIG. 23 is a flowchart for another example method for video processing described in the present document.

20

DETAILED DESCRIPTION

[0044] Section headings are used in the present document to improve readability and do not limit the techniques and embodiments described in a section to only that section.

25 [0045] To improve compression ratio of video, researchers are continually looking for new techniques by which to encode video.

[0046] 1. Introduction

30 [0047] This invention is related to video/image coding technologies. Specifically, it is related to sub-block based prediction in video/image coding. It may be applied to the existing video coding standard like HEVC, or the standard (Versatile Video Coding) to be finalized. It may be also applicable to future video/image coding standards or video/image codec. This invention may further improve P1805026601.

[0048] Brief discussion

[0049] Sub-block based prediction is first introduced into the video coding standard by HEVC Annex I (3D-HEVC). With sub-block based prediction, a block, such as a Coding Unit (CU) or a Prediction Unit (PU), is divided into several non-overlapped sub-blocks. Different sub-block may be assigned different motion information, such as reference index or Motion Vector (MV), and Motion Compensation (MC) is performed individually for each sub-block. Fig. 1 demonstrates the concept of sub-block based prediction.

[0050] To explore the future video coding technologies beyond HEVC, Joint Video Exploration Team (JVET) was founded by VCEG and MPEG jointly in 2015. Since then, many new methods have been adopted by JVET and put into the reference software named Joint Exploration Model (JEM).

[0051] In JEM, sub-block based prediction is adopted in several coding tools, such as affine prediction, Alternative temporal motion vector prediction (ATMVP), spatial-temporal motion vector prediction (STMVP), Bi-directional Optical flow (BIO) and Frame-Rate Up Conversion (FRUC).

[0052] 2.1 Affine Prediction

[0053] In HEVC, only translation motion model is applied for motion compensation prediction (MCP). While in the real world, there are many kinds of motion, e.g. zoom in/out, rotation, perspective motions and the other irregular motions. In the JEM, a simplified affine transform motion compensation prediction is applied. As shown Fig. 2, the affine motion field of the block is described by two control point motion vectors.

[0054] The motion vector field (MVF) of a block is described by the following equation:

$$\begin{cases} v_x = ax - by + c = \frac{(v_{1x} - v_{0x})}{w}x - \frac{(v_{1y} - v_{0y})}{w}y + v_{0x} \\ v_y = bx + ay + d = \frac{(v_{1y} - v_{0y})}{w}x + \frac{(v_{1x} - v_{0x})}{w}y + v_{0y} \end{cases} \quad (1)$$

[0055] Where (v_{0x}, v_{0y}) is motion vector of the top-left corner control point, and (v_{1x}, v_{1y}) is motion vector of the top-right corner control point.

[0056] In order to further simplify the motion compensation prediction, sub-block based affine transform prediction is applied. The sub-block size $M \times N$ is derived as in Eq. (2), where $MvPre$ is the motion vector fraction accuracy (1/16 in JEM), (v_{2x}, v_{2y}) is motion vector of the

bottom-left control point, calculated according to Equation 1.

$$\begin{cases} M = \text{clip3}(4, w, \frac{w \times MvPre}{\max(\text{abs}(v_{1x} - v_{0x}), \text{abs}(v_{1y} - v_{0y}))}) \\ N = \text{clip3}(4, h, \frac{h \times MvPre}{\max(\text{abs}(v_{2x} - v_{0x}), \text{abs}(v_{2y} - v_{0y}))}) \end{cases} \quad (2)$$

[0057] After derived by Eq. (2), M and N should be adjusted downward if necessary to make it a divisor of w and h, respectively.

5 [0058] To derive motion vector of each M×N sub-block, the motion vector of the center sample of each sub-block, as shown in Fig. 3, is calculated according to Eq. (1), and rounded to 1/16 fraction accuracy. Then the motion compensation interpolation filters are applied to generate the prediction of each sub-block with derived motion vector.

[0059] After MCP, the high accuracy motion vector of each sub-block is rounded and saved
10 as the same accuracy as the normal motion vector.

[0060] In the JEM, there are two affine motion modes: AF_INTER mode and AF_MERGE mode. For CUs with both width and height larger than 8, AF_INTER mode can be applied. An affine flag in CU level is signalled in the bitstream to indicate whether AF_INTER mode is used. In this mode, a candidate list with motion vector pair $\{(v_0, v_1) | v_0 = \{v_A, v_B, v_C\}, v_1 = \{v_D, v_E\}\}$ is
15 constructed using the neighbour blocks. As shown in Fig. 4, v_0 is selected from the motion vectors of the block A, B or C. The motion vector from the neighbour block is scaled according to the reference list and the relationship among the POC of the reference for the neighbour block, the POC of the reference for the current CU and the POC of the current CU. And the approach to select v_1 from the neighbour block D and E is similar. If the number of candidate list is smaller
20 than 2, the list is padded by the motion vector pair composed by duplicating each of the AMVP candidates. When the candidate list is larger than 2, the candidates are firstly sorted according to the consistency of the neighbouring motion vectors (similarity of the two motion vectors in a pair candidate) and only the first two candidates are kept. An RD cost check is used to determine which motion vector pair candidate is selected as the control point motion vector prediction
25 (CPMVP) of the current CU. And an index indicating the position of the CPMVP in the candidate list is signalled in the bitstream. After the CPMVP of the current affine CU is determined, affine motion estimation is applied and the control point motion vector (CPMV) is found. Then the difference of the CPMV and the CPMVP is signalled in the bitstream.

[0061] When a CU is applied in AF_MERGE mode, it gets the first block coded with affine mode from the valid neighbour reconstructed blocks. And the selection order for the candidate block is from left, above, above right, left bottom to above left as shown in FIG. 5A. If the neighbour left bottom block A is coded in affine mode as shown in FIG. 5B., the motion

5 vectors v_2 , v_3 and v_4 of the top left corner, above right corner and left bottom corner of the CU which contains the block A are derived. And the motion vector v_0 of the top left corner on the current CU is calculated according to v_2 , v_3 and v_4 . Secondly, the motion vector v_1 of the above right of the current CU is calculated.

[0062] After the CPMV of the current CU v_0 and v_1 are derived, according to the simplified affine motion model Eq (1), the MVF of the current CU is generated. In order to identify whether the current CU is coded with AF_MERGE mode, an affine flag is signalled in the bitstream when there is at least one neighbour block is coded in affine mode.

[0063] 2.2 ATMVP

[0064] In the alternative temporal motion vector prediction (ATMVP) method, the motion vectors temporal motion vector prediction (TMVP) is modified by fetching multiple sets of motion information (including motion vectors and reference indices) from blocks smaller than the current CU. As shown in Fig. 6, the sub-CUs are square $N \times N$ blocks (N is set to 4 by default).

[0065] ATMVP predicts the motion vectors of the sub-CUs within a CU in two steps. The first step is to identify the corresponding block in a reference picture with a so-called temporal vector. The reference picture is called the motion source picture. The second step is to split the current CU into sub-CUs and obtain the motion vectors as well as the reference indices of each sub-CU from the block corresponding to each sub-CU, as shown in Fig. 6.

[0066] In the first step, a reference picture and the corresponding block is determined by the motion information of the spatial neighbouring blocks of the current CU. To avoid the repetitive scanning process of neighbouring blocks, the first merge candidate in the merge candidate list of the current CU is used. The first available motion vector as well as its associated reference index are set to be the **temporal vector** and the index to the motion source picture. This way, in ATMVP, the corresponding block may be more accurately identified, compared with TMVP, wherein the corresponding block (sometimes called collocated block) is always in a bottom-right or center position relative to the current CU.

[0067] In the second step, a corresponding block of the sub-CU is identified by the temporal vector in the motion source picture, by adding to the coordinate of the current CU the temporal vector. For each sub-CU, the motion information of its corresponding block (the smallest motion grid that covers the center sample) is used to derive the motion information for the sub-CU. After the motion information of a corresponding $N \times N$ block is identified, it is converted to the motion vectors and reference indices of the current sub-CU, in the same way as TMVP of HEVC, wherein motion scaling and other procedures apply. For example, the decoder checks whether the low-delay condition (i.e. the POCs of all reference pictures of the current picture are smaller than the POC of the current picture) is fulfilled and possibly uses motion vector MV_x (the motion vector corresponding to reference picture list X) to predict motion vector MV_y (with X being equal to 0 or 1 and Y being equal to 1-X) for each sub-CU.

[0068] 3. STMVP

[0069] In this method, the motion vectors of the sub-CUs are derived recursively, following raster scan order. Fig. 7 illustrates this concept. Let us consider an 8×8 CU which contains four 4×4 sub-CUs A, B, C, and D. The neighbouring 4×4 blocks in the current frame are labelled as a, b, c, and d.

[0070] The motion derivation for sub-CU A starts by identifying its two spatial neighbours. The first neighbour is the $N \times N$ block above sub-CU A (block c). If this block c is not available or is intra coded the other $N \times N$ blocks above sub-CU A are checked (from left to right, starting at block c). The second neighbour is a block to the left of the sub-CU A (block b). If block b is not available or is intra coded other blocks to the left of sub-CU A are checked (from top to bottom, starting at block b). The motion information obtained from the neighbouring blocks for each list is scaled to the first reference frame for a given list. Next, temporal motion vector predictor (TMVP) of sub-block A is derived by following the same procedure of TMVP derivation as specified in HEVC. The motion information of the collocated block at location D is fetched and scaled accordingly. Finally, after retrieving and scaling the motion information, all available motion vectors (up to 3) are averaged separately for each reference list. The averaged motion vector is assigned as the motion vector of the current sub-CU.

[0071] 4. BIO

[0072] Bi-directional Optical flow (BIO) is sample-wise motion refinement which is performed on top of block-wise motion compensation for bi-prediction. The sample-level motion

refinement doesn't use signalling.

[0073] Let $I^{(k)}$ be the luma value from reference k ($k=0, 1$) after block motion compensation, and $\partial I^{(k)}/\partial x$, $\partial I^{(k)}/\partial y$ are horizontal and vertical components of the $I^{(k)}$ gradient, respectively. Assuming the optical flow is valid, the motion vector field (v_x, v_y) is given by an equation

$$\partial I^{(k)}/\partial t + v_x \partial I^{(k)}/\partial x + v_y \partial I^{(k)}/\partial y = 0. \quad (3)$$

[0074] Combining this optical flow equation with Hermite interpolation for the motion trajectory of each sample results in a unique third-order polynomial that matches both the function values $I^{(k)}$ and derivatives $\partial I^{(k)}/\partial x$, $\partial I^{(k)}/\partial y$ at the ends. The value of this polynomial at $t=0$ is the BIO prediction:

$$pred_{BIO} = 1/2 \cdot (I^{(0)} + I^{(1)} + v_x/2 \cdot (\tau_1 \partial I^{(1)}/\partial x - \tau_0 \partial I^{(0)}/\partial x) + v_y/2 \cdot (\tau_1 \partial I^{(1)}/\partial y - \tau_0 \partial I^{(0)}/\partial y)) \quad (4)$$

[0075] Here, τ_0 and τ_1 denote the distances to the reference frames as shown on a Fig. 8.

Distances τ_0 and τ_1 are calculated based on POC for Ref0 and Ref1: $\tau_0 = \text{POC}(\text{current}) - \text{POC}(\text{Ref0})$, $\tau_1 = \text{POC}(\text{Ref1}) - \text{POC}(\text{current})$. If both predictions come from the same time

direction (either both from the past or both from the future) then the signs are different (i.e., $\tau_0 \cdot \tau_1 < 0$). In this case, BIO is applied only if the prediction is not from the same time moment (i.e., $\tau_0 \neq \tau_1$), both referenced regions have non-zero motion ($MVx_0, MVy_0, MVx_1, MVy_1 \neq 0$) and the block motion vectors are proportional to the time distance ($MVx_0/MVx_1 = MVy_0/MVy_1 = -\tau_0/\tau_1$).

[0076] The motion vector field (v_x, v_y) is determined by minimizing the difference Δ between values in points A and B (intersection of motion trajectory and reference frame planes on Fig. 9A and 9B). Model uses only first linear term of a local Taylor expansion for Δ :

$$\Delta = (I^{(0)} - I^{(1)}_0 + v_x (\tau_1 \partial I^{(1)}/\partial x + \tau_0 \partial I^{(0)}/\partial x) + v_y (\tau_1 \partial I^{(1)}/\partial y + \tau_0 \partial I^{(0)}/\partial y)) \quad (5)$$

[0077] All values in Equation 21 depend on the sample location (i', j') , which was omitted from the notation so far. Assuming the motion is consistent in the local surrounding area, we minimize Δ inside the $(2M+1) \times (2M+1)$ square window Ω centered on the currently predicted

point (i, j) , where M is equal to 2:

$$(v_x, v_y) = \underset{v_x, v_y}{\operatorname{argmin}} \sum_{[i', j'] \in \Omega} \Delta^2[i', j'] \quad (6)$$

[0078] For this optimization problem, the JEM uses a simplified approach making first a minimization in the vertical direction and then in the horizontal direction. This results in

$$v_x = (s_1 + r) > m? \operatorname{clip3} \left(-thBIO, thBIO, -\frac{s_3}{(s_1 + r)} \right) : 0 \quad (7)$$

$$v_y = (s_5 + r) > m? \operatorname{clip3} \left(-thBIO, thBIO, -\frac{s_6 - v_x s_2 / 2}{(s_5 + r)} \right) : 0 \quad (8)$$

where,

$$\begin{aligned} s_1 &= \sum_{[i', j'] \in \Omega} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x)^2; s_3 = \sum_{[i', j'] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) \\ s_2 &= \sum_{[i', j'] \in \Omega} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y) \\ s_5 &= \sum_{[i', j'] \in \Omega} (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)^2; s_6 = \sum_{[i', j'] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y) \end{aligned} \quad (9)$$

[0079] In order to avoid division by zero or a very small value, regularization parameters r and m are introduced in Eq (7) and Eq (8).

$$r = 500 \cdot 4^{d-8} \quad (10)$$

$$m = 700 \cdot 4^{d-8} \quad (11)$$

[0080] Here d is bit depth of the video samples.

[0081] In order to keep the memory access for BIO the same as for regular bi-predictive motion compensation, all prediction and gradients values, $I^{(k)}$, $\partial I^{(k)} / \partial x$, $\partial I^{(k)} / \partial y$, are calculated only for positions inside the current block. In Eq. (9), $(2M+1) \times (2M+1)$ square window Ω centered in currently predicted point on a boundary of predicted block needs to accesses positions outside of the block (as shown in Fig. 9A). In the JEM, values of $I^{(k)}$, $\partial I^{(k)} / \partial x$, $\partial I^{(k)} / \partial y$ outside of the block are set to be equal to the nearest available value inside the block. For example, this can be implemented as padding, as shown in Fig. 9B.

[0082] With BIO, it's possible that the motion field can be refined for each sample. To reduce the computational complexity, a block-based design of BIO is used in the JEM. The motion refinement is calculated based on 4×4 block. In the block-based BIO, the values of s_n in Eq. (9) of all samples in a 4×4 block are aggregated, and then the aggregated values of s_n in are

used to derived BIO motion vectors offset for the 4×4 block. More specifically, the following formula is used for block-based BIO derivation:

$$\begin{aligned}
 s_{1,b_k} &= \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega(x,y)} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x)^2; s_{3,b_k} = \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x); \\
 s_{2,b_k} &= \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y); \\
 s_{5,b_k} &= \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega} (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)^2; s_{6,b_k} = \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)
 \end{aligned} \tag{12}$$

- 5 **[0083]** where b_k denotes the set of samples belonging to the k-th 4×4 block of the predicted block. s_n in Eq (7) and Eq (8) are replaced by $((s_{n,b_k}) \gg 4)$ to derive the associated motion vector offsets.

[0084] In some cases, MV regiment of BIO might be unreliable due to noise or irregular motion. Therefore, in BIO, the magnitude of MV regiment is clipped to a threshold value th_{BIO} .

- 10 The threshold value is determined based on whether the reference pictures of the current picture are all from one direction. If all the reference pictures of the current picture are from one direction, the value of the threshold is set to $12 \times 2^{14-d}$; otherwise, it is set to $12 \times 2^{13-d}$.

