



- (51) **International Patent Classification:**
H04N 7/36 (2006.01) *H04N 7/26* (2006.01)
- (21) **International Application Number:**
PCT/US2013/054965
- (22) **International Filing Date:**
14 August 2013 (14.08.2013)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
61/683,714 15 August 2012 (15.08.2012) US
13/862,297 12 April 2013 (12.04.2013) US
- (71) **Applicant:** MOTOROLA MOBILITY LLC [US/US];
600 North US Highway 45, Libertyville, Illinois 60048 (US).
- (72) **Inventors:** YU, Yue; 10721 Calston Way, San Diego, CS 92126 (US). LOU, Jian; 20651 Forge Way 243, Cupertino, California 95014 (US). WANG, Limin; 5240 Fiore Terrace Apt. 413, San Diego, California 92122 (US).
- (74) **Agents:** BRETSCHER, John T. et al.; 600 North US Highway 45, Libertyville, Illinois 60048 (US).
- (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, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) **Designated States** (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH,

[Continued on next page]

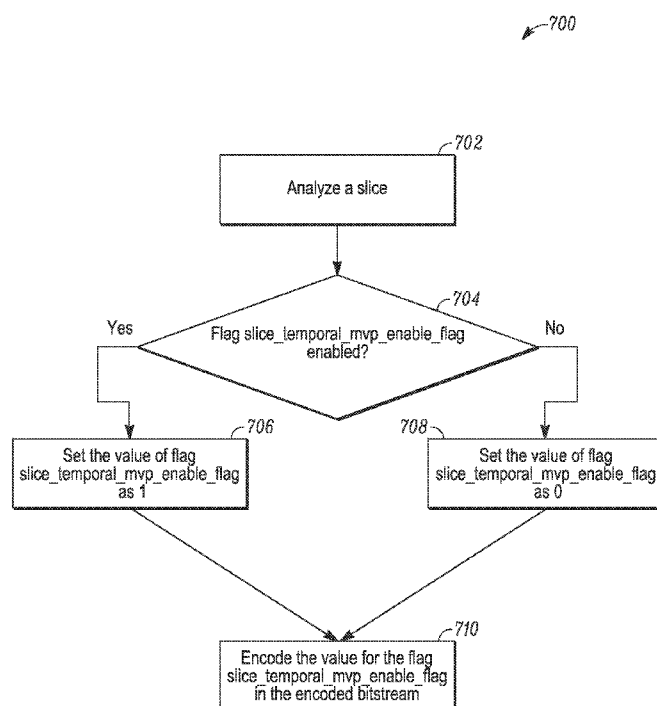
(54) **Title:** SIGNALING OF TEMPORAL MOTION VECTOR PREDICTOR (MVP) ENABLE FLAG

FIG. 7

(57) **Abstract:** In one embodiment, a method determines (702) an I slice in a current picture that is currently being encoded and a value (704) for a flag to enable or disable using temporal motion vector predictors for the I slice. The value for the flag is encoded (710) in an encoded bitstream associated with the I slice being encoded. The method then sends (802) the encoded bitstream to a decoder where encoding (804) the value for the flag for the I slice allows (808) pictures that follow the I picture in a decoding order to use temporal motion vector predictors from a picture that precedes the I picture in the decoding order when temporal motion vector predictors are enabled.



GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, 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))*

SIGNALING OF TEMPORAL MOTION VECTOR PREDICTOR (MVP) ENABLE FLAG

BACKGROUND

[0001] Video compression systems employ block processing for most of the compression operations. A block is a group of neighboring pixels and may be treated as one coding unit in terms of the compression operations. Theoretically, a larger coding unit is preferred to take advantage of correlation among immediate neighboring pixels. Various video compression standards, e.g., Motion Picture Expert Group (MPEG)-1, MPEG-2, and MPEG-4, use block sizes of 4x4, 8x8, and 16x16 (referred to as a macroblock (MB)).

[0002] High efficiency video coding (HEVC) is also a block-based hybrid spatial and temporal predictive coding scheme. HEVC partitions an input picture into square blocks referred to as coding tree units (CTUs) as shown in FIG. 1. Unlike prior coding standards, the CTU can be as large as 64x64 pixels. Each CTU can be partitioned into smaller square blocks called coding units (CUs). FIG. 2 shows an example of a CTU partition of CUs. A CTU 100 is first partitioned into four CUs 102. Each CU 102 may also be further split into four smaller CUs 102 that are a quarter of the size of the CU 102. This partitioning process can be repeated based on certain criteria, such as limits to the number of times a CU can be partitioned may be imposed. As shown, CUs 102-1, 102-3, and 102-4 are a quarter of the size of CTU 100. Further, a CU 102-2 has been split into four CUs 102-5, 102-6, 102-7, and 102-8.

[0003] Each CU 102 may include one or more blocks, which may be referred to as prediction units (PUs). FIG. 3 shows an example of a CU partition of PUs. The PUs may be used to perform spatial prediction or temporal prediction. A CU can be either spatially or temporally predictive coded. If a CU is coded in intra mode, each PU of the CU can

have its own spatial prediction direction. If a CU is coded in inter mode, each PU of the CU can have its own motion vector(s) and associated reference picture(s).

[0004] In HEVC, motion vectors (MVs) are predictively coded in a spatial/temporal prediction process. For a current PU having one current motion vector and an associated reference index, a motion vector predictor (MVP) is derived from motion vectors of spatially neighboring or temporally collocated PUs of the current PU. The difference between the current motion vector and the MVP is then determined and coded. This reduces overhead as only the difference is sent instead of information for the current motion vector. Also, when in merge mode, a single motion vector may be applied to a group of spatially neighboring or temporally collocated PUs.

[0005] In some cases, temporal motion vector predictors may not be used. For example, for an I slice, temporal motion vector predictors are not used to encode or decode PUs of the I slice.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] FIG. 1 shows an input picture partitioned into square blocks referred to as coding tree units (CTUs).

[0007] FIG. 2 shows an example of a CTU partition of CUs.

[0008] FIG. 3 shows an example of a CU partition of PUs.

[0009] FIG. 4 depicts an example of a system for encoding and decoding video content according to one embodiment.

[0010] FIG. 5A shows an example of a decoding order of pictures according to one embodiment.

[0011] FIG. 5B depicts another example of a decoding order in which temporal motion vector predictors may be used from pictures preceding a base layer picture according to one embodiment.

[0012] FIG. 6 depicts an example of a syntax according to one embodiment.

[0013] FIG. 7 depicts a simplified flowchart of a method for encoding video according to one embodiment.

[0014] FIG. 8 depicts a simplified flowchart of a method for decoding video according to one embodiment.

[0015] FIG. 9A depicts an example of encoder according to one embodiment.

[0016] FIG. 9B depicts an example of decoder according to one embodiment.

DETAILED DESCRIPTION

[0017] Described herein are techniques for a video compression system. In the following description, for purposes of explanation, numerous examples and specific details are set forth in order to provide a thorough understanding of particular embodiments. Particular embodiments as defined by the claims may include some or all of the features in these examples alone or in combination with other features described below, and may further include modifications and equivalents of the features and concepts described herein.

[0018] In one embodiment, a method determines a current picture and an I slice of the current picture. A flag is determined for the I slice and a value for the flag is determined for the I slice to enable or disable using temporal motion vector predictors. The method then encodes the value for the flag in an encoded bitstream associated with the I slice and sends the encoded bitstream to a decoder. Encoding the value for the flag allows pictures that follow the current picture in a decoding order to use temporal motion vector predictors from a picture that precedes the current picture in the decoding order when the flag for using temporal motion vector predictors is enabled.

