

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

(19) World Intellectual Property

Organization

International Bureau

(43) International Publication Date

25 June 2020 (25.06.2020)



(10) International Publication Number

WO 2020/131232 A1

(51) International Patent Classification:

H04N 19/70 (2014.01) H04N 19/573 (2014.01)

H04N 19/105 (2014.01) H04N 19/172 (2014.01)

H04N 19/58 (2014.01)

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).

(21) International Application Number:

PCT/US2019/059429

Published:

— with international search report (Art. 21(3))

(22) International Filing Date:

01 November 2019 (01.11.2019)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

62/780,848 17 December 2018 (17.12.2018) US

62/786,771 31 December 2018 (31.12.2018) US

(71) Applicant: **APPLE INC.** [US/US]; One Apple Park Way, Cupertino, California 95014 (US).

(72) Inventors: **TOURAPIS, Alexandros Michael**; One Apple Park Way, Cupertino, California 95014 (US). **KIM, Jungsun**; One Apple Park Way, Cupertino, California 95014 (US). **RAPAKA, Krishnakanth**; One Apple Park Way, Cupertino, California 95014 (US). **SINGER, David William**; One Apple Park Way, Cupertino, California 95014 (US).

(74) Agent: **HAILS, Robert L.** et al.; 1050 Connecticut, NW, Suite 1100, Washington, District of Columbia 20036 (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, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV,

(54) Title: REFERENCE PICTURE MANAGEMENT AND LIST CONSTRUCTION

(57) Abstract: Techniques for encoding video with motion compensation include a compressed bitstream syntax that includes a list of all motion prediction reference frames without distinguishing between short-term reference frame and long-term reference frames. The list of reference frames may be provided in a slice header and may apply to encoded data video data within the corresponding slice. The list may be prefaced with a single number indicating the total number of reference frames. In an aspect delta POC reference numbers may be encoded with a flag indicating the sign of the delta POC when the absolute value of the POC is not equal to zero. In another aspect, a flag may be encoded for every reference frame indicating if POC information should be used when scaling prediction references, and a weighting parameter may be included when POC information should be used.



WO 2020/131232 A1

## **REFERENCE PICTURE MANAGEMENT AND LIST CONSTRUCTION**

### **CROSS REFERENCE TO RELATED APPLICATIONS**

- [01] This application claims the benefit of priority under 35 USC 119(e) of U.S. provisional application no. 62/780,848, filed on December 17, 2018, and U.S. provisional application no. 62/786,771, filed on December 31, 2018.

### **BACKGROUND**

- [02] The present disclosure is directed to video coding and decoding techniques and, in particular, management of lists of prediction reference frames.

- [03] Inter prediction is a key component of modern video codecs. This process permits the use of previously encoded images as prediction references for the currently encoded image and can considerably impact the coding efficiency of a video codec. Although, theoretically, one could potentially use all previously coded pictures as prediction references, that can have considerably implications in memory as well as signaling of which references to be used. This is because in such case, all such pictures would have to be retained in memory while a rather large index would have to be associated with them and referenced when specifying the references to be used for prediction for each image.

### **SUMMARY**

- [05] In an aspect, techniques for managing a decoded picture buffer (DPB) may include deriving, from a bitstream of a coded sequence of pictures, a list of all reference pictures that may be used in decoding a portion, such as a slice layer, of bitstream syntax. The bitstream syntax indicating the list of reference pictures may essentially consist of a single list of reference pictures including single number indicating the total number of all reference frames. The contents of a DPB may then be managed by evicting any decoded pictures from the DPB that are not in the list.

- [06] In further aspects of these techniques, reference frames in the single list may be identified by a difference in a picture order count (POC) between a current frame and an

identified reference frame. In a further aspect, the reference pictures may be defined by a communication syntax according to Table 3. In a further aspect, the bitstream syntax for the difference in POC may include a magnitude of the difference, and, if the magnitude is not zero, a flag indicating a sign of the difference. In a further aspect, the reference pictures may be defined by a communication syntax according to Table 2. In a further aspect, the bitstream syntax for the difference in POC may include: for the first reference frame in the list, a magnitude of the difference, and always includes a flag indicating a sign of the difference; and for subsequent reference frames in the list, a magnitude of the difference, and, only if the magnitude is not zero, a flag indicating a sign of the difference. In a further aspect, the reference pictures may be defined by a communication syntax according to Table 4. In a further aspect, the bitstream syntax for each entry in the list of reference frames may include a flag indicating if an absolute POC is used and an indicator of the reference frame, wherein: if the flag indicates an absolute reference is used, the indicator of the reference frame includes least significant bits of the POC of the reference frame; and if the flag indicates an absolute reference frame is not used, the indicator of the reference frame includes a difference in a (POC) between a current frame and the reference frame. In a further aspect, the reference pictures may be defined by a communication syntax according to Table 5.

[07] In further aspects of these techniques, the bitstream syntax for each entry in the list of reference frames includes a flag indicating if prediction of motion vectors from the respective reference picture are to be based on POC information or not based on POC information. In further aspect, the reference pictures may be defined by a communication syntax according to Table 6. In further aspect, the bitstream syntax for each entry in the list of reference frames may include a `consider_POC` flag indicating if prediction of motion vectors from the respective reference picture are to be based on POC information or not, an `absolute_ref` flag indicating if an absolute POC is used, and an indicator of the reference frame, wherein: if the `absolute_ref` flag indicates an absolute reference is used, the indicator of the reference frame includes least significant bits of the POC of the reference frame; if the `absolute_ref` flag indicates an absolute reference frame is not used, the indicator of the reference frame includes a difference in a (POC) between a current frame and the reference frame. In further aspect, the reference pictures may be defined by a communication syntax according to Table 6.5. In further aspect, if the `consider_POC` flag indicates prediction is based on POC information, the bitstream syntax for each entry may further include a weighting parameter. In further aspect, the reference pictures may be defined by a communication syntax according to Table 7. In further aspect, the list of

all reference pictures may be included in a slice header, and the portion of bitstream syntax in which the list of reference pictures may be used corresponds to a slice.