[0085] Gradients for BIO are calculated at the same time with motion compensation interpolation using operations consistent with HEVC motion compensation process (2D

- 15 separable FIR). The input for this 2D separable FIR is the same reference frame sample as for motion compensation process and fractional position $(fracX, fracY)$ according to the fractional part of block motion vector. In case of horizontal gradient $\partial I / \partial x$ signal first interpolated vertically using *BIOfilterS* corresponding to the fractional position $fracY$ with de-scaling shift $d-8$, then gradient filter *BIOfilterG* is applied in horizontal direction corresponding to the
- 20 fractional position $fracX$ with de-scaling shift by $18-d$. In case of vertical gradient $\partial I / \partial y$ first gradient filter is applied vertically using *BIOfilterG* corresponding to the fractional position $fracY$ with de-scaling shift $d-8$, then signal displacement is performed using *BIOfilterS* in horizontal direction corresponding to the fractional position $fracX$ with de-scaling shift by $18-d$. The length of interpolation filter for gradients calculation *BIOfilterG* and signal displacement
- 25 *BIOfilterF* is shorter (6-tap) in order to maintain reasonable complexity. Table shows the filters used for gradients calculation for different fractional positions of block motion vector in BIO.

Table shows the interpolation filters used for prediction signal generation in BIO.

Table 1 Filters for gradients calculation in BIO

Fractional pel position	Interpolation filter for gradient(BIOfilterG)
0	{ 8, -39, -3, 46, -17, 5}
1/16	{ 8, -32, -13, 50, -18, 5}
1/8	{ 7, -27, -20, 54, -19, 5}
3/16	{ 6, -21, -29, 57, -18, 5}
1/4	{ 4, -17, -36, 60, -15, 4}
5/16	{ 3, -9, -44, 61, -15, 4}
3/8	{ 1, -4, -48, 61, -13, 3}
7/16	{ 0, 1, -54, 60, -9, 2}
1/2	{ -1, 4, -57, 57, -4, 1}

Table 2 Interpolation filters for prediction signal generation in BIO

Fractional pel position	Interpolation filter for prediction signal(BIOfilterS)
0	{ 0, 0, 64, 0, 0, 0}
1/16	{ 1, -3, 64, 4, -2, 0}
1/8	{ 1, -6, 62, 9, -3, 1}
3/16	{ 2, -8, 60, 14, -5, 1}
1/4	{ 2, -9, 57, 19, -7, 2}
5/16	{ 3, -10, 53, 24, -8, 2}
3/8	{ 3, -11, 50, 29, -9, 2}
7/16	{ 3, -11, 44, 35, -10, 3}
1/2	{ 3, -10, 35, 44, -11, 3}

[0086] In the JEM, BIO is applied to all bi-predicted blocks when the two predictions are from different reference pictures. When LIC is enabled for a CU, BIO is disabled.

[0087] In the JEM, OBMC is applied for a block after normal MC process. To reduce the computational complexity, BIO is not applied during the OBMC process. This means that BIO is only applied in the MC process for a block when using its own MV and is not applied in the MC process when the MV of a neighboring block is used during the OBMC process.

[0088] 2.5 FRUC

[0089] A FRUC flag is signalled for a CU when its merge flag is true. When the FRUC flag is false, a merge index is signalled and the regular merge mode is used. When the FRUC flag is true, an additional FRUC mode flag is signalled to indicate which method (bilateral matching or template matching) is to be used to derive motion information for the block.

[0090] At encoder side, the decision on whether using FRUC merge mode for a CU is based on RD cost selection as done for normal merge candidate. That is the two matching modes (bilateral matching and template matching) are both checked for a CU by using RD cost selection. The one leading to the minimal cost is further compared to other CU modes. If a FRUC matching mode is the most efficient one, FRUC flag is set to true for the CU and the related matching mode is used.

[0091] Motion derivation process in FRUC merge mode has two steps. A CU-level motion search is first performed, then followed by a Sub-CU level motion refinement. At CU level, an initial motion vector is derived for the whole CU based on bilateral matching or template matching. First, a list of MV candidates is generated and the candidate which leads to the minimum matching cost is selected as the starting point for further CU level refinement. Then a local search based on bilateral matching or template matching around the starting point is performed and the MV results in the minimum matching cost is taken as the MV for the whole CU. Subsequently, the motion information is further refined at sub-CU level with the derived CU motion vectors as the starting points.

[0092] For example, the following derivation process is performed for a $W \times H$ CU motion information derivation. At the first stage, MV for the whole $W \times H$ CU is derived. At the second stage, the CU is further split into $M \times M$ sub-CUs. The value of M is calculated as in (16), D is a predefined splitting depth which is set to 3 by default in the JEM. Then the MV for each sub-CU is derived.

$$M = \max\{4, \min\{\frac{M}{2D}, \frac{N}{2D}\}\} \quad (13)$$

[0093] As shown in the Fig.10, the bilateral matching is used to derive motion information of the current CU by finding the closest match between two blocks along the motion trajectory of the current CU in two different reference pictures. Under the assumption of continuous motion trajectory, the motion vectors MV0 and MV1 pointing to the two reference blocks shall be proportional to the temporal distances, i.e., TD0 and TD1, between the current picture and the two reference pictures. As a special case, when the current picture is temporally between the two reference pictures and the temporal distance from the current picture to the two reference pictures is the same, the bilateral matching becomes mirror based bi-directional MV.

[0094] As shown in Fig. 11, template matching is used to derive motion information of the current CU by finding the closest match between a template (top and/or left neighbouring blocks of the current CU) in the current picture and a block (same size to the template) in a reference picture. Except the aforementioned FRUC merge mode, the template matching is also applied to AMVP mode. In the JEM, as done in HEVC, AMVP has two candidates. With template matching method, a new candidate is derived. If the newly derived candidate by template matching is different to the first existing AMVP candidate, it is inserted at the very beginning of the AMVP candidate list and then the list size is set to two (meaning remove the second existing AMVP candidate). When applied to AMVP mode, only CU level search is applied.

CU level MV candidate set

[0095] The MV candidate set at CU level consists of:

- (i) Original AMVP candidates if the current CU is in AMVP mode
- (ii) all merge candidates,
- (iii) several MVs in the interpolated MV field (described later).
- (iv) top and left neighbouring motion vectors

[0096] When using bilateral matching, each valid MV of a merge candidate is used as an input to generate a MV pair with the assumption of bilateral matching. For example, one valid MV of a merge candidate is (MVa, refa) at reference list A. Then the reference picture refb of its paired bilateral MV is found in the other reference list B so that refa and refb are temporally at different sides of the current picture. If such a refb is not available in reference list B, refb is determined as a reference which is different from refa and its temporal distance to the current

picture is the minimal one in list B. After refb is determined, MVb is derived by scaling MVa based on the temporal distance between the current picture and refa, refb.

[0097] Four MVs from the interpolated MV field are also added to the CU level candidate list. More specifically, the interpolated MVs at the position (0, 0), (W/2, 0), (0, H/2) and (W/2, H/2) of the current CU are added.

[0098] When FRUC is applied in AMVP mode, the original AMVP candidates are also added to CU level MV candidate set.

[0099] At the CU level, up to 15 MVs for AMVP CUs and up to 13 MVs for merge CUs are added to the candidate list.

10 **Sub-CU level MV candidate set**

[00100] The MV candidate set at sub-CU level consists of:

- (i) an MV determined from a CU-level search,
- (ii) top, left, top-left and top-right neighbouring MVs,
- (iii) scaled versions of collocated MVs from reference pictures,
- 15 (iv) up to 4 ATMVP candidates,
- (v) up to 4 STMVP candidates

[00101] The scaled MVs from reference pictures are derived as follows. All the reference pictures in both lists are traversed. The MVs at a collocated position of the sub-CU in a reference picture are scaled to the reference of the starting CU-level MV.

20 [00102] ATMVP and STMVP candidates are limited to the four first ones.

[00103] At the sub-CU level, up to 17 MVs are added to the candidate list.

Generation of interpolated MV field

[00104] Before coding a frame, interpolated motion field is generated for the whole picture based on unilateral ME. Then the motion field may be used later as CU level or sub-CU level
25 MV candidates.

[00105] First, the motion field of each reference pictures in both reference lists is traversed at 4×4 block level. For each 4×4 block, if the motion associated to the block passing through a 4×4 block in the current picture (as shown in Fig. 12) and the block has not been assigned any interpolated motion, the motion of the reference block is scaled to the current picture according
30 to the temporal distance TD0 and TD1 (the same way as that of MV scaling of TMVP in HEVC)

and the scaled motion is assigned to the block in the current frame. If no scaled MV is assigned to a 4×4 block, the block's motion is marked as unavailable in the interpolated motion field.

Interpolation and matching cost

[00106] When a motion vector points to a fractional sample position, motion compensated interpolation is needed. To reduce complexity, bi-linear interpolation instead of regular 8-tap HEVC interpolation is used for both bilateral matching and template matching.

[00107] The calculation of matching cost is a bit different at different steps. When selecting the candidate from the candidate set at the CU level, the matching cost is the absolute sum difference (SAD) of bilateral matching or template matching. After the starting MV is

determined, the matching cost C of bilateral matching at sub-CU level search is calculated as follows:

$$C = SAD + w \cdot (|MV_x - MV_x^s| + |MV_y - MV_y^s|) \quad (14)$$

[00108] where w is a weighting factor which is empirically set to 4, MV and MV^s indicate the current MV and the starting MV, respectively. SAD is still used as the matching cost of template matching at sub-CU level search.

[00109] In FRUC mode, MV is derived by using luma samples only. The derived motion will be used for both luma and chroma for MC inter prediction. After MV is decided, final MC is performed using 8-taps interpolation filter for luma and 4-taps interpolation filter for chroma.

MV refinement

[00110] MV refinement is a pattern based MV search with the criterion of bilateral matching cost or template matching cost. In the JEM, two search patterns are supported – an unrestricted center-biased diamond search (UCBDS) and an adaptive cross search for MV refinement at the CU level and sub-CU level, respectively. For both CU and sub-CU level MV refinement, the MV is directly searched at quarter luma sample MV accuracy, and this is followed by one-eighth luma sample MV refinement. The search range of MV refinement for the CU and sub-CU step are set equal to 8 luma samples.

Selection of prediction direction in template matching FRUC merge mode

[00111] In the bilateral matching merge mode, bi-prediction is always applied since the motion information of a CU is derived based on the closest match between two blocks along the

motion trajectory of the current CU in two different reference pictures. There is no such limitation for the template matching merge mode. In the template matching merge mode, the encoder can choose among uni-prediction from list0, uni-prediction from list1 or bi-prediction for a CU. The selection is based on a template matching cost as follows:

5 If $\text{costBi} \leq \text{factor} * \min(\text{cost0}, \text{cost1})$

bi-prediction is used;

Otherwise, if $cost0 \leq cost1$

uni-prediction from list0 is used;

Otherwise,

10 uni-prediction from list1 is used;

[00112] where cost0 is the SAD of list0 template matching, cost1 is the SAD of list1 template matching and costBi is the SAD of bi-prediction template matching. The value of *factor* is equal to 1.25, which means that the selection process is biased toward bi-prediction.

[00113] The inter prediction direction selection is only applied to the CU-level template
15 matching process.

[00114] Interweaved Prediction Examples

[00115] With interweaved prediction, a block is divided into sub-blocks with more than one dividing patterns. A dividing pattern is defined as the way to divide a block into sub-blocks, including the size of sub-blocks and the position of sub-blocks. For each dividing pattern, a corresponding prediction block may be generated by deriving motion information of each sub-block based on the dividing pattern. Therefore, even for one prediction direction, multiple prediction blocks may be generated by multiple dividing patterns. Alternatively, for each prediction direction, only a dividing pattern may be applied.

[00116] Suppose there are X dividing patterns, and X prediction blocks of the current block, denoted as $P_0, P_1, \dots, P_{X-1}$ are generated by sub-block based prediction with the X dividing patterns. The final prediction of the current block, denoted as P , can be generated as

$$P(x, y) = \frac{\sum_{i=0}^{X-1} w_i(x, y) \times P_i(x, y)}{\sum_{i=0}^{X-1} w_i(x, y)} \quad (15)$$

[00117] where (x, y) is the coordinate of a pixel in the block and $w_i(x, y)$ is the weighting

value of P_i . Without losing generalization, it is supposed that $\sum_{i=0}^{X-1} w_i(x, y) = (1 \ll N)$ wherein N is a non-negative value. Fig. 13 shows an example of interweaved prediction with two dividing patterns.

[00118] 3. Example problems solved by the described embodiments

5 **[00119]** There are two potential drawbacks of the affine merge MV derivation process as shown in Fig. 5.

[00120] First, the coordinate of the left-top point of a CU and the size of the CU must be stored by each 4×4 block belonging to the CU. This information is not required to be stored in HEVC

10 **[00121]** Second, the decoder must access MVs of 4×4 blocks not adjacent to the current CU. In HEVC, the decoder only needs to access MVs of 4×4 blocks adjacent to the current CU.

[00122] 4. Examples of embodiments

[00123] We propose several methods to further improve sub-block based prediction, including the interweaved prediction and the affine merge MV derivation process.

15 **[00124]** The detailed inventions below should be considered as examples to explain general concepts. These inventions should not be interpreted in a narrow way. Furthermore, these inventions can be combined in any manner. Combination between this invention and other invention is also applicable.

[00125] Usage of interweaved prediction

- 20 1. In one embodiment, whether and how to apply interweaved prediction may depend on colour components.
- a. For example, interweaved prediction is only applied on the Luma component, but not on the chroma components;
 - b. For example, the dividing patterns are different for different colour components;
 - 25 c. For example, the weighting values are different for different colour components.
2. In one embodiment, whether and how to apply interweaved prediction may depend on the inter-prediction direction and/or reference pictures same or not.
- a. For example, interweaved prediction can only be used for uni-prediction, but not for bi-prediction.
 - 30 b. For example, interweaved prediction can only be used for bi-prediction, but the two reference pictures for the two reference picture lists are the same.
 - c. In one example, interweaved prediction is disabled for Low Delayed-P (LDP) case.
 - d. In one example, interweaved prediction is also enabled when the current block is predicted from the current picture.

[00126] Partial Interweaved prediction

3. In one embodiment, interweaved prediction may be applied to partial of the whole block.
- a. The second dividing patterns may only cover part of the whole block. Samples out of the part are not affected by the interweaved prediction.
 - 5 b. The part may exclude samples located at block boundaries, e.g., the first/last n rows or the first/last m columns.
 - c. The part may exclude samples located at sub-blocks with different sizes compared to the majority of sub-block sizes within the block in the second dividing patterns.
 - 10 d. FIG. 14A and FIG. 14B show some examples of partial interweaved affine prediction. The first dividing pattern is the same as that in JEM, i.e., the top-left point of a sub-block is at $(i \times w, j \times h)$, then the MV for this sub-block (the (i, j) th sub-block) is calculated from eq.(1) with $(x, y) = (i \times w + w/2, j \times h + h/2)$. The size of a sub-block is $w \times h$, for both dividing patterns. For example, $w = h = 4$ or $w = h = 8$.
 - 15 i. In FIG. 14A, The top-left top of a sub-block block (the (i, j) th sub-block) of the second dividing pattern is $(i \times w + w/2, j \times h)$ and the MV for this sub-block is calculated from eq.(1) with $(x, y) = (i \times w + w, j \times h + h/2)$.
 - ii. In FIG. 14B, the top-left top of a sub-block of the second dividing pattern is $(i \times w, j \times h + h/2)$ and the MV for this sub-block is calculated from eq.(1) with $(x, y) = (i \times w + w/2, j \times h + h)$.
 - 20 iii. In FIG. 14B, the top-left top of a sub-block of the second dividing pattern is $(i \times w + w/2, j \times h + h/2)$ and the MV for this sub-block is calculated from eq.(1) with $(x, y) = (i \times w + w, j \times h + h)$.

[00127] FIG. 14A - 14C show examples of partial interweaved prediction. Dashed lines

represent the first dividing pattern; solid lines represent the second; bold lines represent the

25 region to apply the interweaved prediction. Outside the region, the interweaved prediction is not applied.