[0019] In one embodiment, a method includes: receiving, by a computing device, an encoded bitstream; determining, by the computing device, a current picture in the

encoded bitstream; determining, by the computing device, an I slice in the current picture; determining, by the computing device, a flag for the I slice; decoding, by the computing device, a value for the flag for the I slice to enable or disable using temporal motion vector predictors; determining, by the computing device, if the value for the flag enables using temporal motion vector predictors for the I slice; and if the value for the flag enables using temporal motion vector predictors for the I slice, allowing, by the computing device, pictures that follow the current picture in a decoding order to use temporal motion vector predictors from a picture that precedes the current picture in the decoding order when temporal motion vector predictors are enabled.

[0020] In one embodiment, a decoder includes: A decoder includes: one or more computer processors; and a non-transitory computer-readable storage medium comprising instructions that, when executed, control the one or more computer processors to be configured for: receiving an encoded bitstream; determining a current picture in the encoded bitstream; determining an I slice in the current picture; determining a flag for the I slice; decoding a value for the flag for the I slice to enable or disable using temporal motion vector predictors; determining if the value for the flag enables using temporal motion vector predictors for the I slice; and if the value for the flag enables using temporal motion vector predictors for the I slice, allowing pictures that follow the current picture in a decoding order to use temporal motion vector predictors from a picture that precedes the current picture in the decoding order when temporal motion vector predictors are enabled.

[0021] In one embodiment, an encoder includes: one or more computer processors; and a non-transitory computer-readable storage medium comprising instructions that, when executed, control the one or more computer processors to be configured for: determining a current picture; determining an I slice of the current picture; determining a flag for the I slice; determining a value for the flag for the I slice to enable or disable using temporal motion vector predictors; encoding the value for the flag in an encoded bitstream associated with the I slice; and allows pictures that follow the current picture in a

decoding order to use temporal motion vector predictors from a picture that precedes the current picture in the decoding order when the flag for using temporal motion vector predictors is enabled.

Overview

[0022] FIG. 4 depicts an example of a system 400 for encoding and decoding video content according to one embodiment. Encoder 402 and decoder 403 may encode and decode a bitstream using HEVC; however, other video compression standards may also be appreciated.

[0023] Encoder 402 includes a temporal motion vector predictor (MVP) block 404-1 and decoder 403 includes a temporal MVP block 404-2. Temporal motion vector predictor (MVP) block 404-1 and temporal MVP block 404-2 may perform functions associated with temporal motion vector prediction as will be described below.

[0024] A flag `slice_temporal_mvp_enable_flag` specifies whether temporal motion vector predictors can be used for inter-prediction. The flag `slice_temporal_mvp_enable_flag` may be associated with a slice of a picture. The slice may include one or more blocks, such as prediction units (PUs). If the flag `slice_temporal_mvp_enable_flag` is disabled, such as equal to 0, then temporal motion vector predictors shall not be used in encoding or decoding of a current slice, or picture. If the flag `slice_temporal_mvp_enable_flag` is enabled, such as equal to 1, then temporal motion vector predictors may be used in encoding or decoding of the current slice, or picture. That is, when temporal MVPs are enabled, temporal MVP blocks 404-1 and 404-2 may decide whether or not temporal motion vector predictors are used in encoding or decoding blocks of a slice. Also, when the flag `slice_temporal_mvp_enable_flag` is not present in an encoded bitstream, then decoder 403 infers that the value of the flag `slice_temporal_mvp_enable_flag` is disabled, or equal to 0.

[0025] A collocated reference picture includes motion vector (MV) information that can be used in a temporal prediction process. For example, a motion vector for a partition in the collocated reference picture may be used as a temporal motion vector predictor (MVP) in advanced motion vector prediction (AMVP) or as a temporal merge/skip candidate motion vector in a merge process. When the motion vector of the collocated block is used as a temporal MVP, a difference of the motion vector of the collocated block and the current motion vector of the current block is coded and sent from encoder 402 to decoder 403. When the motion vector of the collocated block is used in a merge process, the current block and the collocated block are merged and use the same motion vector of the collocated block.

[0026] The temporal MVP or temporal merge/skip candidate MV is defined as the MV of the selected collocated block, which resides in a reference picture and has the same geometrical position as a current block in the current picture. In one example, given a PU in a current slice, an associated collocated PU can reside in a preceding or subsequent PU in time. The collocated PU can be used as a candidate to enable MVP and merge/skip mode for the current PU. Like with a PU, a collocated block can be provided for a transform unit (TU), CU or a picture itself. Subsequent discussion will reference a slice, although the same could apply to any of the listed subcomponents of a picture. A slice may be one or more blocks. Each picture may have one or more slices. Additionally, a block may be referenced, which may be a PU, but may also be a TU, CU, a picture, or a slice.

[0027] Pictures may be classified in different layers. For example, when a picture is classified as a base layer picture, the picture is a reference picture with a temporal identifier (TemporalID) equal to 0. A base layer picture may be an I, P or B picture and can be references of other layers (e.g., the enhancement layer). When both the flag `slice_temporal_mvp_enable_flag` and TemporalID are equal to 0, all coded pictures that follow the current picture in the decoding order shall not use temporal motion vector predictors from any picture that precedes the current picture in the decoding order.

[0028] FIG. 5A shows an example of a decoding order of pictures according to one embodiment. The decoding order is the order pictures are decoded and may not necessarily represent the order in which pictures are displayed in the video. It should be noted that the decoding may be performed in the encoding process in encoder 402 or the decoding process in decoder 403. FIG. 5A shows a group of pictures 500. A current picture is shown at 502. Pictures 0, 1, and 2 are pictures that have been decoded already. Pictures 4, 5, and 6 are pictures that have not been decoded yet. At 502, a current picture is being decoded.

[0029] In this case, current picture 3 is an I picture, which means that current picture 3 includes I slices. Also, the TemporalID for current picture 3 is equal to 0 meaning that current picture 3 is a base layer picture. Conventionally, the flag `slice_temporal_mvp_enable_flag` for an I slice included in current picture 3 is disabled, or equal to 0. When these conditions are true (e.g., an I picture with TemporalID = 0 and the flag `slice_temporal_mvp_enable_flag` = 0), then pictures 4, 5, and 6 cannot use temporal motion vector predictors from pictures that precede current picture 3 in the decoding order, such as pictures 0, 1, and 2. In one example, pictures 4, 5, and 6 may use temporal motion vector predictors from current picture 3, and any pictures after current picture 3 in the decoding order. As shown, picture 4 may not use any temporal motion vector predictors because picture 3 is a I picture, picture 5 may use temporal motion vector predictors from picture 4, and picture 6 may use temporal motion vector predictors from picture 5.

[0030] FIG. 5B depicts another example of a decoding order in which temporal motion vector predictors may be used from pictures preceding an I picture according to one embodiment. The same decoding order as shown in FIG. 5A is shown in FIG. 5B. However, in this case, the flag `slice_temporal_mvp_enable_flag` for an I slice included in current picture 3 is enabled, or equal to 1. Additionally, the TemporalID for current picture 3 is equal to 0. In this case, the condition to test whether temporal MVs can be used for pictures preceding the current picture 3 is not met as the value of the flag

slice_temporal_mvp_enable_flag is equal to 1 and the TemporalID is equal to 0. Thus, pictures 4, 5, and 6 may use temporal motion vector predictors from pictures that precede current picture 3 in the decoding order. For example, as shown at 504, picture 4 may use a temporal motion vector predictor from a picture 2. By allowing pictures 4, 5, and 6 to use temporal motion vector predictors from pictures preceding current picture 3, the coding performance may be improved. For example, pictures before current picture 3 may include blocks that may better predict blocks in picture 4.