[00128] Weighting values in interweaved prediction

4. In one embodiment, there are two possible weighting values W_a and W_b , satisfying $W_a + W_b = 2^N$. Exemplary weighting values $\{W_a, W_b\}$ are $\{3, 1\}$, $\{7, 1\}$, $\{5, 3\}$, $\{13, 3\}$, etc.
- 30 a. If the weighting value w_1 associated with the prediction sample P_1 generated by the first dividing pattern, and the weighting value w_2 associated with the prediction sample P_2 generated by the second dividing pattern, are the same (both equal to W_a or W_b), then the final prediction P for this sample is calculated as $P = (P_1 + P_2) \gg 1$ or $P = (P_1 + P_2 + 1) \gg 1$.
 - 35 b. If the weighting value w_1 associated with the prediction sample P_1 generated by the first dividing pattern, and the weighting value w_2 associated with the prediction sample P_2 generated by the second dividing pattern, are different ($\{w_1, w_2\} = \{W_a, W_b\}$ or $\{w_1, w_2\} = \{W_b, W_a\}$), then the final prediction P for this sample is calculated as $P = (w_1 \times P_1 + w_2 \times P_2 + \text{offset}) \gg N$, where offset can be $1 \ll (N-1)$ or 0.
 - 40

c. It could be similarly extended to the case when there are more than 2 dividing patterns.

5. In one embodiment, the weighting value of a sample *A* in a sub-block is larger than that of a sample *B* in a sub-block, if sample *A* is closer to the position which the MV of the sub-block is derived for than sample *B*. Exemplary weighting values of a 4×4 sub-block, a 4×2 sub-block, a 2×4 sub-block, or a 2×2 sub-block are shown in Fig. 16.

[00129] FIG. 15 shows examples of weighting values in a sub-block. Exemplary weighting values {*W_a*, *W_b*} are {3, 1}, {7, 1}, {5, 3}, {13, 3}, etc.

[00130] Examples of interweaved prediction

[00131] FIG. 16 shows an example of interweaved prediction with two dividing patterns in accordance with the disclosed technology. A current block 1300 can be divided into multiple patterns. For example, as shown in FIG. 16, the current block is divided into both Pattern 0 (1301) and Pattern 1 (1302). Two prediction blocks, *P*₀ (1303) and *P*₁ (1304), are generated. A final prediction block *P* (1305) of the current block 1300 can be generated by computing a weighted sum of *P*₀ (1303) and *P*₁ (1304).

[00132] More generally, given *X* dividing patterns, *X* prediction blocks of the current block, denoted as *P*₀, *P*₁, ..., *P*_{*X*-1}, can be generated by sub-block based prediction with the *X* dividing patterns. The final prediction of the current block, denoted as *P*, can be generated as

$$[00133] \quad P(x, y) = \frac{\sum_{i=0}^{X-1} w_i(x, y) \times P_i(x, y)}{\sum_{i=0}^{X-1} w_i(x, y)} \quad \text{Eq. (15)}$$

[00134] Here, (*x*, *y*) is the coordinate of a pixel in the block and *w_i*(*x*, *y*) is the weighting value of *P_i*. By the way of example, and not by limitation, the weights can be expressed as:

$$[00135] \quad \sum_{i=0}^{X-1} w_i(x, y) = (1 \ll N) \quad \text{Eq. (16)}$$

[00136] *N* is a non-negative value. Alternatively, the bit-shifting operation in Eq. (16) can also be expressed as:

$$[00137] \quad \sum_{i=0}^{X-1} w_i(x, y) = 2^N \quad \text{Eq. (17)}$$

[00138] The sum of the weights being a power of two allows a more efficient computation of the weighted sum *P* by performing a bit-shifting operation instead of a floating-point division.

[00139] Dividing patterns can have different shapes, or sizes, or positions of sub-blocks. In some embodiments, a dividing pattern may include irregular sub-block sizes. FIGS. 17A-G show several examples of dividing patterns for a 16×16 block. In FIG. 17A, a block is divided into 4×4 sub-blocks in accordance with the disclosed technology. This pattern is also used in

JEM. FIG. 17B shows an example of a block being divided into 8×8 sub-blocks in accordance with the disclosed technology. FIG. 17C shows an example of the block being divided into 8×4 sub-blocks in accordance with the disclosed technology. FIG. 17D shows an example of the block being divided into 4×8 sub-blocks in accordance with the disclosed technology. In FIG. 17E, a portion of the block is divided into 4×4 sub-blocks in accordance with the disclosed technology. The pixels at block boundaries are divided in smaller sub-blocks with sizes like 2×4 , 4×2 or 2×2 . Some sub-blocks may be merged to form larger sub-blocks. FIG. 17F shows an example of adjacent sub-blocks, such as 4×4 sub-blocks and 2×4 sub-blocks, that are merged to form larger sub-blocks with sizes like 6×4 , 4×6 or 6×6 . In FIG. 17G, a portion of the block is divided into 8×8 sub-blocks. The pixels at block boundaries are divided in smaller sub-blocks with sizes like 8×4 , 4×8 or 4×4 instead.

[00140] The shapes and sizes of sub-blocks in sub-block based prediction can be determined based on the shape and/or size of the coding block and/or coded block information. For example, in some embodiments, the sub-blocks have a size of $4 \times N$ (or $8 \times N$, etc.) when the current block has a size of $M \times N$. That is, the sub-blocks have the same height as the current block. In some embodiments, the sub-blocks have a size of $M \times 4$ (or $M \times 8$, etc.) when the current block has a size of $M \times N$. That is, the sub-blocks have the same width as the current block. In some embodiments, the sub-blocks have a size of $A \times B$ with $A > B$ (e.g., 8×4) when the current block has a size of $M \times N$, where $M > N$. Alternatively, the sub-blocks can have the size of $B \times A$ (e.g., 4×8).

[00141] In some embodiments, the current block has a size of $M \times N$. The sub-blocks have a size of $A \times B$ when $M \times N \leq T$ (or $\text{Min}(M, N) \leq T$, or $\text{Max}(M, N) \leq T$, etc.), and the sub-blocks have a size of $C \times D$ when $M \times N > T$ (or $\text{Min}(M, N) > T$, or $\text{Max}(M, N) > T$, etc.), where $A \leq C$ and $B \leq D$. For example, if $M \times N \leq 256$, sub-blocks can be in a size of 4×4 . In some implementations, the sub-blocks have a size of 8×8 .

[00142] In some embodiments, whether to apply interweaved prediction can be determined based on the inter-prediction direction. For example, in some embodiments, the interweaved prediction may be applied for bi-prediction but not for uni-prediction. As another example, when multiple-hypothesis is applied, the interweaved prediction may be applied for one prediction direction when there are more than one reference blocks.

[00143] In some embodiments, how to apply interweaved prediction may also be determined

based on the inter-prediction direction. In some embodiments, a bi-predicted block with sub-block based prediction is divided into sub-blocks with two different dividing patterns for two different reference lists. For example, a bi-predicted block is divided into 4×8 sub-blocks as shown in FIG. 17D when predicted from reference list 0 (L0). The same block is divided into 8×4 sub-blocks as shown in FIG. 17C when predicted from reference list 1 (L1). The final prediction P is calculated as

$$[00144] \quad P(x, y) = \frac{w^0(x, y) \times P^0(x, y) + w^1(x, y) \times P^1(x, y)}{w^0(x, y) + w^1(x, y)} \quad \text{Eq. (18)}$$

[00145] Here, P^0 and P^1 are predictions from L0 and L1, respectively. w^0 and w^1 are weighting values for L0 and L1, respectively. As shown in Eq. (16), the weighting values can be determined as: $w^0(x, y) + w^1(x, y) = 1 \ll N$ (wherein N is non-negative integer value). Because fewer sub-blocks are used for prediction in each direction (e.g., 4×8 sub-blocks as opposed to 8×8 sub-blocks), the computation requires less bandwidth as compared to the existing sub-block based methods. By using larger sub-blocks, the prediction results are also less susceptible to noise interference.

[00146] In some embodiments, a uni-predicted block with sub-block based prediction is divided into sub-blocks with two or more different dividing patterns for the same reference list. For example, the prediction for list L ($L = 0$ or 1) P^L is calculated as

$$[00147] \quad P^L(x, y) = \frac{\sum_{i=0}^{XL-1} w_i^L(x, y) \times P_i^L(x, y)}{\sum_{i=0}^{XL-1} w_i^L(x, y)} \quad \text{Eq. (19)}$$

[00148] Here XL is the number of dividing patterns for list L . $P_i^L(x, y)$ is the prediction generated with the i^{th} dividing pattern and $w_i^L(x, y)$ is the weighting value of $P_i^L(x, y)$. For example, when XL is 2, two dividing patterns are applied for list L . In the first dividing pattern, the block is divided into 4×8 sub-blocks as shown in FIG. 17D. In the second dividing pattern, the block is divided into 8×4 sub-blocks as shown in FIG. 17D.

[00149] In some embodiments, a bi-predicted block with sub-block based prediction is considered as a combination of two uni-predicted block from L0 and L1 respectively. The prediction from each list can be derived as described in the above example. The final prediction P can be calculated as

$$[00150] \quad P(x, y) = \frac{a * \frac{\sum_{i=0}^{XL0-1} w_i^0(x, y) \times P_i^0(x, y)}{\sum_{i=0}^{XL0-1} w_i^0(x, y)} + b * \frac{\sum_{i=0}^{XL1-1} w_i^1(x, y) \times P_i^1(x, y)}{\sum_{i=0}^{XL1-1} w_i^1(x, y)}}{a + b} \quad \text{Eq. (20)}$$

[00151] Here parameters a and b are two additional weights applied to the two internal prediction blocks. In this specific example, both a and b can be set to 1. Similar to the example above, because fewer sub-blocks are used for prediction in each direction (e.g., 4×8 sub-blocks as opposed to 8×8 sub-blocks), the bandwidth usage is better than or on par with the existing sub-block based methods. At the same time, the prediction results can be improved by using larger sub-blocks.

[00152] In some embodiments, a single non-uniform pattern can be used in each uni-predicted block. For example, for each list L (e.g., L_0 or L_1), the block is divided into a different pattern (e.g., as shown in FIG. 17E or FIG. 17F). The use of a smaller number of sub-blocks reduces the demand on bandwidth. The non-uniformity of the sub-blocks also increases robustness of the prediction results.

[00153] In some embodiments, for a multiple-hypothesis coded block, there can be more than one prediction blocks generated by different dividing patterns for each prediction direction (or reference picture list). Multiple prediction blocks can be used to generate the final prediction with additional weights applied. For example, the additional weights may be set to $1/M$ wherein M is the total number of generated prediction blocks.

[00154] In some embodiments, the encoder can determine whether and how to apply the interweaved prediction. The encoder then can transmit information corresponding to the determination to the decoder at a sequence level, a picture level, a view level, a slice level, a Coding Tree Unit (CTU) (also known as a Largest Coding Unit (LCU)) level, a CU level, a PU level, a Tree Unit (TU) level, or a region level (which may include multiple CUs/PUs/TUs/LCUs). The information can be signaled in a Sequence Parameter Set (SPS), a view parameter set (VPS), a Picture Parameter Set (PPS), a Slice Header (SH), a CTU/LCU, a CU, a PU, a TU, or a first block of a region.

[00155] In some implementations, the interweaved prediction applies to existing sub-block methods like the affine prediction, ATMVP, STMVP, FRUC, or BIO. In such cases, no additional signaling cost is needed. In some implementations, new sub-block merge candidates generated by the interweaved prediction can be inserted into a merge list, e.g., interweaved prediction + ATMVP, interweaved prediction + STMVP, interweaved prediction + FRUC etc.

[00156] In some embodiments, the dividing patterns to be used by the current block can be derived based on information from spatial and/or temporal neighboring blocks. For example,

instead of relying on the encoder to signal the relevant information, both encoder and decoder can adopt a set of predetermined rules to obtain dividing patterns based on temporal adjacency (e.g., previously used dividing patterns of the same block) or spatial adjacency (e.g., dividing patterns used by neighboring blocks).

5 **[00157]** In some embodiments, the weighting values w can be fixed. For example, all dividing patterns can be weighted equally: $w_i(x, y) = 1$. In some embodiments, the weighting values can be determined based on positions of blocks as well as the dividing patterns used. For example, $w_i(x, y)$ may be different for different (x, y) . In some embodiments, the weighting values may further depend on the sub-block prediction based coding techniques (e.g., affine, or ATMVP) and/or other coded information (e.g., skip or non-skip modes, and/or MV information).

10 **[00158]** In some embodiments, the encoder can determine the weighting values, and transmit the values to the decoder at sequence level, picture level, slice level, CTU/LCU level, CU level, PU level, or region level (which may include multiple CUs/PUs/TUs/LCUs). The weighting values can be signaled in a Sequence Parameter Set (SPS), a Picture Parameter Set (PPS), a Slice Header (SH), a CTU/LCU, a CU, a PU, or a first block of a region. In some embodiments, the weighting values can be derived from the weighting values of a spatial and/or temporal neighboring block.

15 **[00159]** It is noted that the interweaved prediction techniques disclosed herein can be applied in one, some, or all coding techniques of sub-block based prediction. For example, the interweaved prediction techniques can be applied to affine prediction, while other coding techniques of sub-block based prediction (e.g., ATMVP, STMVP, FRUC or BIO) do not use the interweaved prediction. As another example, all of affine, ATMVP, and STMVP apply the interweaved prediction techniques disclosed herein.

20 **[00160]** FIG. 18 is a block diagram of an example video processing apparatus 1800. The apparatus 1800 may be used to implement one or more of the methods described herein. The apparatus 1800 may be embodied in a smartphone, tablet, computer, Internet of Things (IoT) receiver, and so on. The apparatus 1800 may include one or more processors 1802, one or more memories 1804 and video processing hardware 1806. The processor(s) 1802 may be configured to implement one or more methods described in the present document. The memory (memories) 1804 may be used for storing data and code used for implementing the methods and techniques described herein. The video processing hardware 1806 may be used to implement, in hardware

circuitry, some techniques described in the present document.

[00161] FIG. 19 shows a flowchart for an example method 1900 for video processing. The method 1900 includes, at step 1902, determining, based on a component type of a current video block, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block. The method 1900 further includes, at step 1904, performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating a predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[00162] FIG. 20 shows a flowchart for an example method 2000 for video processing. The method 2000 includes, at step 2002, determining, based on a prediction direction of a current video block, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block. The method 2000 further includes, at step 2004, performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, and wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing pattern.

[00163] FIG. 21 shows a flowchart for an example method 2100 for video processing. The method 2100 includes, at step 2102, determining, based on a low delay mode of a current picture, whether an interweaved prediction mode is applicable to a conversion between the current video block in the current picture and a bitstream representation of the current video block. The method 2100 further includes, at step 2104, performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, and, wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[00164] FIG. 22 shows a flowchart for an example method 2200 for video processing. The method 2200 includes, at step 2202, determining, based on a usage of a current picture including a current video block as a reference, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block. The method 2200 further includes, at step 2204, performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, and wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block patterns using more than one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[00165] Another example method for video processing is provided. The method includes performing (2300), selectively based on a video condition, interweaved prediction-based coding of one or more components of the video from a luma component, a first chroma component, and a second chroma component of a video frame. The performing of the interweaved prediction comprises, determining a prediction block for a current block of a component of video by: selecting (2302) a set of pixels of the component of the video frame to form a block; partitioning (2304) the block into a first set of sub-blocks according to a first pattern; generating (2306) a first intermediate prediction block based on the first set of sub-blocks; partitioning (2308) the block into a second set of sub-blocks according to a second pattern, wherein at least one sub-block in the second set is not in the first set; generating (2310) a second intermediate prediction block based on the second set of sub-blocks; and determining (2312) a prediction block based on the first intermediate prediction block and the second intermediate prediction block.

[00166] Additional features and embodiments of the above-described methods/techniques are described below using a clause-based description format.

[00167] 1. A method for video processing, comprising: determining, based on a component type of a current video block, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block; and performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating a predictor for the

current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[00168] 2. The method of clause 1, wherein the interweaved prediction is applied due to the component type being equal to luma component.

5 [00169] 3. The method of clause 1, wherein the more than one dividing patterns used for a first color component of video are different from another more than one dividing patterns used for a second color component of video.

[00170] 4. The method of clause 1, wherein the weighted average uses weights whose values are dependent on the component type.

10 [00171] 5. A method of video processing, comprising: determining, based on a prediction direction of a current video block, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block; and performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, wherein
15 the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[00172] 6. The method of clause 5, wherein the interweaved prediction is applied due to the
20 prediction direction being equal to a uni-prediction.