[0031] Conventionally, the flag slice_temporal_mvp_enable_flag was not coded for an I slice. That is, encoder 402 would not have encoded a value for the flag slice_temporal_mvp_enable_flag for an I slice in the encoded bitstream and decoder 403 would not have decoded a value for the flag slice_temporal_mvp_enable_flag from the encoded bitstream. The reasoning for this is that an I slice does not use temporal MVs and thus coding a flag for the I slice would be unnecessary. This also reduces overhead as fewer bits need to be encoded if the flag slice_temporal_mvp_enable_flag is not coded for an I slice. When the flag slice_temporal_mvp_enable_flag is not included in the encoded bitstream, then decoder 403 infers that the value is equal to 0. Thus, the value of the flag slice_temporal_mvp_enable_flag would always be inferred to be 0 for an I slice.

[0032] In contrast to the conventional process, in particular embodiments, encoder 402 may encode a value for the flag slice_temporal_mvp_enable_flag for I slices. Thus, it is possible that the value for the flag slice_temporal_mvp_enable_flag is equal to 1 when coding an I slice. Even though the value of the flag slice_temporal_mvp_enable_flag is equal to 1, temporal MVs are not used for the I slice, such as decoder 403 may choose not to use temporal MVs for the I slice. However, the effect of this is shown in FIG. 5B, where when the value of the flag slice_temporal_mvp_enable_flag is equal to 1 and the value of the TemporalID is equal to 0, pictures after a current picture in the decoding order can use temporal MVs from pictures preceding the current picture in the decoding order. Although this may increase overhead as encoder 402 sends more bits in the encoded bitstream, this provides flexibility to improve coding performance. That is, the

flexibility to allow pictures after an I picture in the decoding order to use temporal MVs from pictures preceding the I picture is possible. In one embodiment, a value for the flag `slice_temporal_mvp_enable_flag` may be coded for all slices, such as I slices, P slices, and B slices. In another embodiment, encoder 402 may encode the value for the flag `slice_temporal_mvp_enable_flag` when encoder 402 determines that the value should be 1. When encoder 402 determines that the value is 0, then encoder 402 may choose not to encode the value and then decoder 403 infers that the value is 0 by default.

[0033] A syntax may be used to govern when and where to insert the flag `slice_temporal_mvp_enable_flag` in the encoded bitstream. FIG. 6 depicts an example of a syntax 600 according to one embodiment. Syntax 600 may be included in a slice header, but the syntax may apply for other headers, such as a sequence parameter set (SPS) header, picture parameter set (PPS) header, or block header. The slice header may be associated with a slice of blocks for a picture. At 602, syntax 600 tests a conditional statement, such as an if statement, for a flag `sps_temporal_mvp_enable_flag`. The flag `sps_temporal_mvp_enable_flag` may be a flag that indicates whether temporal motion vector predictors are used for the whole sequence. When temporal MVs are not enabled for a sequence of pictures that includes a current picture (or slice), then the value for the flag `slice_temporal_mvp_enable_flag` does not need to be included in the encoded bitstream because temporal MVs are not enabled for any pictures for a whole sequence.

[0034] If temporal MVs are enabled for the SPS, then syntax 600 may always include a value for the flag `slice_temporal_mvp_enable_flag`. That is, for I slices, P slices, and B slices, at 604, a value for the flag `slice_temporal_mvp_enable_flag` is included in the encoded bitstream if the value of the flag `sps_temporal_mvp_enable_flag` is 1. In other embodiments, when encoder 402 determines that the value is 0, then encoder 402 may choose not to encode the value and then decoder 403 infers that the value is 0 by default.

[0035] FIG. 7 depicts a simplified flowchart 700 of a method for encoding video according to one embodiment. At 702, encoder 402 analyzes a slice, such as an I slice of

an I picture. For example, encoder 402 analyzes characteristics of the video to determine, at 704, if the flag `slice_temporal_mvp_enable_flag` should be enabled for the slice. For example, in some cases, encoder 402 might want to enable the flag `slice_temporal_mvp_enable_flag` for an I slice even though temporal MVs cannot be used for the I slice. This would allow pictures after the I slice in the decoding order to use temporal MVs from pictures preceding the I slice in the decoding order. In one example, encoder 402 determines that pictures preceding a current picture may better predict pictures occurring after the current picture in the decoding order. Then, encoder 402 enables the flag `slice_temporal_mvp_enable_flag`.

[0036] If the flag `slice_temporal_mvp_enable_flag` is enabled, at 706, temporal MVP block 404-1 sets the value as 1. At 708, encoder 402 encodes the value for the flag `slice_temporal_mvp_enable_flag` in the encoded bitstream. For example, encoder 402 encodes the value in a slice header associated with the I slice.

[0037] At 708, if the flag `slice_temporal_mvp_enable_flag` should not be enabled, temporal MVP block 404-1 sets the value as 0. In other embodiments, temporal MVP block 404-1 may not set the value as 0 and the flag `slice_temporal_mvp_enable_flag` is not included in the encoded bitstream. As discussed above, the decoding process may then infer that this value is 0.

[0038] At 710, encoder 402 encodes the value for the flag `slice_temporal_mvp_enable_flag` in the encoded bitstream. As discussed above, encoder 402 may always include the value for the flag `slice_temporal_mvp_enable_flag` in the encoded bitstream for all slice types. In other embodiments, encoder 402 may encode the value for the flag `slice_temporal_mvp_enable_flag` when the value is "1" for all slice types.

[0039] FIG. 8 depicts a simplified flowchart 800 of a method for decoding video according to one embodiment. At 802, decoder 403 receives an encoded bitstream from

encoder 402. The encoded bitstream includes encoded video along with header information defined by syntax 600 in FIG. 6.

[0040] At 804, temporal MVP block 404-2 determines a value for the flag `slice_temporal_mvp_enable_flag`. For example, a slice header includes an encoded value for the flag `slice_temporal_mvp_enable_flag` for an I slice currently being decoded.

[0041] At 806, decoder 403 determines if pictures following the I slice can use temporal MVs from pictures preceding the I slice in the decoding order. For example, pictures following the I slice can use temporal MVs from pictures before the I slice when the value of the flag `slice_temporal_mvp_enable_flag` is equal to 1 and the TemporalID is equal to 0. At 808, if the above conditions are true, decoder 403 allows pictures that follow an I picture including the current I slice (but the current I slice does not need to) to use temporal MVs from pictures that precede the I picture in the decoding order. However, if the above conditions are not true (i.e., the value of the flag `slice_temporal_mvp_enable_flag` is equal to 0 and the TemporalID is equal to 0), at 810, decoder 403 does not allow pictures that follow the I slice to use temporal MVs from pictures that precede the I picture in the decoding order.

[0042] Accordingly, particular embodiments may add overhead to the encoded bitstream by having a value for the flag `slice_temporal_mvp_enable_flag` encoded for I slices (in addition to P slices and B slices). Even though overhead may be added, coding efficiency may be gained by allowing temporal MVs to be used from pictures that precede an I picture.

[0043] The flag `slice_temporal_mvp_enable_flag` is a slice level syntax, and therefore should be used to specify slice-level constraints. If the flag `slice_temporal_mvp_enable_flag` is set equal to 0 for one slice of a picture and to 1 for another slice of the same picture, decoder 403 will have problems to decide if temporal motion vector predictors shall be used or not for the picture. In one embodiment, the values of flag `slice_temporal_mvp_enable_flag` for different slices within a same picture

should be identical. Furthermore, the flag `slice_temporal_mvp_enable_flag` will be used to specify whether temporal motion vector predictors shall be used for a current slice, instead of a current picture. The following semantic summarizes the above:

The flag `slice_temporal_mvp_enable_flag` specifies whether temporal motion vector predictors can be used for inter prediction. If the flag `slice_temporal_mvp_enable_flag` is equal to 0, the temporal motion vector predictors shall not be used in decoding of the current slice. If the flag `slice_temporal_mvp_enable_flag` is equal to 1, temporal motion vector predictors may be used in decoding of the current slice. When not present, the value of the flag `slice_temporal_mvp_enable_flag` shall be inferred to be equal to 0. The flag `slice_temporal_mvp_enable_flag` should have the same value for all slices within a same picture.

Particular embodiments set the value of `slice_temporal_mvp_enable_flag` for different slices within a same picture to be identical. The corresponding semantics changes for `slice_temporal_mvp_enable_flag` are as follows.

The flag `slice_temporal_mvp_enable_flag` specifies whether temporal motion vector predictors can be used for inter prediction. If the flag `slice_temporal_mvp_enable_flag` is equal to 0, the temporal motion vector predictors shall not be used in decoding of the current picture. If the flag `slice_temporal_mvp_enable_flag` is equal to 1, temporal motion vector predictors may be used in decoding of the current picture. When not present, the value of the flag `slice_temporal_mvp_enable_flag` shall be inferred to be equal to 0. The flag `slice_temporal_mvp_enable_flag` should have the same value for all slices within a same picture.

In the above semantic, if the value of the flag `slice_temporal_mvp_enable_flag` is 1, then temporal motion vector predictors may be used in all slices of the picture.

Encoder and Decoder Examples

[0044] In various embodiments, encoder 402 described can be incorporated or otherwise associated with a transcoder or an encoding apparatus at a headend and decoder 403 can be incorporated or otherwise associated with a downstream device, such as a mobile device, a set top box or a transcoder. FIG. 9A depicts an example of encoder 402 according to one embodiment. A general operation of encoder 402 will now be described; however, it will be understood that variations on the encoding process described will be appreciated by a person skilled in the art based on the disclosure and teachings herein.

[0045] For a current PU, x , a prediction PU, x' , is obtained through either spatial prediction or temporal prediction. The prediction PU is then subtracted from the current PU, resulting in a residual PU, e . Spatial prediction relates to intra mode pictures. Intra mode coding can use data from the current input image, without referring to other images, to code an I picture. A spatial prediction block 904 may include different spatial prediction directions per PU, such as horizontal, vertical, 45-degree diagonal, 135-degree diagonal, DC (flat averaging), and planar, or any other direction. The spatial prediction direction for the PU can be coded as a syntax element. In some embodiments, brightness information (Luma) and color information (Chroma) for the PU can be predicted separately. In one embodiment, the number of Luma intra prediction modes for all block sizes is 35. An additional mode can be used for the Chroma intra prediction mode. In some embodiments, the Chroma prediction mode can be called "IntraFromLuma."

[0046] Temporal prediction block 906 performs temporal prediction. Inter mode coding can use data from the current input image and one or more reference images to code "P" pictures and/or "B" pictures. In some situations and/or embodiments, inter mode coding can result in higher compression than intra mode coding. In inter mode PUs 204 can be temporally predictive coded, such that each PU 204 of the CU 202 can have one or more motion vectors and one or more associated reference images. Temporal prediction can be performed through a motion estimation operation that searches for a best match

prediction for the PU over the associated reference images. The best match prediction can be described by the motion vectors and associated reference images. P pictures use data from the current input image and one or more reference images, and can have up to one motion vector. B pictures may use data from the current input image and one or more reference images, and can have up to two motion vectors. The motion vectors and reference pictures can be coded in the encoded bitstream. In some embodiments, the motion vectors can be syntax elements “MV,” and the reference pictures can be syntax elements “refIdx.” In some embodiments, inter mode can allow both spatial and temporal predictive coding. The best match prediction is described by the motion vector (MV) and associated reference picture index (refIdx). The motion vector and associated reference picture index are included in the coded bitstream.

[0047] Transform block 907 performs a transform operation with the residual PU, e. A set of block transforms of different sizes can be performed on a CU, such that some PUs can be divided into smaller TUs and other PUs can have TUs the same size as the PU. Division of CUs and PUs into TUs 20 can be shown by a quadtree representation. Transform block 907 outputs the residual PU in a transform domain, E.

[0048] A quantizer 908 then quantizes the transform coefficients of the residual PU, E. Quantizer 908 converts the transform coefficients into a finite number of possible values. In some embodiments, this is a lossy operation in which data lost by quantization may not be recoverable. After the transform coefficients have been quantized, entropy coding block 910 entropy encodes the quantized coefficients, which results in final compression bits to be transmitted. Different entropy coding methods may be used, such as context-adaptive variable length coding (CAVLC) or context-adaptive binary arithmetic coding (CABAC).

[0049] Also, in a decoding process within encoder 402, a de-quantizer 912 de-quantizes the quantized transform coefficients of the residual PU. De-quantizer 912 then outputs the de-quantized transform coefficients of the residual PU, E'. An inverse transform

block 914 receives the de-quantized transform coefficients, which are then inverse transformed resulting in a reconstructed residual PU, e' . The reconstructed PU, e' , is then added to the corresponding prediction, x' , either spatial or temporal, to form the new reconstructed PU, x'' . Particular embodiments may be used in determining the prediction, such as collocated reference picture manager 404 is used in the prediction process to determine the collocated reference picture to use. A loop filter 916 performs de-blocking on the reconstructed PU, x'' , to reduce blocking artifacts. Additionally, loop filter 916 may perform a sample adaptive offset process after the completion of the de-blocking filter process for the decoded picture, which compensates for a pixel value offset between reconstructed pixels and original pixels. Also, loop filter 916 may perform adaptive loop filtering over the reconstructed PU, which minimizes coding distortion between the input and output pictures. Additionally, if the reconstructed pictures are reference pictures, the reference pictures are stored in a reference buffer 918 for future temporal prediction. Intra mode coded images can be a possible point where decoding can begin without needing additional reconstructed images.

[0050] FIG. 9B depicts an example of decoder 403 according to one embodiment. A general operation of decoder 403 will now be described; however, it will be understood that variations on the decoding process described will be appreciated by a person skilled in the art based on the disclosure and teachings herein. Decoder 403 receives input bits from encoder 402 for encoded video content.

[0051] An entropy decoding block 930 performs entropy decoding on the input bitstream to generate quantized transform coefficients of a residual PU. A de-quantizer 932 de-quantizes the quantized transform coefficients of the residual PU. De-quantizer 932 then outputs the de-quantized transform coefficients of the residual PU, E' . An inverse transform block 934 receives the de-quantized transform coefficients, which are then inverse transformed resulting in a reconstructed residual PU, e' .

[0052] The reconstructed PU, e' , is then added to the corresponding prediction, x' , either spatial or temporal, to form the new reconstructed PU, x'' . A loop filter 936 performs de-blocking on the reconstructed PU, x'' , to reduce blocking artifacts. Additionally, loop filter 936 may perform a sample adaptive offset process after the completion of the de-blocking filter process for the decoded picture, which compensates for a pixel value offset between reconstructed pixels and original pixels. Also, loop filter 936 may perform adaptive loop filtering over the reconstructed PU, which minimizes coding distortion between the input and output pictures. Additionally, if the reconstructed pictures are reference pictures, the reference pictures are stored in a reference buffer 938 for future temporal prediction.

[0053] The prediction PU, x' , is obtained through either spatial prediction or temporal prediction. A spatial prediction block 940 may receive decoded spatial prediction directions per PU, such as horizontal, vertical, 45-degree diagonal, 135-degree diagonal, DC (flat averaging), and planar. The spatial prediction directions are used to determine the prediction PU, x' .

[0054] A temporal prediction block 906 performs temporal prediction through a motion estimation operation. Particular embodiments may be used in determining the prediction, such as collocated reference picture manager 404 is used in the prediction process to determine the collocated reference picture to use. A decoded motion vector is used to determine the prediction PU, x' . Interpolation may be used in the motion estimation operation.

[0055] Particular embodiments may be implemented in a non-transitory computer-readable storage medium for use by or in connection with the instruction execution system, apparatus, system, or machine. The computer-readable storage medium contains instructions for controlling a computer system to perform a method described by particular embodiments. The instructions, when executed by one or more computer

processors, may be operable to perform that which is described in particular embodiments.