[00173] 7. The method of clause 5, wherein the interweaved prediction is applied due to the prediction direction being equal to a bi-direction

[00174] 8. A method of video processing, comprising: determining, based on a low delay mode of a current picture, whether an interweaved prediction mode is applicable to a conversion
25 between the current video block in the current picture and a bitstream representation of the current video block; and performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, and wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating predictor
30 for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[00175] 9. The method of clause 8, wherein the interweaved prediction is disabled for the low delay mode of the current picture.

[00176] 10. A method of video processing, comprising: determining, based on a usage of a current picture including a current video block as a reference, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block; and performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, and wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block patterns using more than one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

[00177] 11. The method of clause 10, wherein the interweaved prediction is enabled when the current video block is predicted from the current picture.

[00178] 12. The method of clause 1, 5, 8 or 10, wherein the portion of the current video block comprises less than all of the current video block.

[00179] 13. The method of any one of clause s 1-12, wherein the interweaved prediction mode is applied to a part of the current video block.

[00180] 14. The method of clause 13, wherein at least one of the dividing patterns cover only a part of the current video block.

[00181] 15. The method of clause 14, wherein the part excludes samples located at boundaries of the current video block.

[00182] 16. The method of clause 14, wherein the part excludes samples located at sub-blocks with different sizes from a majority of sub-block sizes within the current video block.

[00183] 17. The method of clause 1, 5, 8, 10 or 12, wherein the dividing the portion of the current video block further includes: partitioning the current video block into a first set of sub-blocks according to a first pattern of the dividing patterns; generating a first intermediate prediction block based on the first set of sub-blocks; partitioning the current video block into a second set of sub-blocks according to a second pattern of the dividing patterns, wherein at least one sub-block in the second set is not in the first set; and generating a second intermediate prediction block based on the second set of sub-blocks.

[00184] 18. The method of clause 17, wherein a weighting value w_1 associated with a

prediction sample generated by the first pattern is same as a weighting value w_2 associated with a prediction sample generated by the second pattern and a final prediction P is calculated as $P=(P_1+P_2)>>1$ or $P=(P_1+P_2+1)>>1$.

[00185] 19. The method of clause 17, wherein a weighting value w_1 associated with a prediction sample generated by the first pattern is different from a weighting value w_2 associated with a prediction sample generated by the second pattern and a final prediction P is calculated as is calculated as $P=(w_1 \times P_1 + w_2 \times P_2 + \text{offset}) >> N$, where offset can be $1 \ll (N-1)$ or 0.

[00186] 20. The method of clause 17, wherein a first weight W_a of the first intermediate prediction block and a second weight W_b of the second intermediate prediction block satisfy a condition that $W_a + W_b = 2^N$, where N is an integer.

[00187] 21. The method of clause 17, wherein a weighting value of a first sample in a sub-block is greater than that of a second sample in a sub-block, if the first sample is closer to a position for which a motion vector of the sub-block is derived than the second sample.

[00188] 22. A method for video processing, comprising: performing, selectively based on a video condition, interweaved prediction-based coding of one or more components of the video from a luma component, a first chroma component, and a second chroma component of a video frame, wherein the performing of the interweaved prediction comprises determining a prediction block for a current block of a component of video by: selecting a set of pixels of the component of the video frame to form a block; partitioning the block into a first set of sub-blocks according to a first pattern; generating a first intermediate prediction block based on the first set of sub-blocks; partitioning the block into a second set of sub-blocks according to a second pattern, wherein at least one sub-block in the second set is not in the first set; generating a second intermediate prediction block based on the second set of sub-blocks; and determining a prediction block based on the first intermediate prediction block and the second intermediate prediction block.

[00189] 23. The method of clause 22, wherein interweaved prediction is used for forming prediction blocks only for the luma component.

[00190] 24. The method of clause 22, wherein different first patterns or second patterns are used for partitioning different components of the video.

[00191] 25. The method of any one of clauses 21 to 24, wherein the video condition comprises a direction of prediction, and wherein the interweaved prediction is performed only for one of uni-prediction or bi-prediction and not for the other of uni-prediction and bi-prediction directions.

[00192] 26. The method of clause 22, wherein the video condition comprises using a low-delay-P encoding mode, and wherein in case that the low-delay-P mode is used, the method includes refraining from performing the interweaved prediction.

[00193] 27. The method of clause 22, wherein the video condition comprises using a current picture that includes the current block as a reference for the prediction.

[00194] 28. The method of any one of clauses 1-27, wherein the interweave-based prediction coding comprises using the first set of sub-blocks and the second set of sub-blocks from only a partial portion of the current block.

[00195] 29. The method of clause 28, wherein the smaller portion of the current block excludes samples in a boundary region of the current block.

[00196] 30. The method of clause 28, wherein interweave-based prediction coding that uses the partial portion of the current block includes performing affine prediction using the partial portion of the current block.

[00197] 31. The method of clause 22, wherein the determining the prediction block includes determining the prediction block using a weighted average of the first intermediate prediction block and the second intermediate prediction block.

[00198] 32. The method of clause 31, wherein a first weight W_a of the first intermediate prediction block and a second weight W_b of the second intermediate prediction block satisfy a condition that $W_a + W_b = 2^N$, where N is an integer.

[00199] 33. The method of clause 32, wherein $W_a = 3$ and $W_b = 1$.

[00200] 34. The method of claim 22, wherein a weighting value of a first sample in a sub-block is greater than that of a second sample in a sub-block when the first sample is closer to a position for which a motion vector of the sub-block is derived than the second sample.

[00201] 35. An apparatus comprising a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to implement the method in any one of clauses 1 to 34.

[00202] 36. A computer program product stored on a non-transitory computer readable media, the computer program product including program code for carrying out the method in any one of clauses 1 to 34.

[00203] From the foregoing, it will be appreciated that specific embodiments of the presently disclosed technology have been described herein for purposes of illustration, but that various

modifications may be made without deviating from the scope of the invention. Accordingly, the presently disclosed technology is not limited except as by the appended claims.

[00204] The disclosed and other embodiments, modules and the functional operations described in this document can be implemented in digital electronic circuitry, or in computer software, firmware, or hardware, including the structures disclosed in this document and their structural equivalents, or in combinations of one or more of them. The disclosed and other embodiments can be implemented as one or more computer program products, i.e., one or more modules of computer program instructions encoded on a computer readable medium for execution by, or to control the operation of, data processing apparatus. The computer readable medium can be a machine-readable storage device, a machine-readable storage substrate, a memory device, a composition of matter effecting a machine-readable propagated signal, or a combination of one or more them. The term “data processing apparatus” encompasses all apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. The apparatus can include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them. A propagated signal is an artificially generated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal, that is generated to encode information for transmission to suitable receiver apparatus.

[00205] A computer program (also known as a program, software, software application, script, or code) can be written in any form of programming language, including compiled or interpreted languages, and it can be deployed in any form, including as a stand-alone program or as a module, component, subroutine, or other unit suitable for use in a computing environment. A computer program does not necessarily correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data (e.g., one or more scripts stored in a markup language document), in a single file dedicated to the program in question, or in multiple coordinated files (e.g., files that store one or more modules, sub programs, or portions of code). A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

[00206] The processes and logic flows described in this document can be performed by one or more programmable processors executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by, and apparatus can also be implemented as, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit).

[00207] Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital computer. Generally, a processor will receive instructions and data from a read only memory or a random-access memory or both. The essential elements of a computer are a processor for performing instructions and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto optical disks, or optical disks. However, a computer need not have such devices. Computer readable media suitable for storing computer program instructions and data include all forms of non-volatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and CD ROM and DVD-ROM disks. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

[00208] While this patent document contains many specifics, these should not be construed as limitations on the scope of any invention or of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments of particular inventions. Certain features that are described in this patent document in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[00209] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order

shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. Moreover, the separation of various system components in the embodiments described in this patent document should not be understood as requiring such separation in all embodiments.

- 5 [00210] Only a few implementations and examples are described and other implementations, enhancements and variations can be made based on what is described and illustrated in this patent document.

CLAIMS

What is claimed is:

1. A method for video processing, comprising:
5 determining, based on a component type of a current video block, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block; and
performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block,
10 wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating a predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.
- 15 2. The method of claim 1, wherein the interweaved prediction is applied due to the component type being equal to luma component.
3. The method of claim 1, wherein the more than one dividing patterns used for a first color component of video are different from another more than one dividing patterns used for a second
20 color component of video.
4. The method of claim 1, wherein the weighted average uses weights whose values are dependent on the component type.
- 25 5. A method of video processing, comprising:
determining, based on a prediction direction of a current video block, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block; and
performing the conversion by applying the interweaved prediction mode due to the
30 determining that the interweaved prediction mode is applicable to the current video block, and

wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

5

6. The method of claim 5, wherein the interweaved prediction is applied due to the prediction direction being equal to a uni-prediction.

7. The method of claim 5, wherein the interweaved prediction is applied due to the prediction direction being equal to a bi-direction.

10

8. A method of video processing, comprising:

determining, based on a low delay mode of a current picture, whether an interweaved prediction mode is applicable to a conversion between the current video block in the current picture and a bitstream representation of the current video block; and

15

performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, and

20

wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block using more than one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

9. The method of claim 8, wherein the interweaved prediction is disabled for the low delay mode of the current picture.

25

10. A method of video processing, comprising:

determining, based on a usage of a current picture including a current video block as a reference, whether an interweaved prediction mode is applicable to a conversion between the current video block and a bitstream representation of the current video block; and

30

performing the conversion by applying the interweaved prediction mode due to the determining that the interweaved prediction mode is applicable to the current video block, and

wherein the applying the interweaved prediction includes dividing a portion of the current video block into at least one sub-block patterns using more than one dividing patterns and generating predictor for the current video block as a weighted average of predictors determined for each of the more than one dividing patterns.

5

11. The method of claim 10, wherein the interweaved prediction is enabled when the current video block is predicted from the current picture.

12. The method of claim 1, 5, 8 or 10, wherein the portion of the current video block
10 comprises less than all of the current video block.

13. The method of any one of claims 1-12, wherein the interweaved prediction mode is applied to a part of the current video block.

14. The method of claim 13, wherein at least one of the dividing patterns cover only a part of
15 the current video block.

15. The method of claim 14, wherein the part excludes samples located at boundaries of the current video block.

20

16. The method of claim 14, wherein the part excludes samples located at sub-blocks with different sizes from a majority of sub-block sizes within the current video block.

17. The method of claim 1, 5, 8, 10 or 12, wherein the dividing the portion of the current
25 video block further includes:

partitioning the current video block into a first set of sub-blocks according to a first pattern of the dividing patterns;

generating a first intermediate prediction block based on the first set of sub-blocks;

partitioning the current video block into a second set of sub-blocks according to a second
30 pattern of the dividing patterns, wherein at least one sub-block in the second set is not in the first set; and

generating a second intermediate prediction block based on the second set of sub-blocks.

18. The method of claim 17, wherein a weighting value w_1 associated with a prediction sample generated by the first pattern is same as a weighting value w_2 associated with a prediction sample generated by the second pattern and a final prediction P is calculated as $P=(P_1+P_2)>>1$ or $P=(P_1+P_2+1)>>1$.

19. The method of claim 17, wherein a weighting value w_1 associated with a prediction sample generated by the first pattern is different from a weighting value w_2 associated with a prediction sample generated by the second pattern and a final prediction P is calculated as is calculated as $P=(w_1 \times P_1 + w_2 \times P_2 + \text{offset}) >> N$, where offset can be $1 \leq (N-1)$ or 0.

20. The method of claim 17, wherein a first weight W_a of the first intermediate prediction block and a second weight W_b of the second intermediate prediction block satisfy a condition that $W_a + W_b = 2^N$, where N is an integer.

21. The method of claim 17, wherein a weighting value of a first sample in a sub-block is greater than that of a second sample in a sub-block, if the first sample is closer to a position for which a motion vector of the sub-block is derived than the second sample.

22. A method for video processing, comprising:

performing, selectively based on a video condition, interweaved prediction-based coding of one or more components of the video from a luma component, a first chroma component, and a second chroma component of a video frame, wherein the performing of the interweaved prediction comprises determining a prediction block for a current block of a component of video by:

selecting a set of pixels of the component of the video frame to form a block;

partitioning the block into a first set of sub-blocks according to a first pattern;

generating a first intermediate prediction block based on the first set of sub-blocks;

partitioning the block into a second set of sub-blocks according to a second pattern,

wherein at least one sub-block in the second set is not in the first set;

generating a second intermediate prediction block based on the second set of sub-blocks;
and

determining a prediction block based on the first intermediate prediction block and the second intermediate prediction block.

5

23. The method of claim 22, wherein interweaved prediction is used for forming prediction blocks only for the luma component.

24. The method of claim 22, wherein different first patterns or second patterns are used for
10 partitioning different components of the video.

25. The method of any one of claims 22 to 24, wherein the video condition comprises a direction of prediction, and wherein the interweaved prediction is performed only for one of uni-prediction or bi-prediction and not for the other of uni-prediction and bi-prediction directions.

15

26. The method of claim 22, wherein the video condition comprises using a low-delay-P encoding mode, and wherein in case that the low-delay-P mode is used, the method includes refraining from performing the interweaved prediction.

20 27. The method of claim 22, wherein the video condition comprises using a current picture that includes the current block as a reference for the prediction.

28. The method of any one of claims 1-27, wherein the interweave-based prediction coding comprises using the first set of sub-blocks and the second set of sub-blocks from only a partial
25 portion of the current block.

29. The method of claim 28, wherein the smaller portion of the current block excludes samples in a boundary region of the current block.

30. The method of claim 28, wherein interweave-based prediction coding that uses the partial portion of the current block includes performing affine prediction using the partial portion of the current block.

5 31. The method of claim 22, wherein the determining the prediction block includes determining the prediction block using a weighted average of the first intermediate prediction block and the second intermediate prediction block.

32. The method of claim 31, wherein a first weight W_a of the first intermediate prediction
10 block and a second weight W_b of the second intermediate prediction block satisfy a condition that $W_a + W_b = 2^N$, where N is an integer.

33. The method of claim 32, wherein $W_a = 3$ and $W_b = 1$.

15 34. The method of claim 22, wherein a weighting value of a first sample in a sub-block is greater than that of a second sample in a sub-block when the first sample is closer to a position for which a motion vector of the sub-block is derived than the second sample.

35. An apparatus comprising a processor and a non-transitory memory with instructions
20 thereon, wherein the instructions upon execution by the processor, cause the processor to implement the method recited in one or more of claims 1 to 34.

36. A computer program product stored on a non-transitory computer readable media, the
25 computer program product including program code for carrying out the method recited in one or more of claims 1 to 34.