[0056] As used in the description herein and throughout the claims that follow, “a”, “an”, and “the” includes plural references unless the context clearly dictates otherwise. Also, as used in the description herein and throughout the claims that follow, the meaning of “in” includes “in” and “on” unless the context clearly dictates otherwise.

[0057] The above description illustrates various embodiments along with examples of how aspects of particular embodiments may be implemented. The above examples and embodiments should not be deemed to be the only embodiments, and are presented to illustrate the flexibility and advantages of particular embodiments as defined by the following claims. Based on the above disclosure and the following claims, other arrangements, embodiments, implementations and equivalents may be employed without departing from the scope hereof as defined by the claims.

CLAIMS

We claim:

1. A method comprising:
 - determining, by a computing device, a current picture;
 - determining (702), by the computing device, an I slice of the current picture;
 - determining (704), by the computing device, a flag for the I slice;
 - determining (706, 708), by the computing device, a value for the flag for the I slice to enable or disable using temporal motion vector predictors;
 - encoding (710), by the computing device, the value for the flag in an encoded bitstream associated with the I slice; and
 - sending, by the computing device, the encoded bitstream to a decoder, wherein encoding the value for the flag allows pictures that follow the current picture in a decoding order to use temporal motion vector predictors from a picture that precedes the current picture in the decoding order when the flag for using temporal motion vector predictors is enabled.
2. The method of claim 1 further comprising:
 - determining if temporal motion vector predictors are enabled for a sequence of pictures including the I slice;
 - when temporal motion vector predictors are enabled for the sequence of pictures, determining the flag for the I slice; and
 - when temporal motion vector predictors are disabled for the sequence of pictures, not determining the flag and not encoding the value for the value.
3. The method of claim 1 wherein the current picture has a temporal identifier (ID) that is equal to 0.

4. The method of claim 3 wherein when the flag is enabled and the temporal ID for the current picture is equal to 0, the pictures that follow the current picture in the decoding order are allowed to use temporal motion vectors from a picture that precedes the current picture in the decoding order.
5. The method of claim 3 wherein when the flag is disabled and the temporal ID is equal to 0, the pictures that follow the current picture in the decoding order are not allowed to use temporal motion vector predictors from a picture that precedes the current picture in the decoding order.
6. The method of claim 1 wherein when the flag is determined, the value for the flag is always encoded.
7. The method of claim 1 wherein when the flag is not determined, the method further comprises not encoding the value for the flag in the encoded bitstream, wherein the decoder infers the value for the flag to be disabled.
8. The method of claim 1 wherein the value for the flag is always encoded for I slices, P slices, and B slices when the flag is determined.
9. The method of claim 1 wherein the value for the flag is the same for all slices in a picture.

10. An encoder comprising:
- one or more computer processors; and
 - a non-transitory computer-readable storage medium comprising instructions that, when executed, control the one or more computer processors to be configured for:
 - determining a current picture;
 - determining (702) an I slice of the current picture;
 - determining (704) a flag for the I slice;
 - determining (706, 708) a value for the flag for the I slice to enable or disable using temporal motion vector predictors;
 - encoding (710) the value for the flag in an encoded bitstream associated with the I slice; and
 - sending the encoded bitstream to a decoder, wherein encoding the value for the flag allows pictures that follow the current picture in a decoding order to use temporal motion vector predictors from a picture that precedes the current picture in the decoding order when the flag for using temporal motion vector predictors is enabled.

1/8

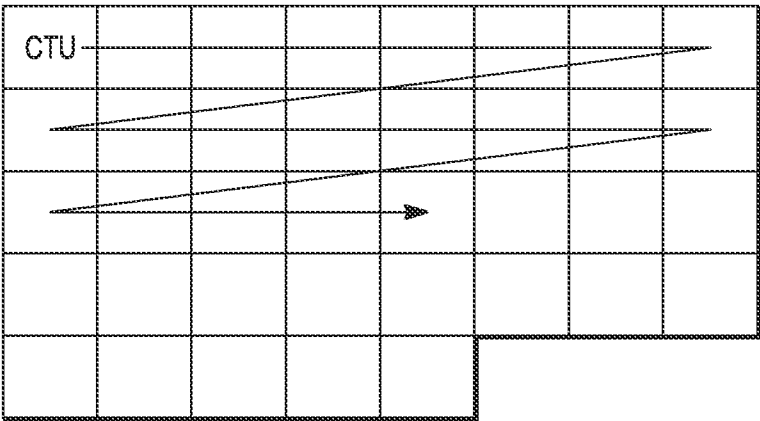


FIG. 1 (PRIOR ART)

100

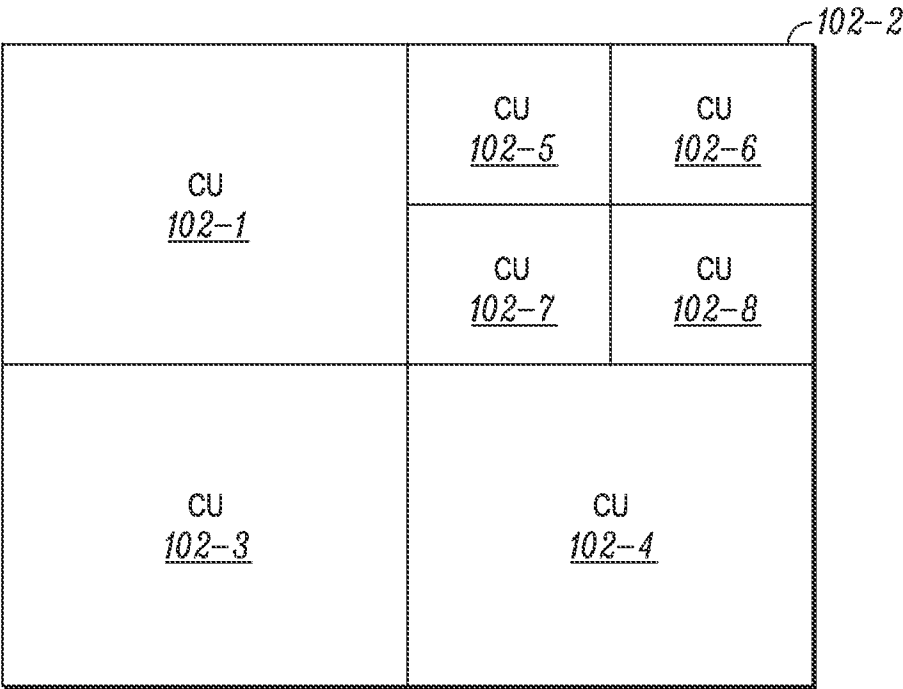


FIG. 2 (PRIOR ART)

2/8

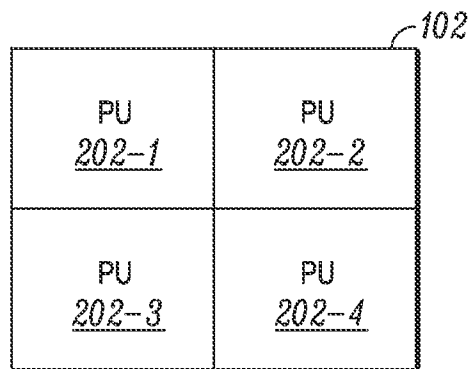


FIG. 3 (PRIOR ART)

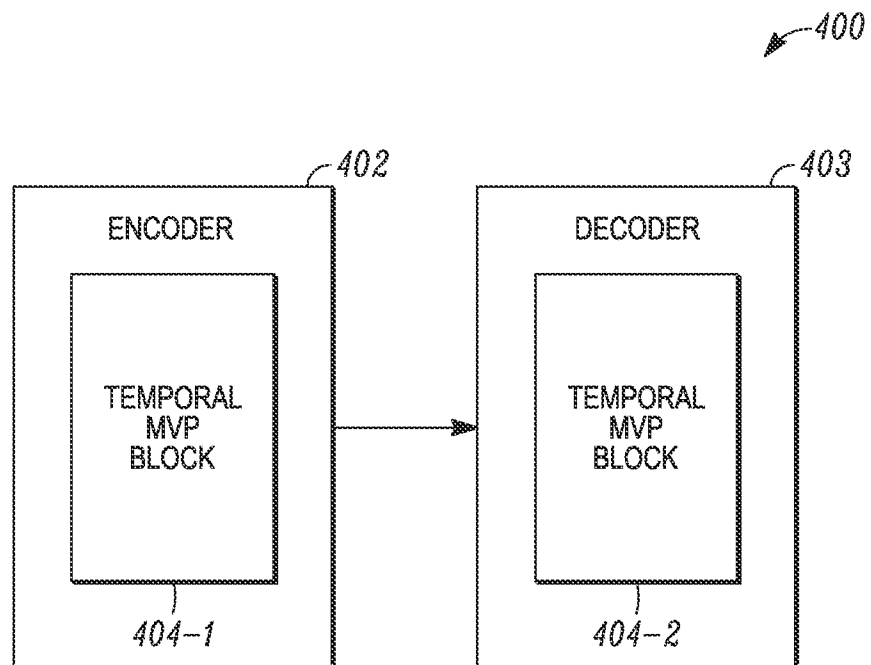


FIG. 4

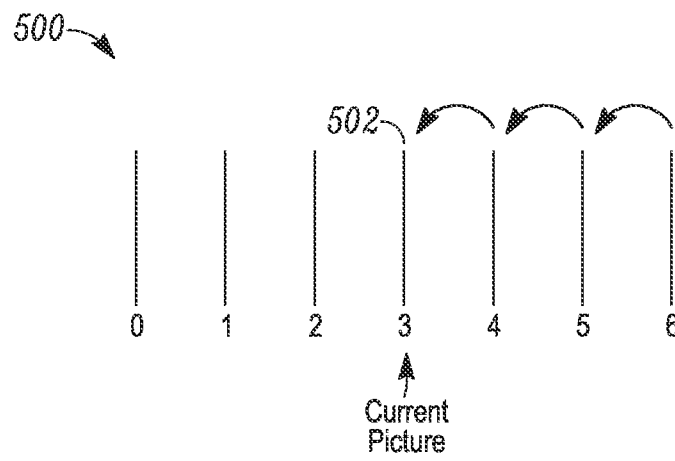
3/8

FIG. 5A

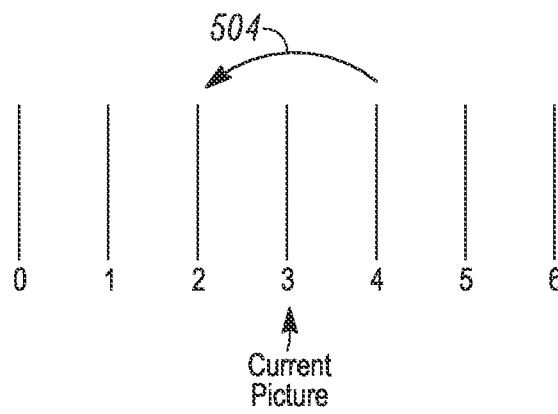


FIG. 5B

4/8

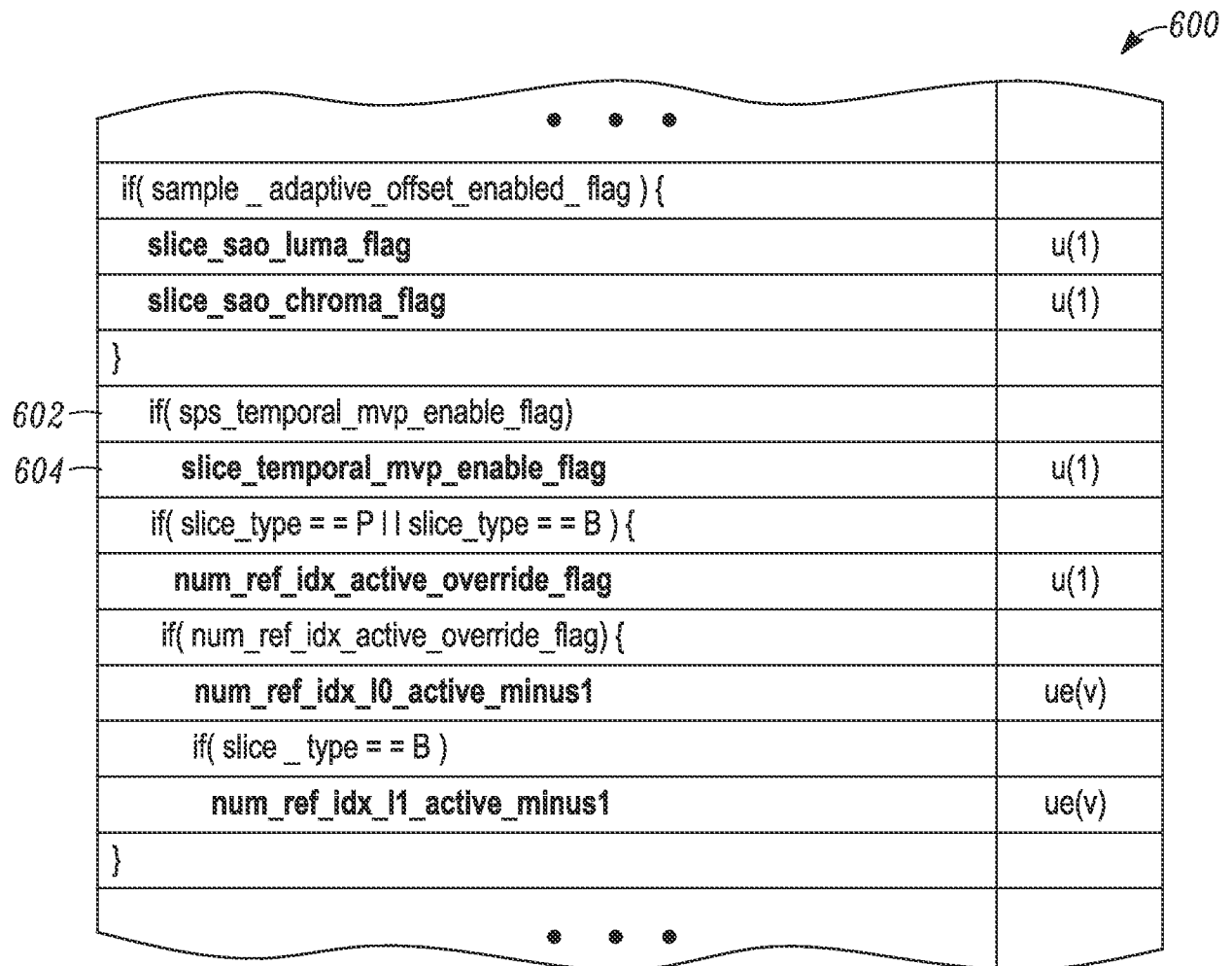


FIG. 6

5/8

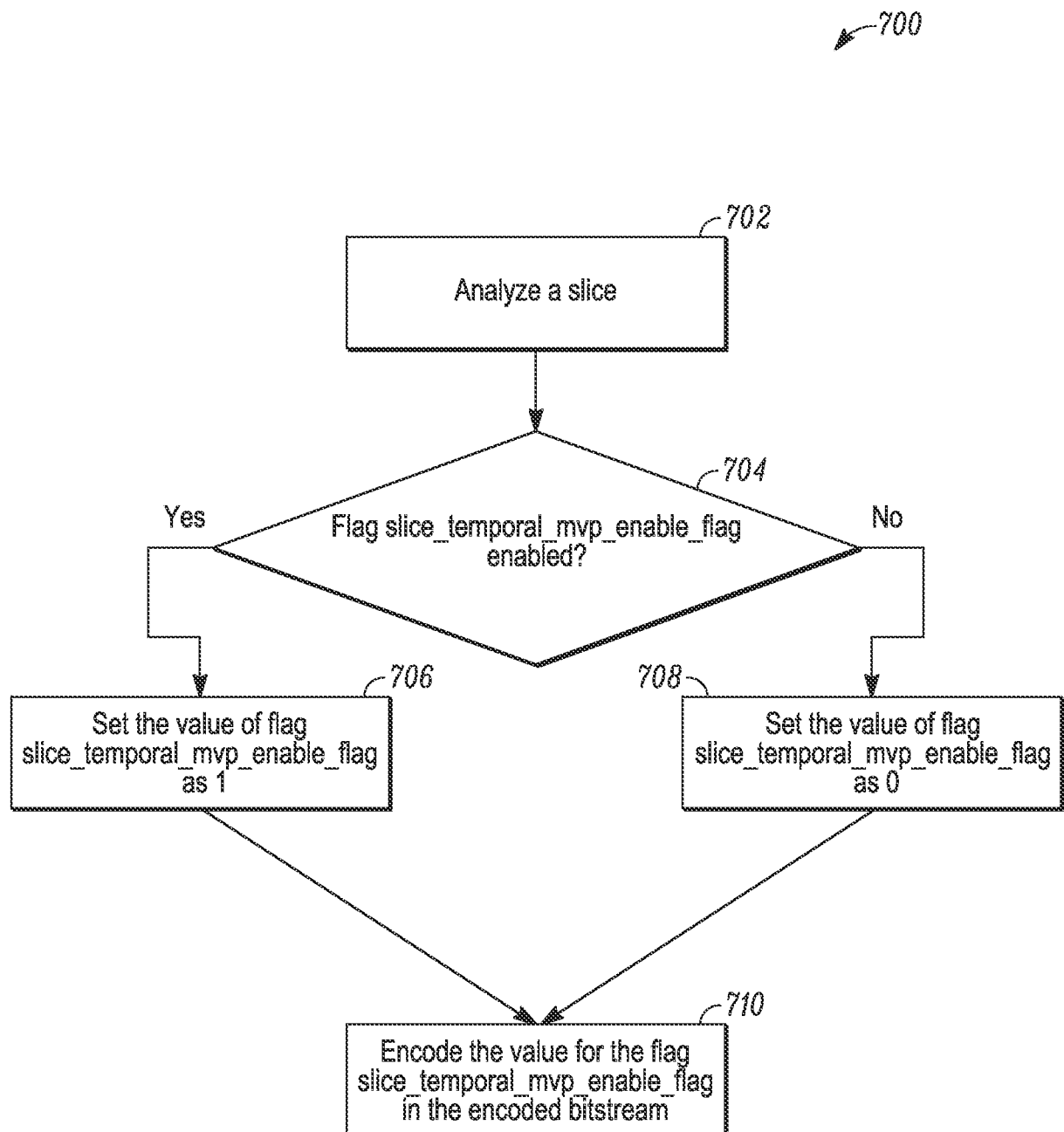


FIG. 7

6/8

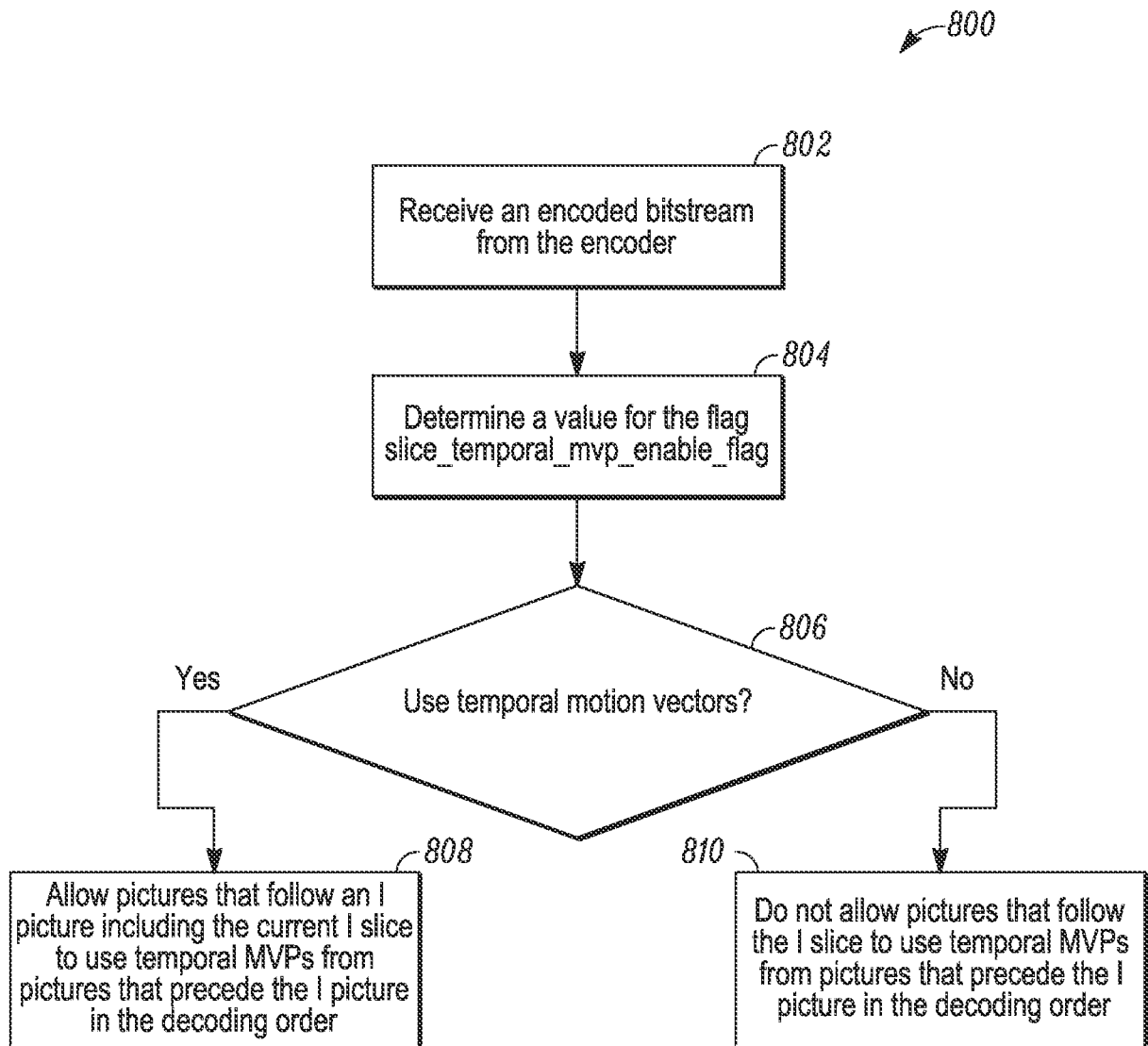


FIG. 8

7/8

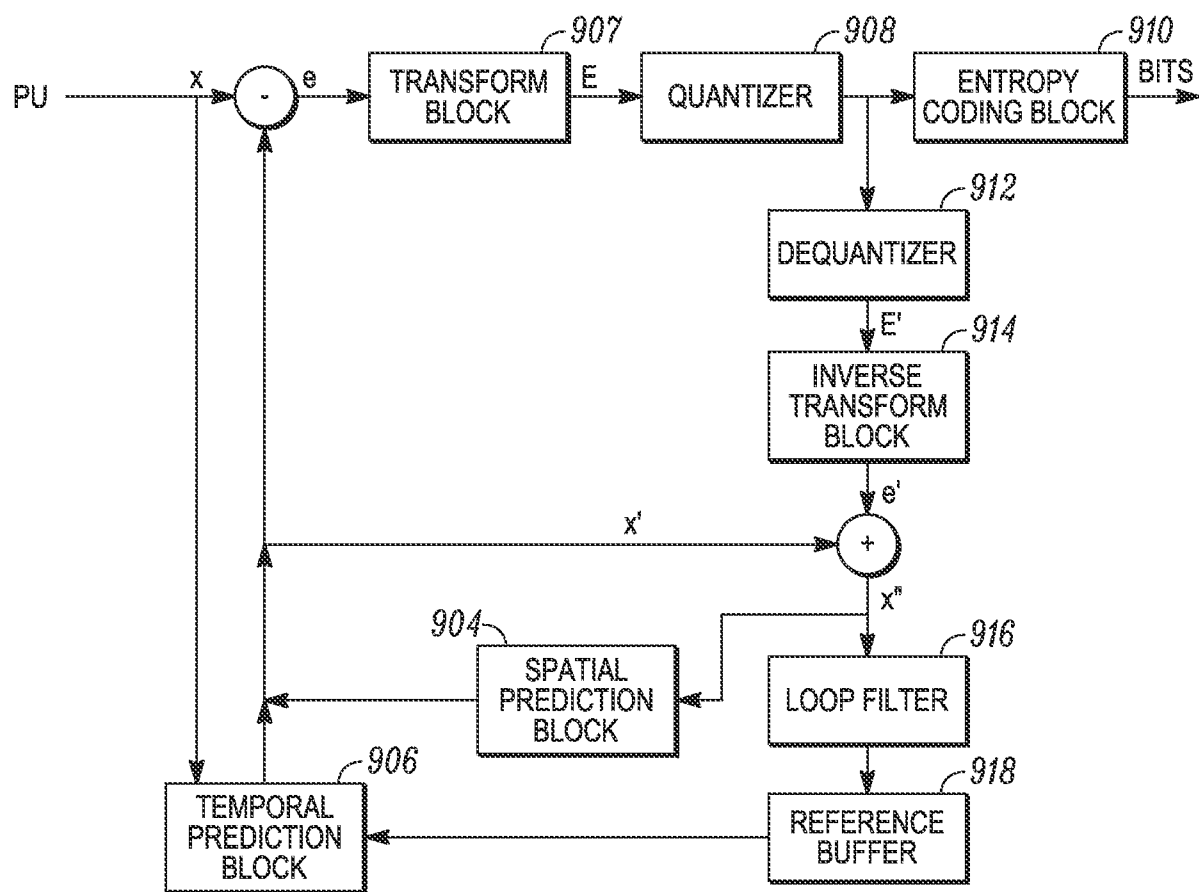


FIG. 9A

8/8

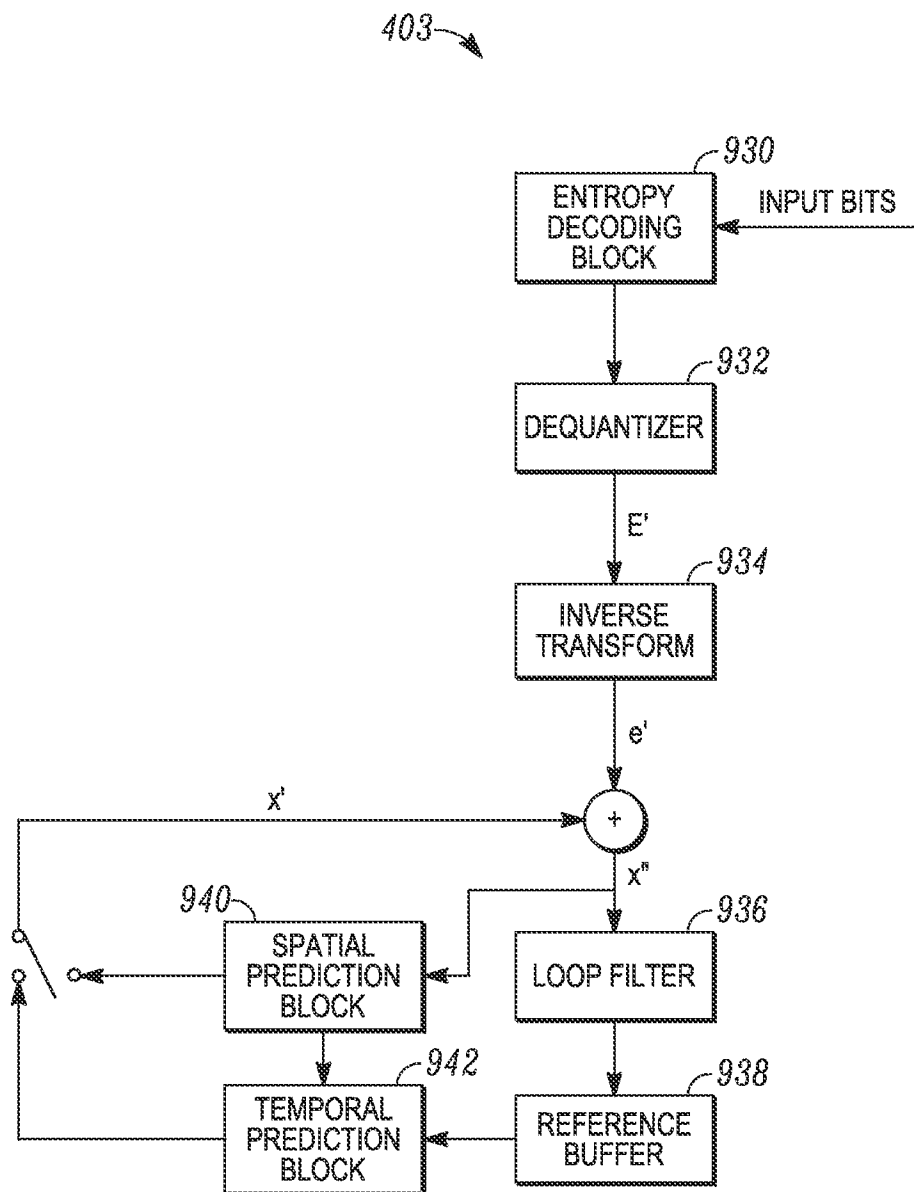


FIG. 9B

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2013/054965

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N7/36 H04N7/26
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04N

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

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