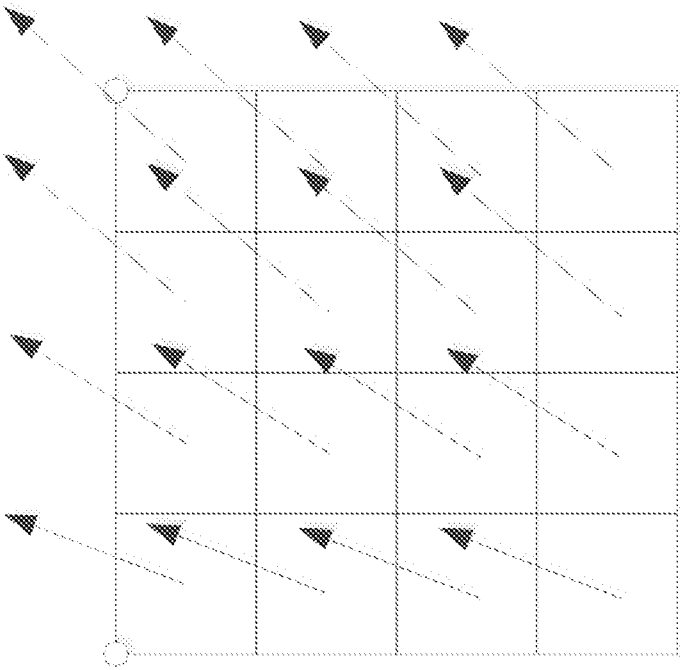


FIG. 1

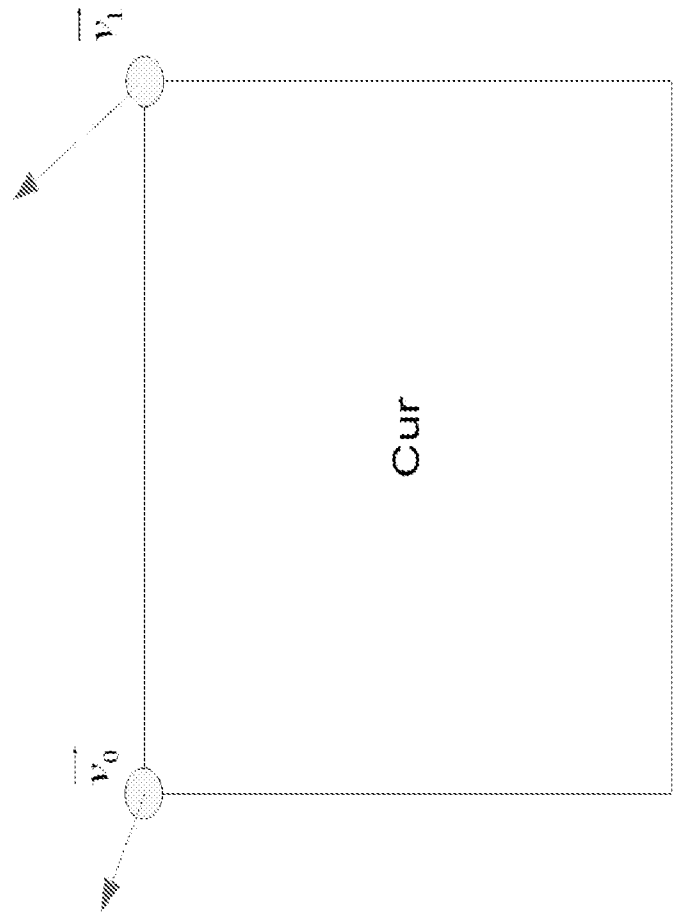


FIG. 2

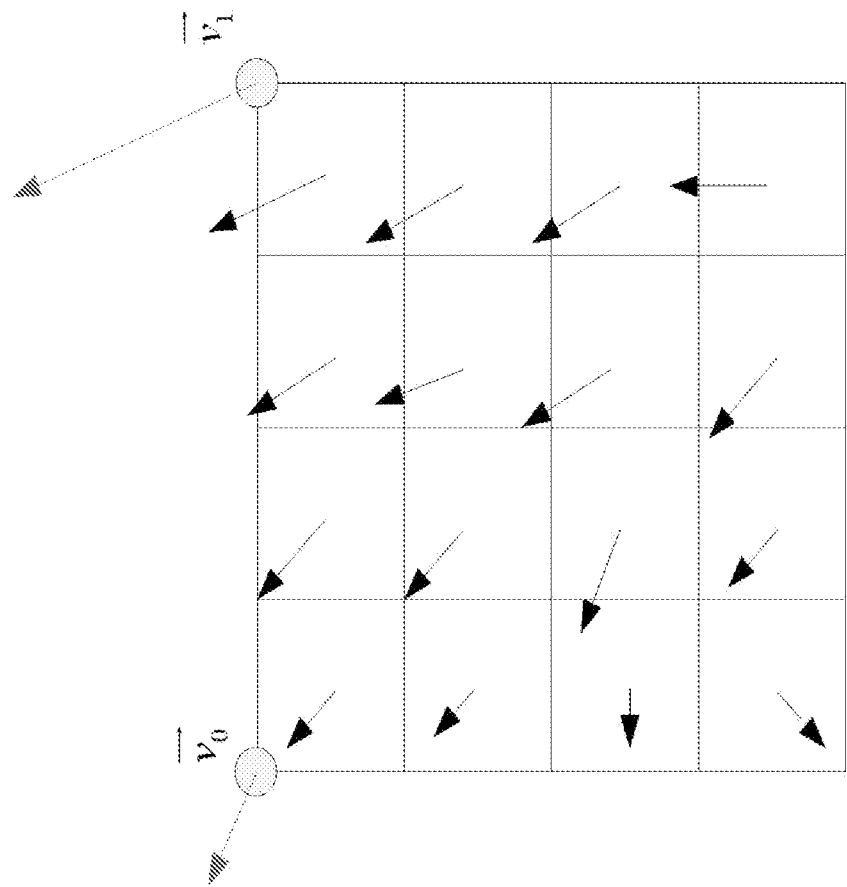


FIG. 3

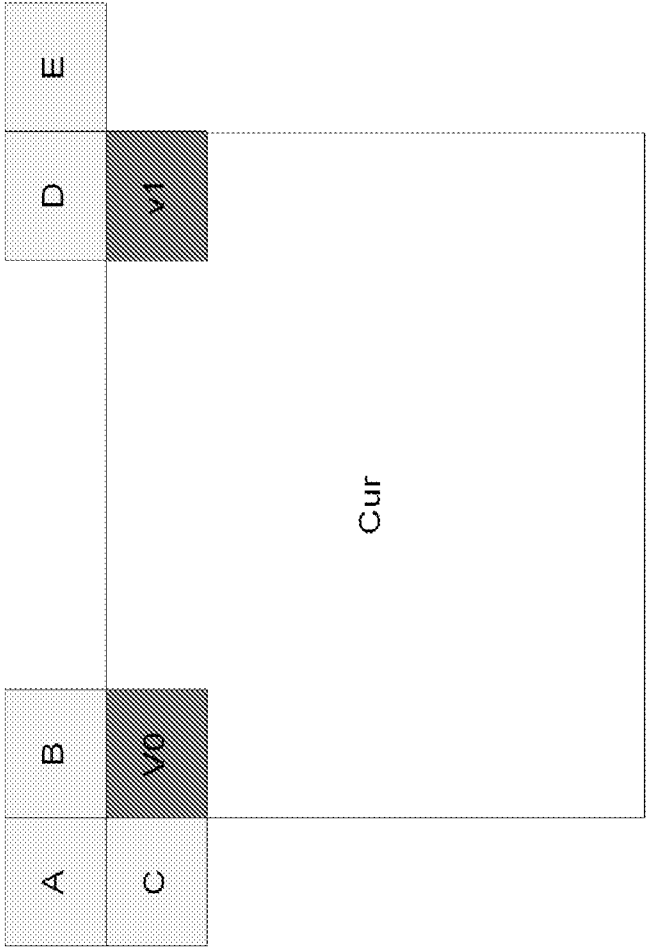


FIG. 4

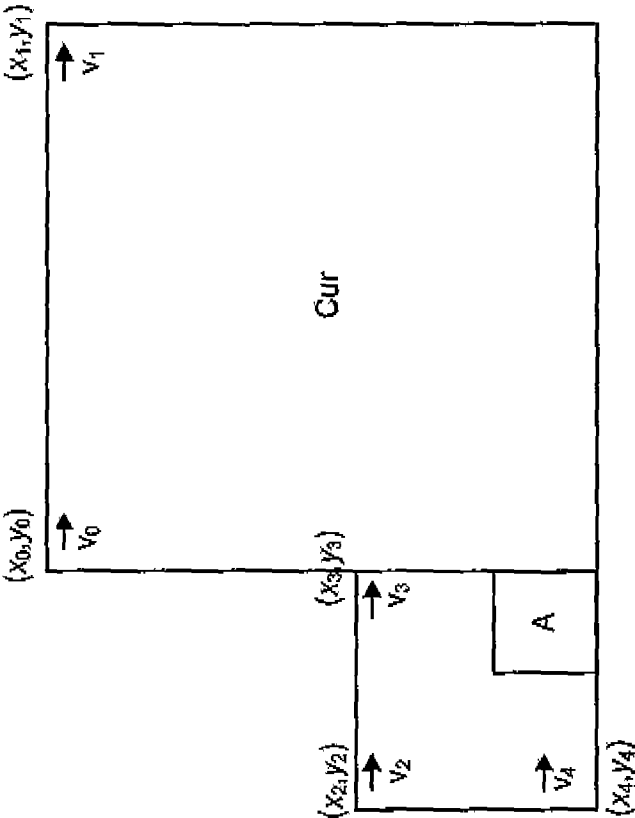


FIG. 5B

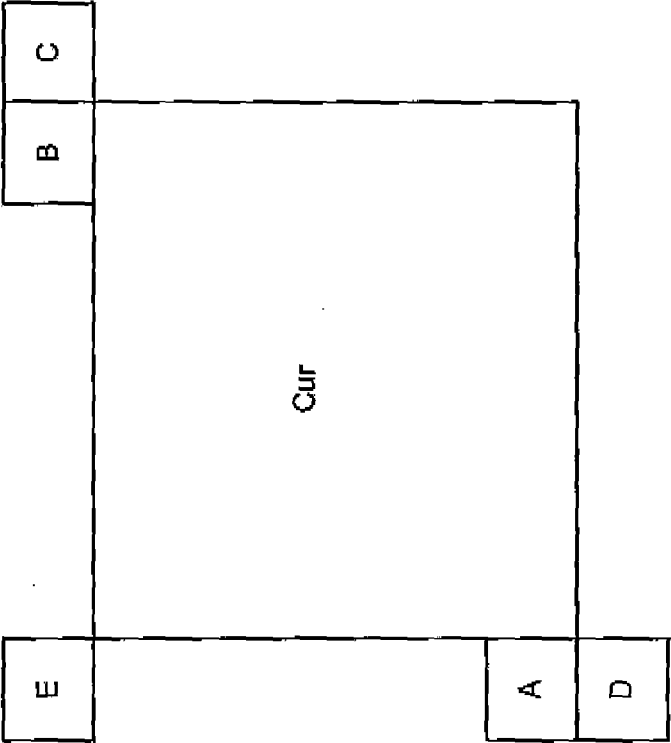


FIG. 5A

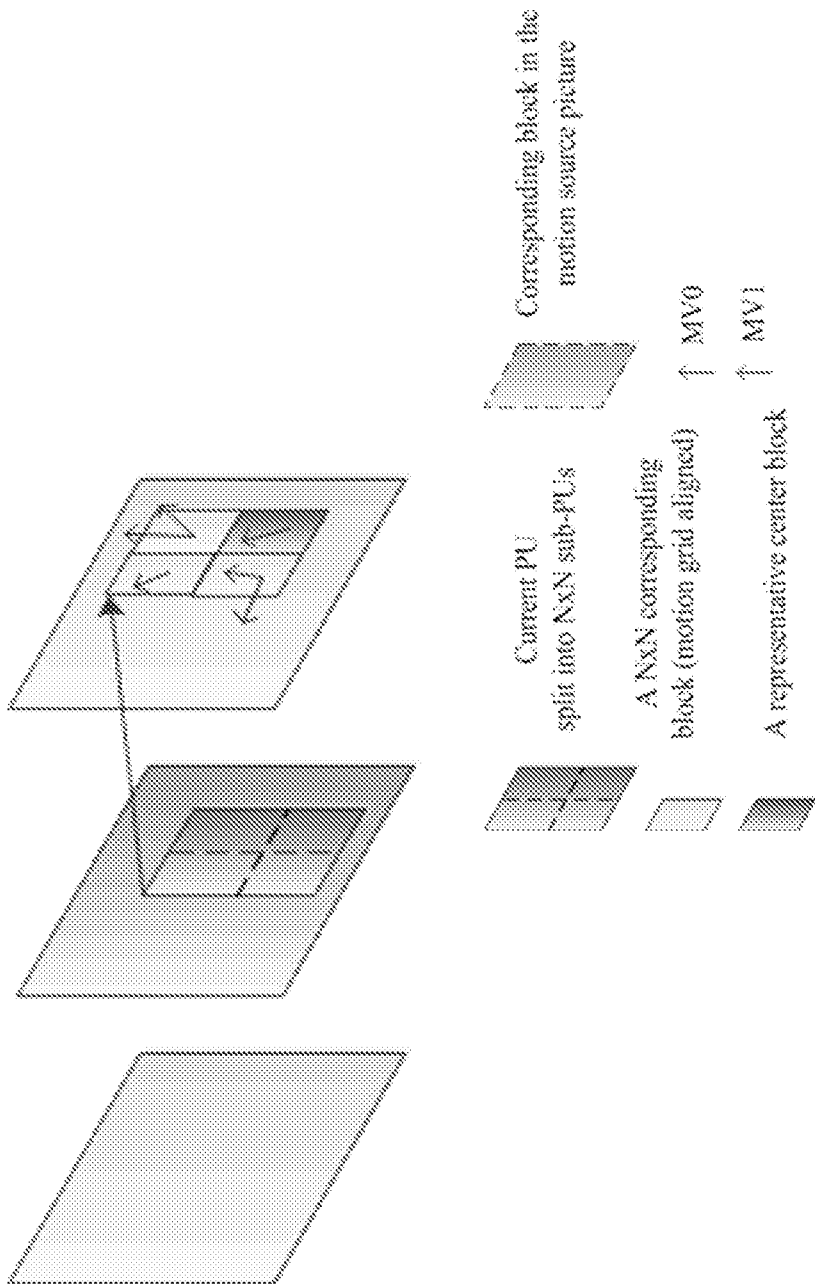


FIG. 6

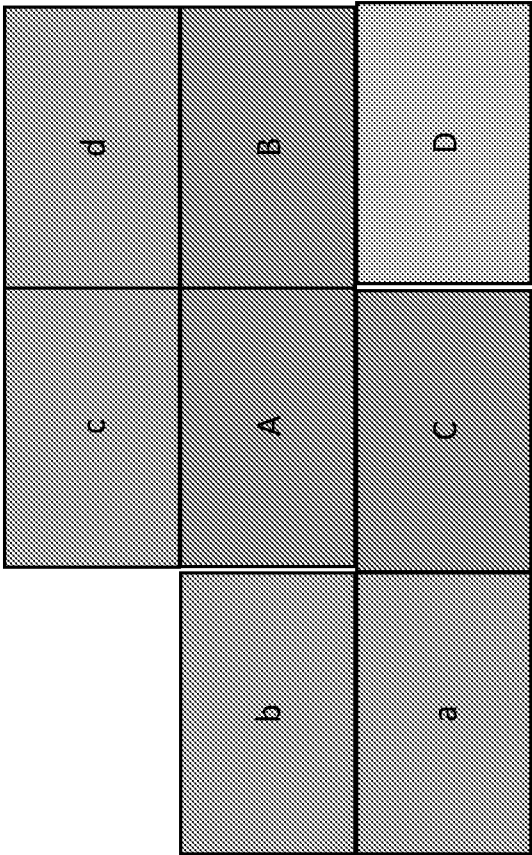


FIG. 7

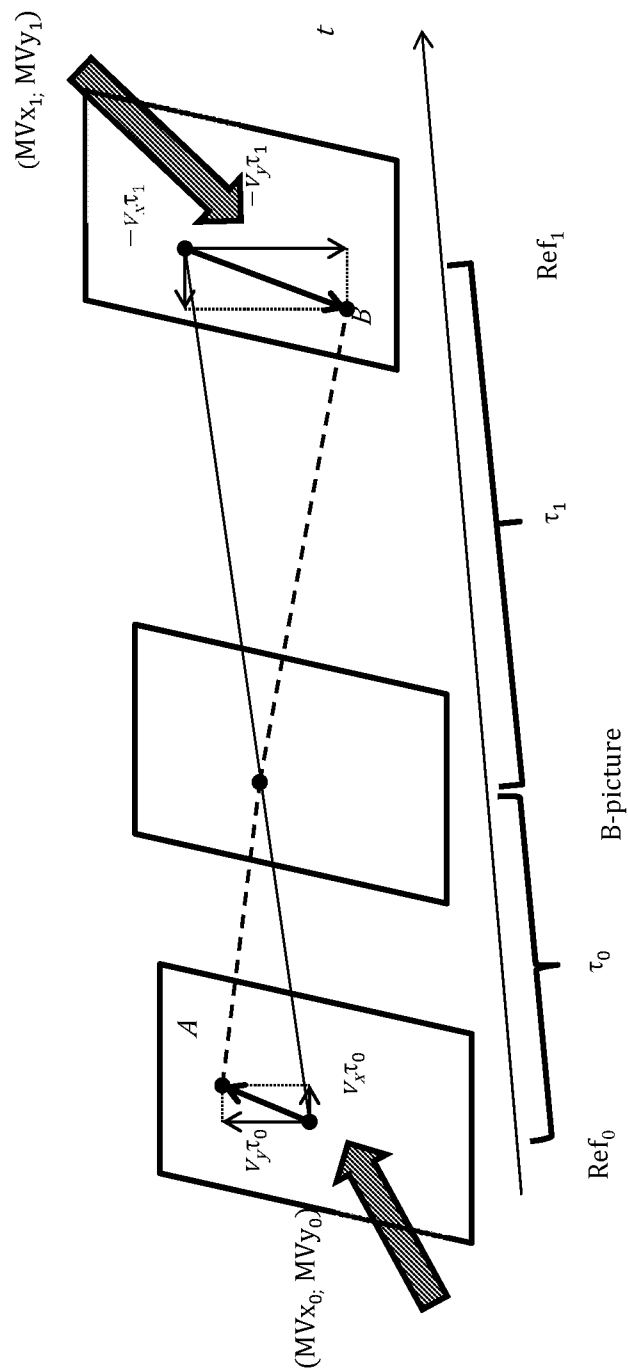


FIG. 8

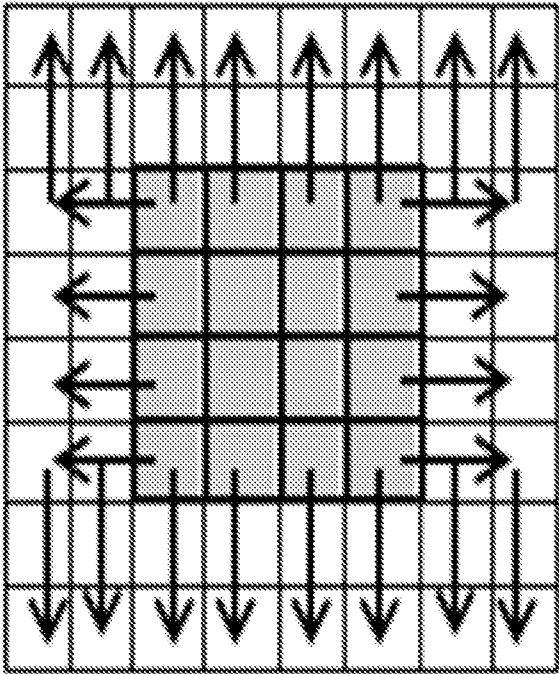


FIG. 9B

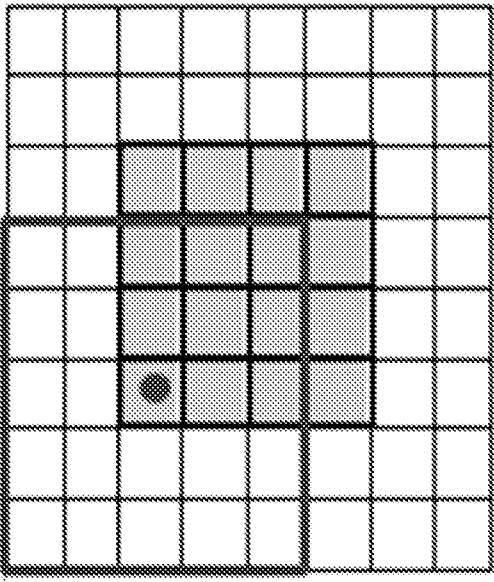


FIG. 9A

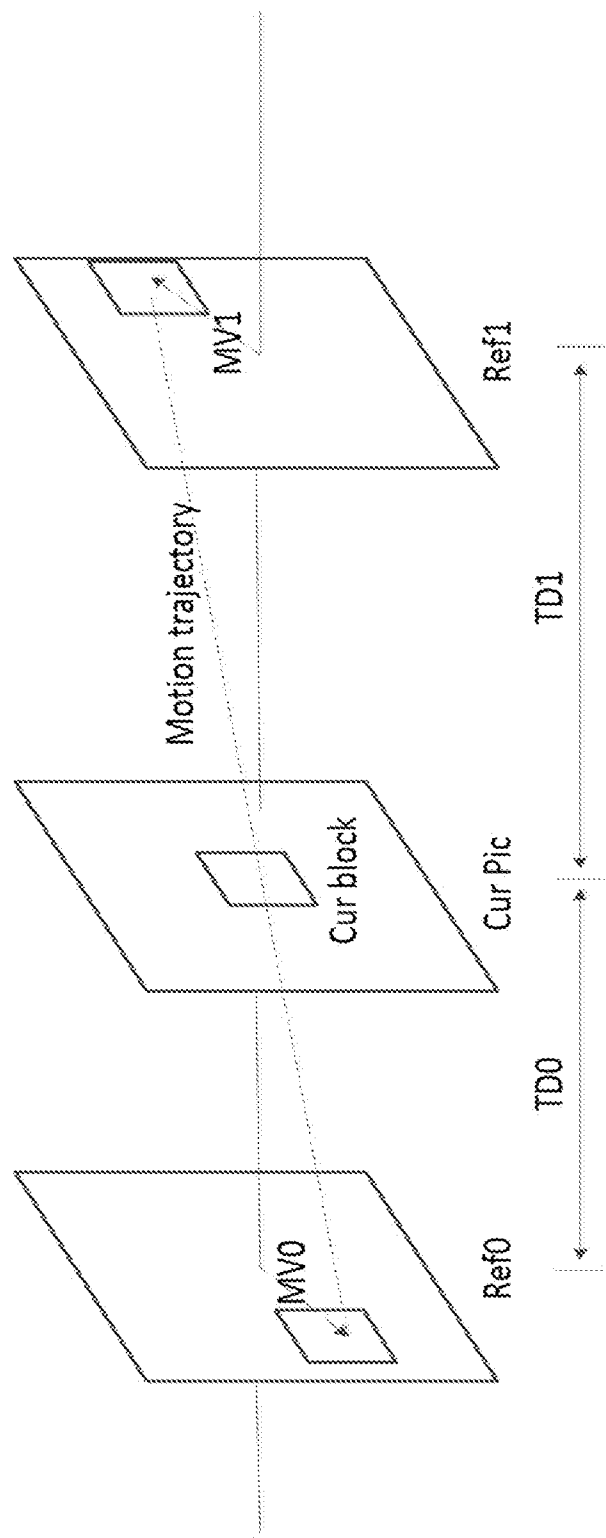


FIG. 10

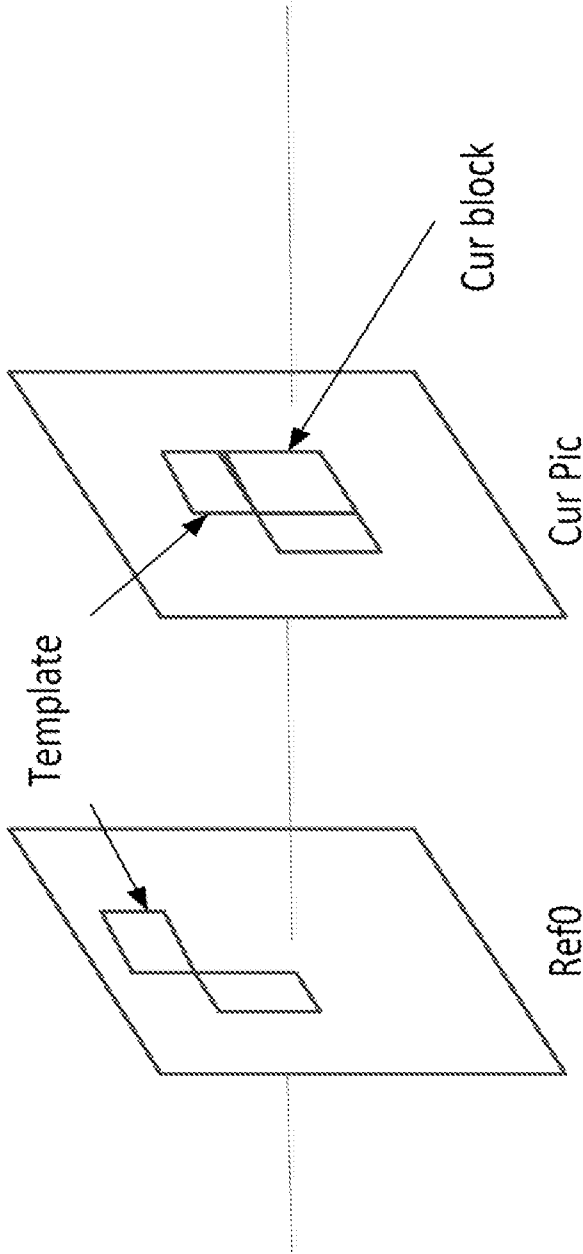


FIG. 11

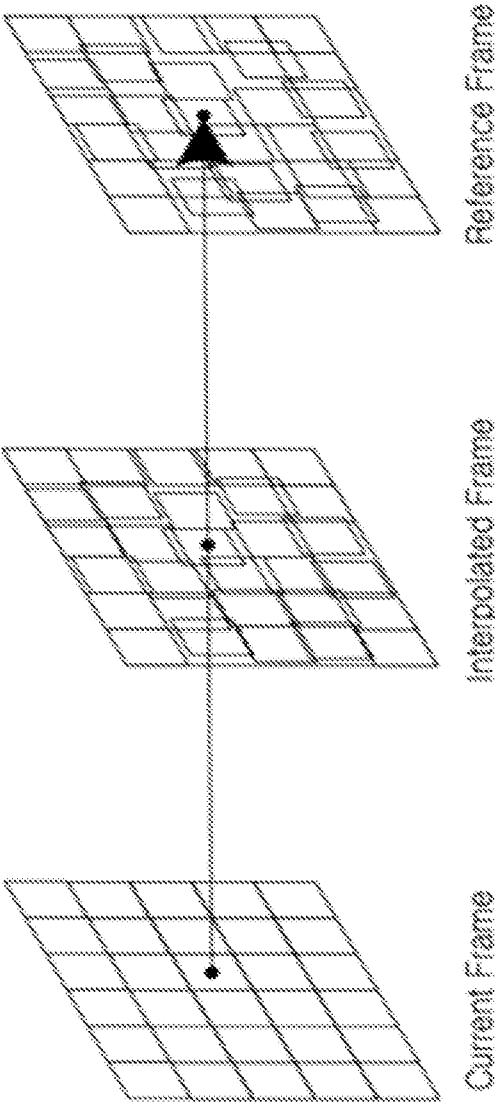


FIG. 12

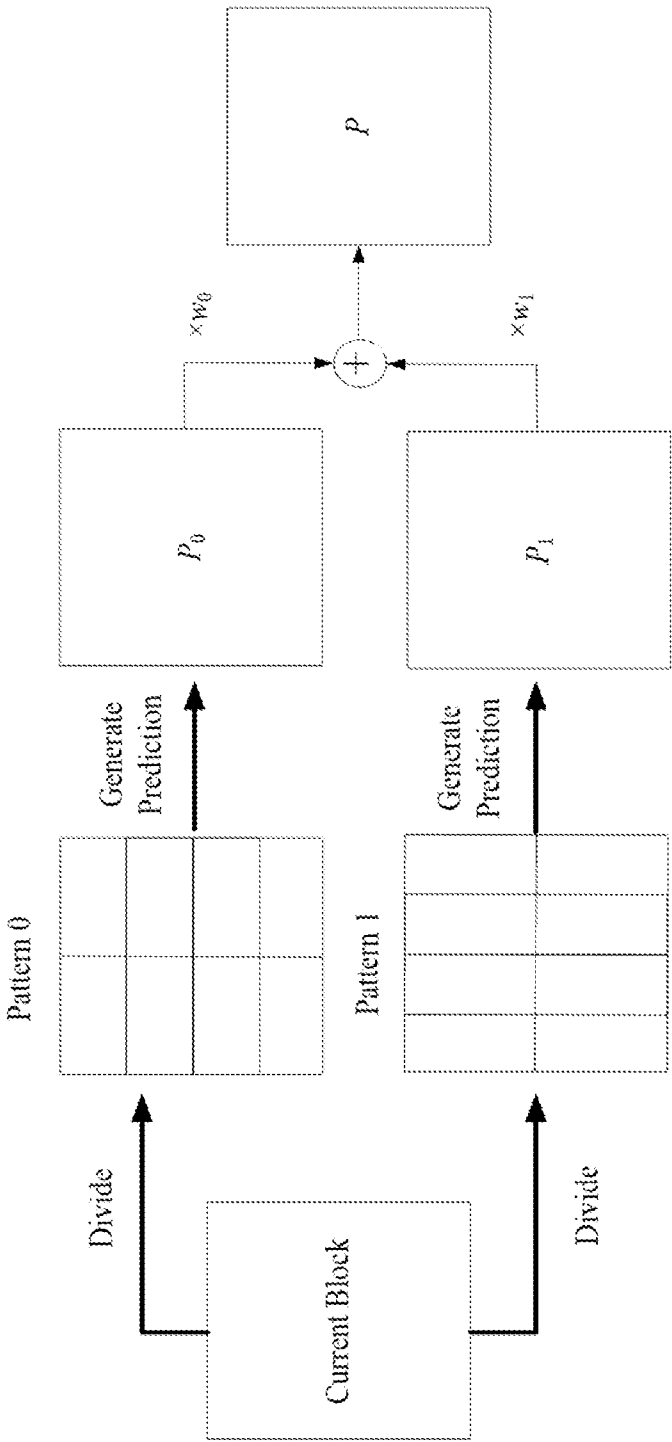


FIG. 13

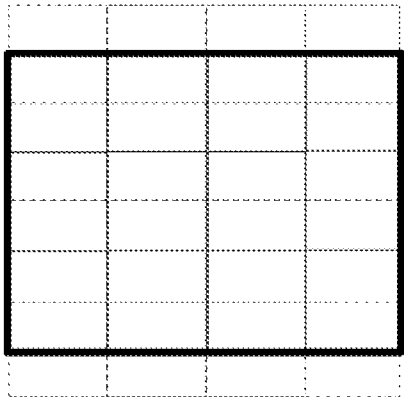


FIG. 14A

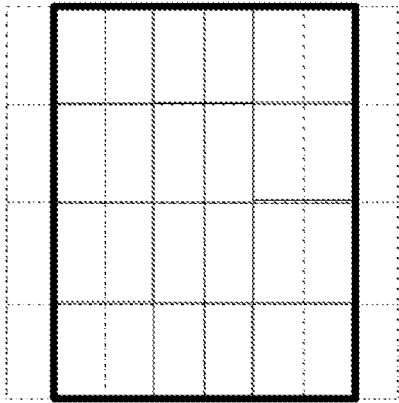


FIG. 14B