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	LIM C S ET AL: "High-level Syntax: Proposed fix on signalling of TMVP disabling flag", 9. JCT-VC MEETING; 100. MPEG MEETING; 27-4-2012 - 7-5-2012; GENEVA; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/, , no. JCTVC-I0420, 17 April 2012 (2012-04-17), XP030112183,	1,3-8,10
Y	section 3 appendix, section 7.3.3 appendix, section 7.4.3 ----- -/--	2,9



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

10 October 2013

Date of mailing of the international search report

22/10/2013

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

Regidor Arenales, R

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2013/054965

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	BROSS B ET AL: "High Efficiency Video Coding (HEVC) text specification draft 7", 9TH. MPEG MEETING; 27-4-2012 - 7-5-2012; GENEVA ; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11), no. JCTVC-I1003, 14 July 2012 (2012-07-14) , pages 37-79, XP002714357, page 67, line 23 - line 26 section 7.4.3, first paragraph -----	2,9
X,P	YU Y ET AL: "Modification of slice temporal mvp enable flag", 11. JCT-VC MEETING; 102. MPEG MEETING; 10-10-2012 - 19-10-2012; SHANGHAI; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-K0251, 2 October 2012 (2012-10-02), XP030113133, the whole document -----	1-10
A	LI (USTC) B ET AL: "High-level Syntax: Marking process for non-TMVP pictures", 7. JCT-VC MEETING; 98. MPEG MEETING; 21-11-2011 - 30-11-2011; GENEVA; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-G398, 8 November 2011 (2011-11-08), XP030110382, page 2 -----	1-10