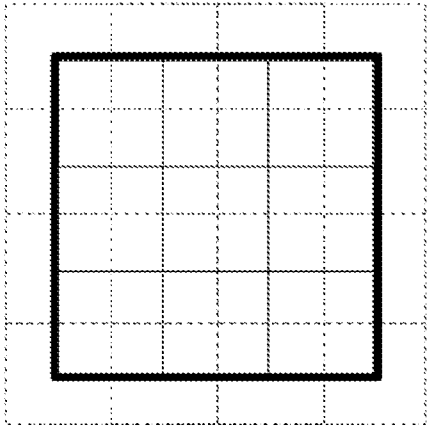


FIG. 14C

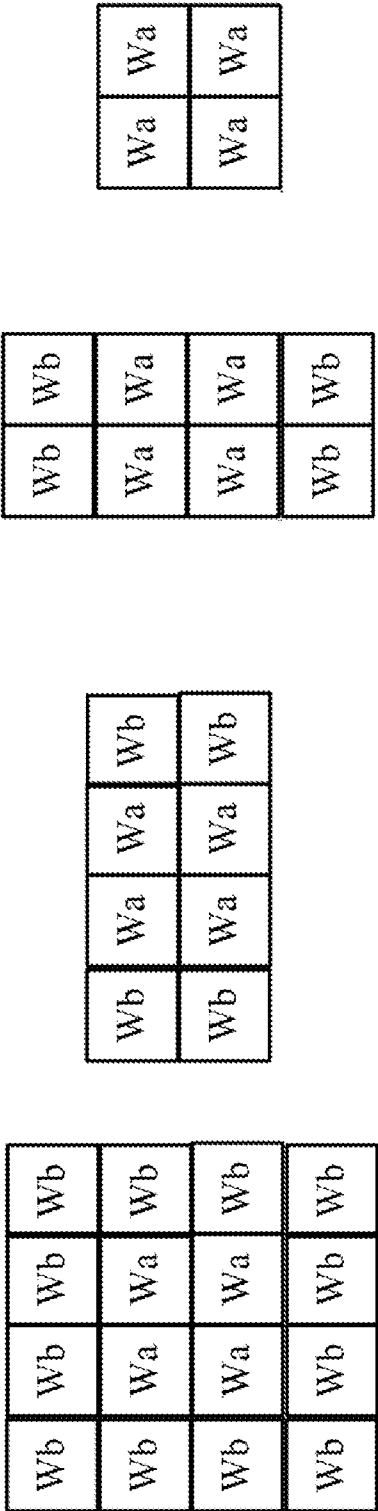


FIG. 15

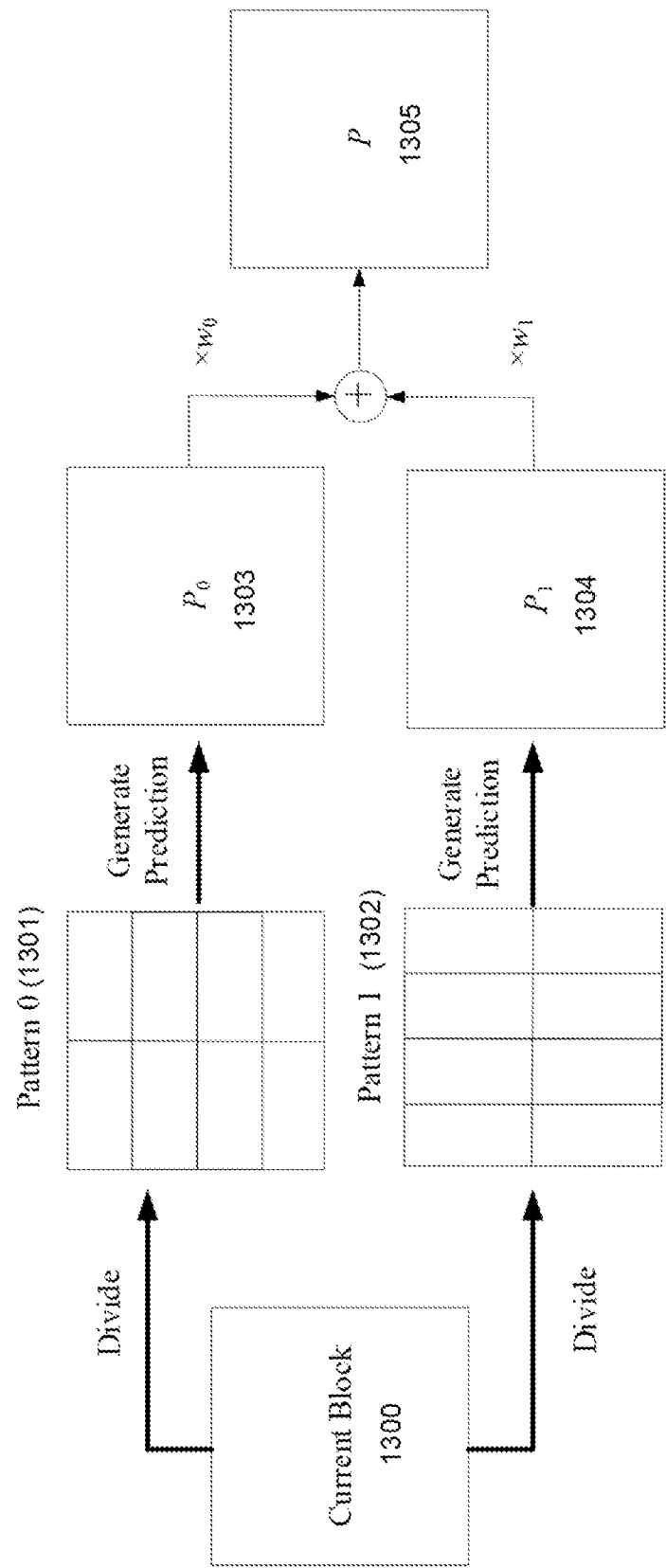


FIG. 16

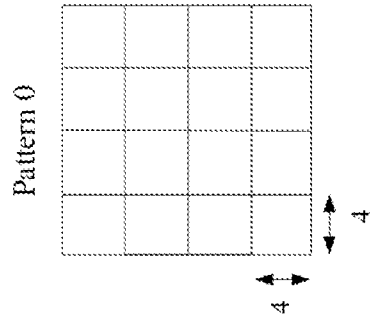


FIG. 17A

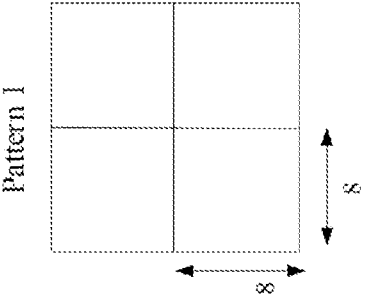


FIG. 17B

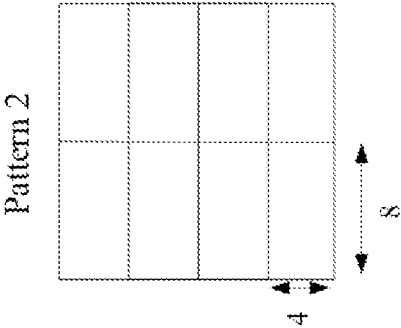


FIG. 17D

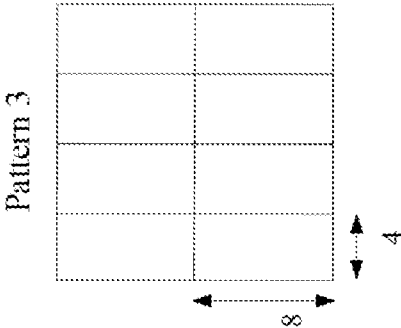


FIG. 17C

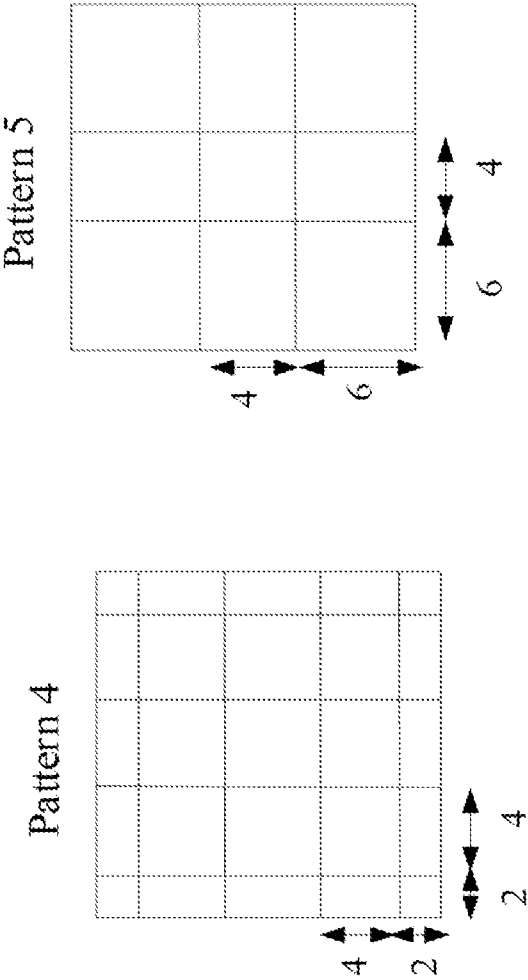
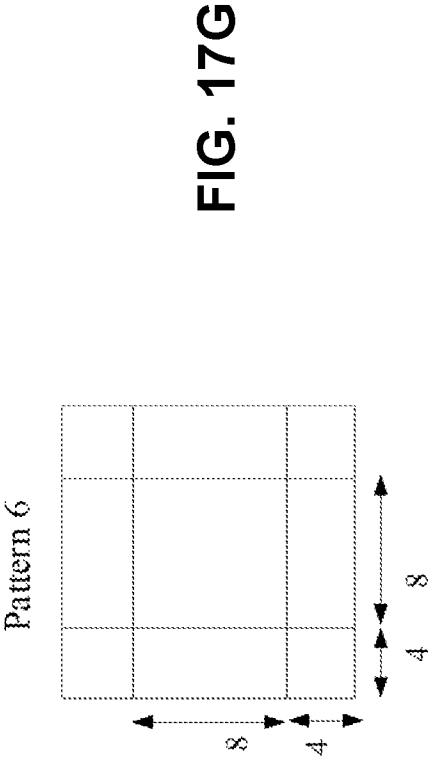


FIG. 17E

FIG. 17F



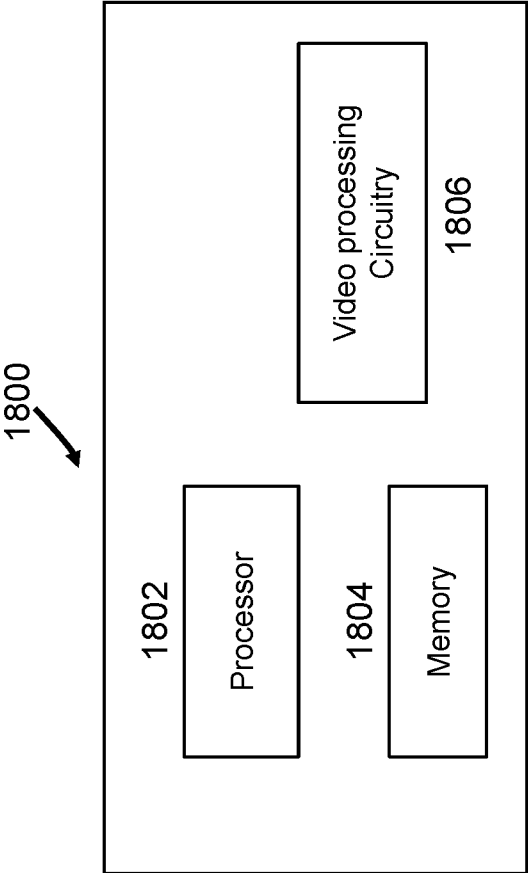


FIG. 18

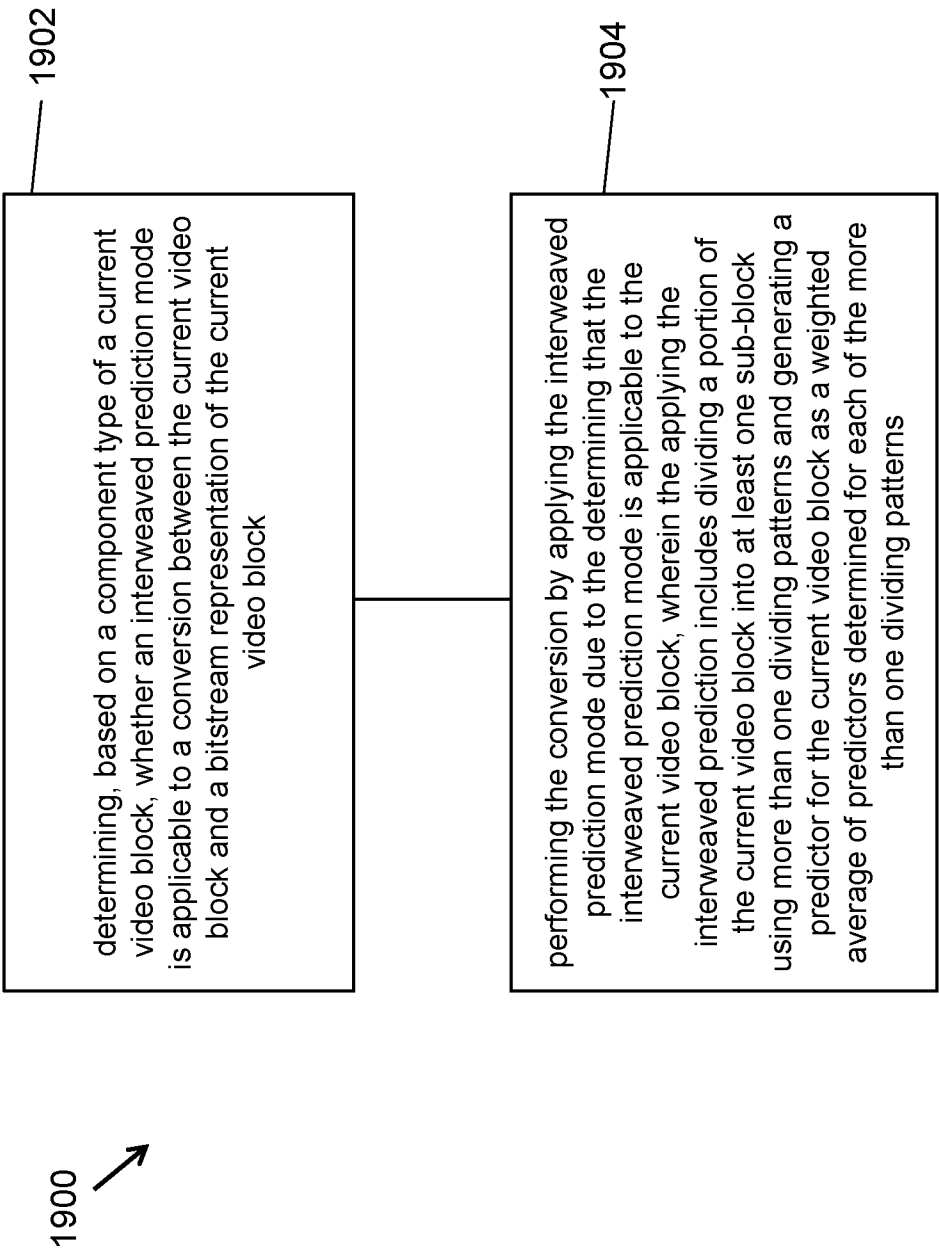


FIG. 19

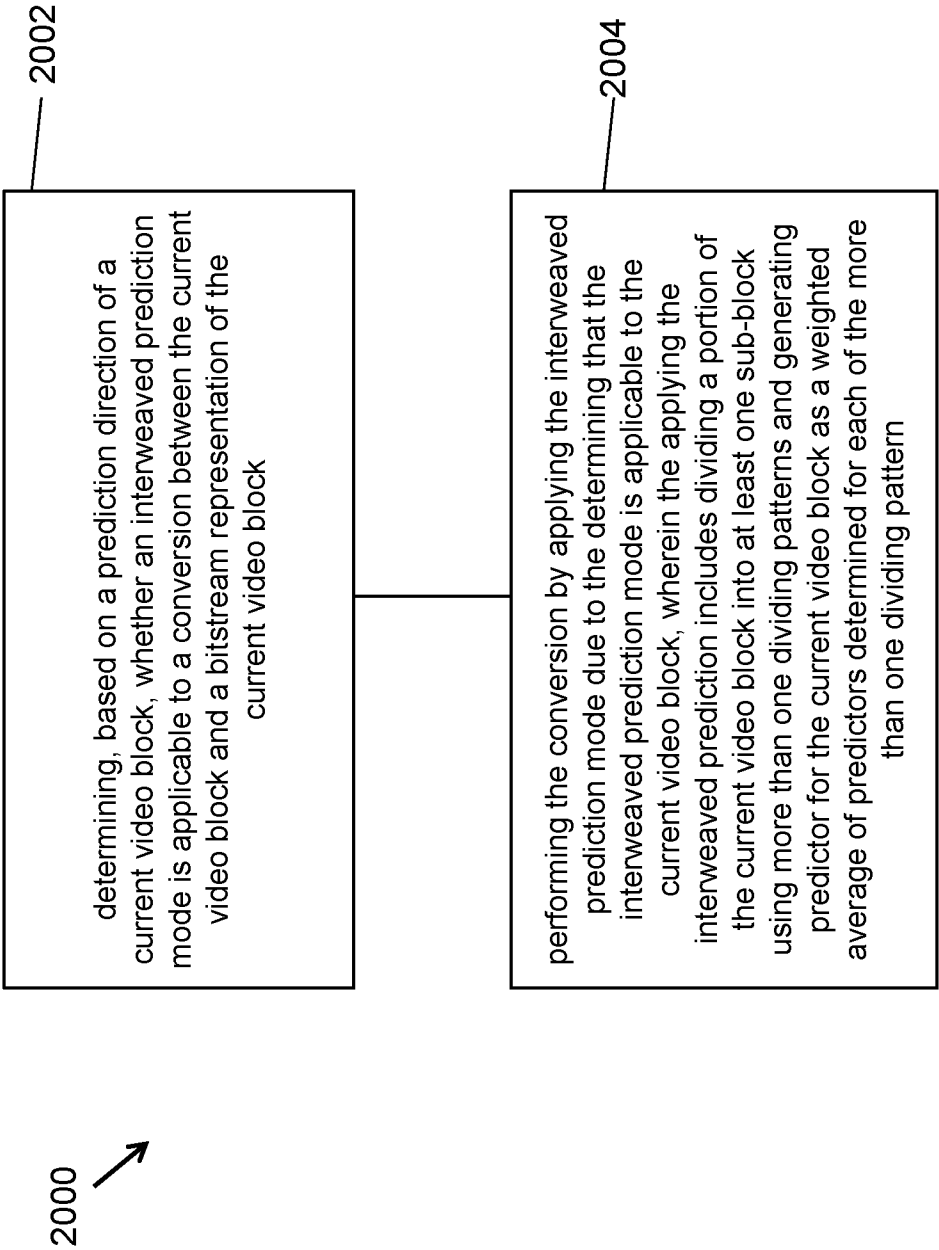
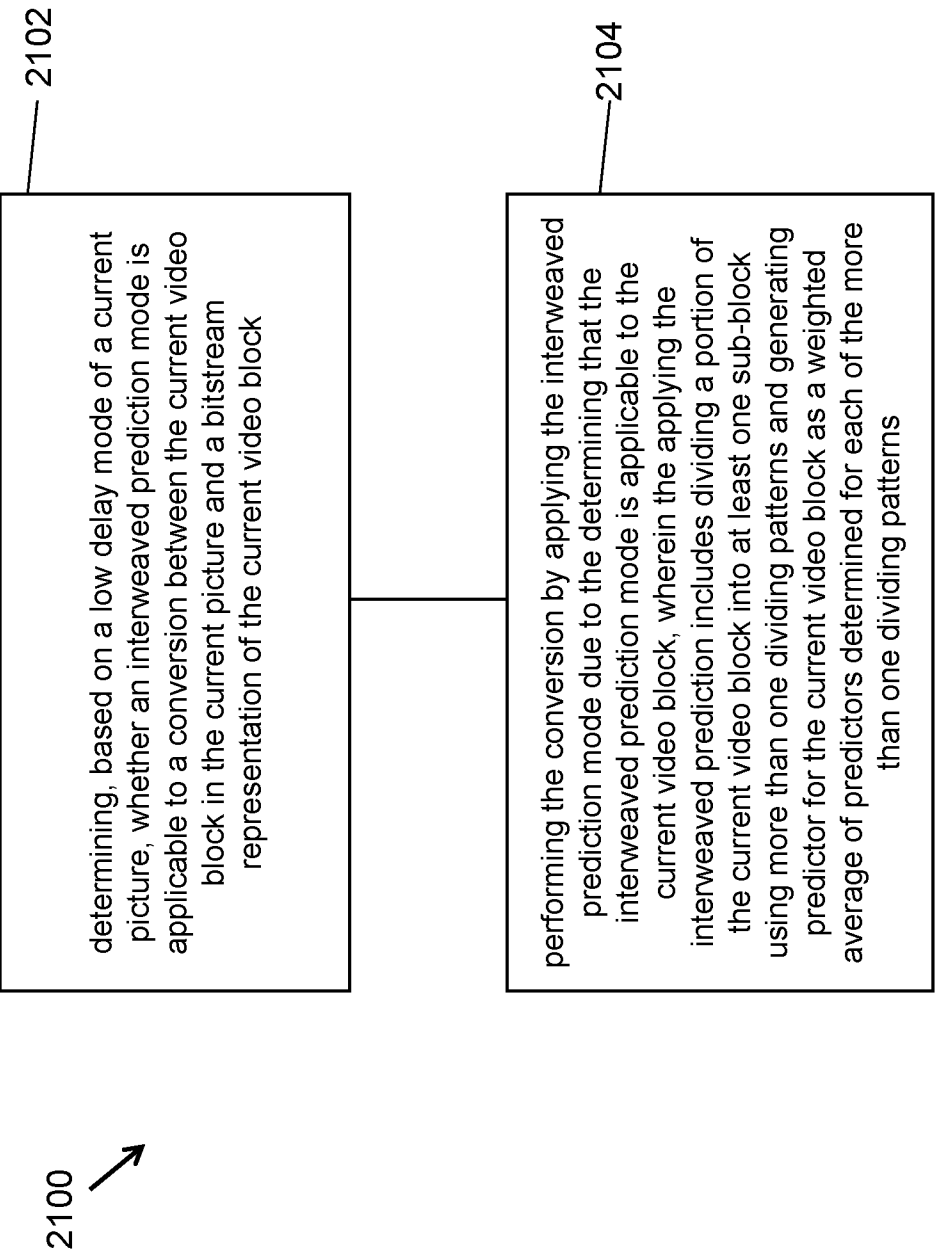
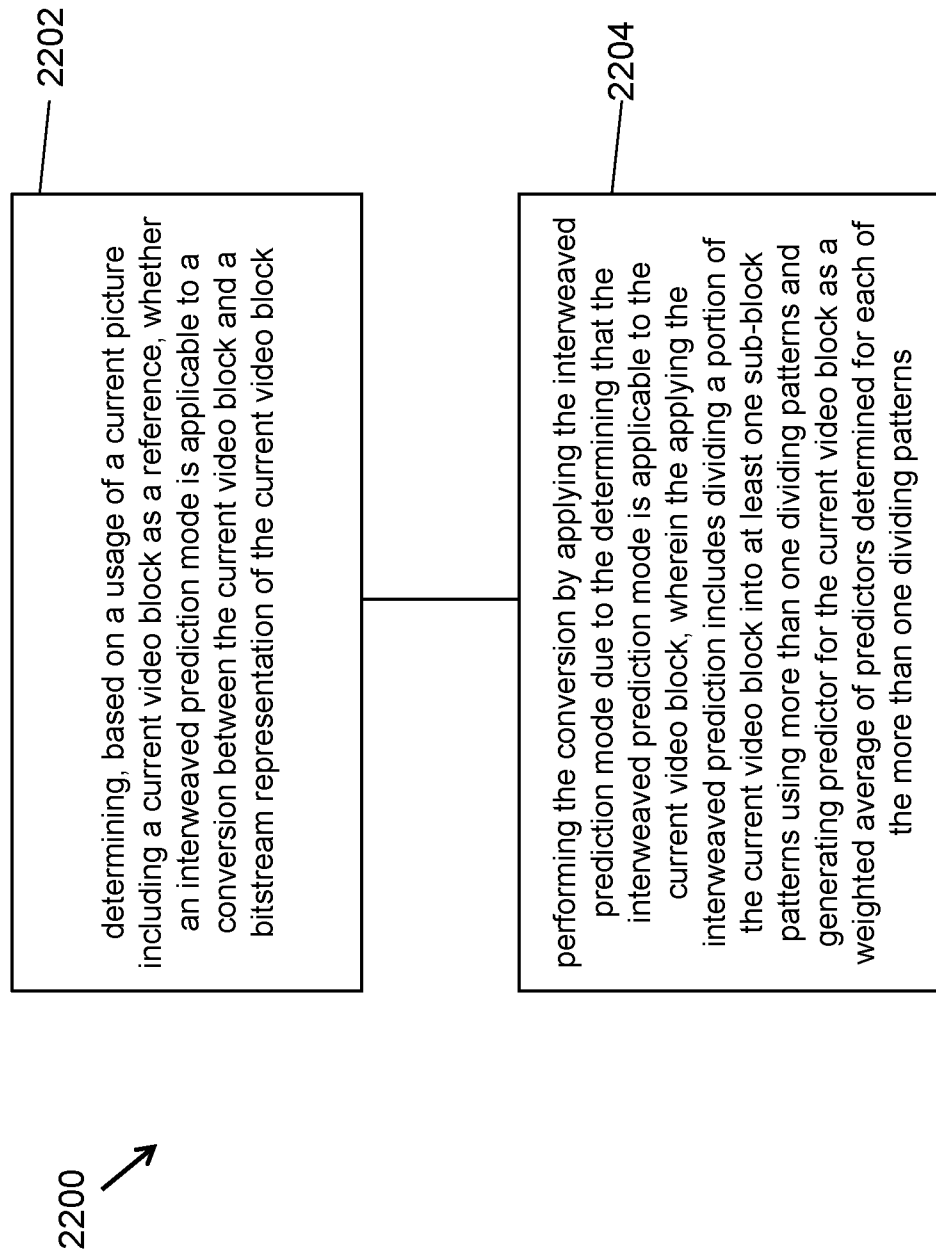


FIG. 20



**FIG. 22**

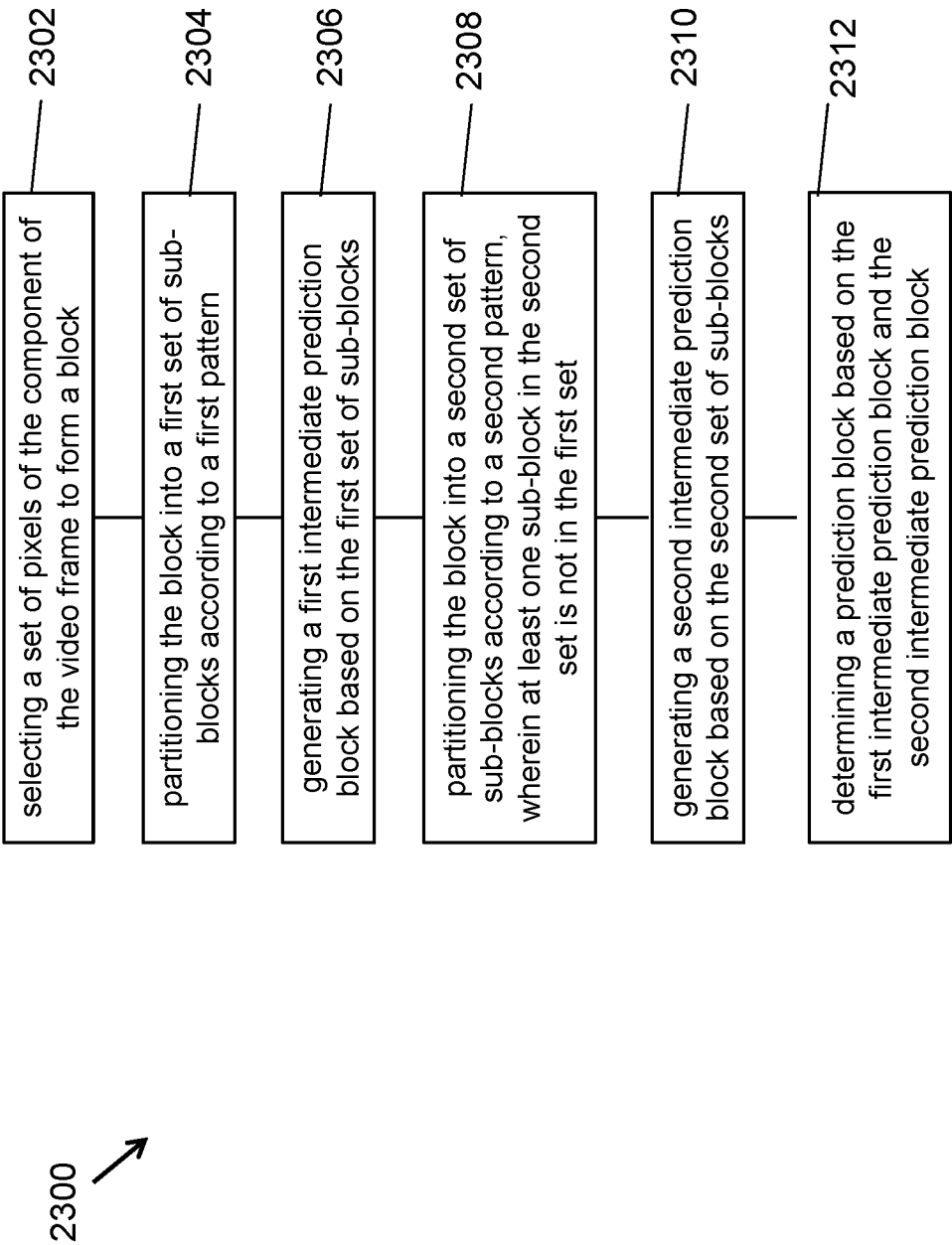


FIG. 23

International application No
PCT/IB2019/055566

INV. H04N19/537 H04N19/573 H04N19/55
ADD.

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)

EP0-Internal, WPI Data, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	HAO LEE ET AL: "OBMC for IVC+", 116. MPEG MEETING; 17-10-2016 - 21-10-2016; CHENGDU; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. m39423, 17 October 2016 (2016-10-17), XP030067769,	1-7, 12-16, 19-21, 28,29, 31-36
A	the whole document ----- -/--	8-11,17, 18, 22-27,30

☒ 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

13 September 2019

Date of mailing of the international search report

26/09/2019

Name and mailing address of the ISA/
European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Cyranka, Oliver

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2019/055566

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	GUO L ET AL: "CE2: Overlapped Block Motion Compensation for Geometry Partition Block", 95. MPEG MEETING; 24-1-2011 - 28-1-2011; DAEGU; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. m19143, 25 January 2011 (2011-01-25), XP030047710,	8,9,17, 24
A	the whole document	1-7, 10-16, 18-23, 25-36
X	----- ALSHINA E ET AL: "Performance of JEM 1 tools analysis", 2. JVET MEETING; 20-2-2016 - 26-2-2016; SAN DIEGO; (THE JOINT VIDEO EXPLORATION TEAM OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://PHENIX.INT-EVRY.FR/JVET/,, no. JVET-B0022-v2, 17 February 2016 (2016-02-17), XP030150010,	10,11, 18,22,27
A	page 10 - page 12 page 16 - page 17	1-9, 12-17, 19-21, 23-26, 28-36
X	----- C-W HSU ET AL: "Description of Core Experiment 10 (CE10): Combined and multi-hypothesis prediction", 10. JVET MEETING; 10-4-2018 - 20-4-2018; SAN DIEGO; (THE JOINT VIDEO EXPLORATION TEAM OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://PHENIX.INT-EVRY.FR/JVET/,, no. JVET-J1030-v2, 12 May 2018 (2018-05-12), XP030151331,	1,5,35, 36
A	the whole document	2-4,6-34
	----- -/--	

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2019/055566

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	ZHANG (BYTEDANCE) K ET AL: "CE4-related: Interweaved Prediction for Affine Motion Compensation", 123. MPEG MEETING; 20180716 - 20180720; LJUBLJANA; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11), , no. m43024 3 July 2018 (2018-07-03), XP030195749, Retrieved from the Internet: URL:http://phenix.int-evry.fr/mpeg/doc_end_user/documents/123_Ljubljana/wg11/m43024-JVET-K0102-v1-JVET-K0102.zip JVET-K0102.docx [retrieved on 2018-07-03] the whole document -----	1-36
X,P	ZHANG (BYTEDANCE) K ET AL: "CE10: Interweaved Prediction for Affine Motion Compensation (Test 10.5.1 and Test 10.5.2)", 124. MPEG MEETING; 20181008 - 20181012; MACAO; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11), , no. m44293 25 September 2018 (2018-09-25), XP030191007, Retrieved from the Internet: URL:http://phenix.int-evry.fr/mpeg/doc_end_user/documents/124_Macao/wg11/m44293-JVET-L0269-v1-JVET-L0269.zip JVET-L0269.docx [retrieved on 2018-09-25] the whole document -----	1-36
X,P	ZHANG (BYTEDANCE) K ET AL: "Non-CE2: Interweaved Prediction for Affine Motion Compensation", 125. MPEG MEETING; 20190114 - 20190118; MARRAKECH; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11), , no. m45534 3 January 2019 (2019-01-03), XP030198080, Retrieved from the Internet: URL:http://phenix.int-evry.fr/mpeg/doc_end_user/documents/125_Marrakech/wg11/m45534-JVET-M0268-v1-JVET-M0268.zip JVET-M0268.docx [retrieved on 2019-01-03] the whole document ----- -/--	1-36

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2019/055566

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	LI (TENCENT) X ET AL: "CE2-related: Simplifications of interweaved affine mode", 14. JVET MEETING; 20190319 - 20190327; GENEVA; (THE JOINT VIDEO EXPLORATION TEAM OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16), , no. JVET-N0399 16 March 2019 (2019-03-16), XP030203313, Retrieved from the Internet: URL:http://phenix.int-evry.fr/jvet/doc_end_user/documents/14_Geneva/wg11/JVET-N0399-v2.zip JVET-N0399_r1.docx [retrieved on 2019-03-16] the whole document -----	1-36