- [08] In an aspect a method may comprise: responsive to a bitstream of a coded sequence of pictures, deriving a list of all reference pictures used in decoding of a portion of bitstream syntax essentially consisting of a single list of reference pictures including single number indicating the total number of all reference frames; and evicting decoded pictures from a DPB that are not in the list; wherein reference frames in the single list are identified by a difference in a picture order count (POC) between a current frame and an identified reference frame.
- [09] In an aspect a method may comprise: responsive to a bitstream of a coded sequence of pictures, deriving a list of all reference pictures used in decoding of a portion of bitstream syntax essentially consisting of a single list of reference pictures including single number indicating the total number of all reference frames; and evicting decoded pictures from a DPB that are not in the list; wherein reference frames in the single list are identified by a difference in a picture order count (POC) between a current frame and an identified reference frame; wherein the bitstream syntax for the difference in POC includes a magnitude of the difference, and, if the magnitude is not zero, a flag indicating a sign of the difference.
- [10] In an aspect a method may comprise: responsive to a bitstream of a coded sequence of pictures, deriving a list of all reference pictures used in decoding of a portion of bitstream syntax essentially consisting of a single list of reference pictures including single number indicating the total number of all reference frames; and evicting decoded pictures from a DPB that are not in the list; wherein reference frames in the single list are identified by a difference in a picture order count (POC) between a current frame and an identified reference frame; wherein the bitstream syntax for the difference in POC includes, for the first reference frame in the list, a magnitude of the difference, and always includes a flag indicating a sign of the difference, and, for subsequent reference frames in the list, a magnitude of the difference, and, only if the magnitude is not zero, a flag indicating a sign of the difference.
- [11] In an aspect a method may comprise: responsive to a bitstream of a coded sequence of pictures, deriving a list of all reference pictures used in decoding of a portion of bitstream syntax essentially consisting of a single list of reference pictures including single number indicating the total number of all reference frames; and evicting decoded pictures from a DPB that are not in the list; wherein the bitstream syntax for each entry in the list of reference frames includes a flag indicating if an absolute POC is used and an indicator of the reference frame,

and wherein, if the flag indicates an absolute reference is used, the indicator of the reference frame includes least significant bits of the POC of the reference frame, and if the flag indicates an absolute reference frame is not used, the indicator of the reference frame includes a difference in a (POC) between a current frame and the reference frame.

[12] In an aspect a method may comprise: responsive to a bitstream of a coded sequence of pictures, deriving a list of all reference pictures used in decoding of a portion of bitstream syntax essentially consisting of a single list of reference pictures including single number indicating the total number of all reference frames; and evicting decoded pictures from a DPB that are not in the list; wherein the bitstream syntax for each entry in the list of reference frames includes a flag indicating if prediction of motion vectors from the respective reference picture are to be based on POC information or not based on POC information.

[13] In an aspect a method may comprise: responsive to a bitstream of a coded sequence of pictures, deriving a list of all reference pictures used in decoding of a portion of bitstream syntax essentially consisting of a single list of reference pictures including single number indicating the total number of all reference frames; and evicting decoded pictures from a DPB that are not in the list; wherein the bitstream syntax for each entry in the list of reference frames includes a consider\_POC flag indicating if prediction of motion vectors from the respective reference picture are to be based on POC information or not, an absolute\_ref flag indicating if an absolute POC is used, and an indicator of the reference frame, and wherein, if the absolute\_ref flag indicates an absolute reference is used, the indicator of the reference frame includes least significant bits of the POC of the reference frame, and if the absolute\_ref flag indicates an absolute reference frame is not used, the indicator of the reference frame includes a difference in a (POC) between a current frame and the reference frame.

[14] In an aspect a method may comprise: responsive to a bitstream of a coded sequence of pictures, deriving a list of all reference pictures used in decoding of a portion of bitstream syntax essentially consisting of a single list of reference pictures including single number indicating the total number of all reference frames; and evicting decoded pictures from a DPB that are not in the list; wherein the bitstream syntax for each entry in the list of reference frames includes a flag indicating if prediction of motion vectors from the respective reference picture are to be based on POC information or not based on POC information; wherein, if the consider\_POC flag indicates prediction is based on POC information, the bitstream syntax for each entry further includes a weighting parameter.

[15] **BRIEF DESCRIPTION OF THE DRAWINGS**

[16] FIG. 1 is a simplified block diagram of an example video coding system.

[17] FIG. 2 is a simplified block diagram of an example decoding system.

[18] FIG. 3 depicts a method for management of a decoded picture buffer (DPB).

**DETAILED DESCRIPTION**

[19] To avoid memory and efficiency issues associated with using all previously coded pictures as prediction references, modern video codecs such as MPEG-4 AVC/H.264, HEVC/H.265, VVC/H.266, and AV1 provide mechanisms that limit the number of the available references for each picture. There is commonly a Decoded Picture Buffer (DPB) of limited size (i.e. how many reference pictures it may maintain any given time) that is updated and maintained using well specified processes. Well defined processes are also used for the construction of the reference picture lists that are used for inter prediction for single prediction (e.g. a prediction based on a single reference) and biprediction (e.g. a prediction based on two references).

[20] In particular, in the AVC specification a well-defined process, based on a “sliding window” mechanism, that is also augmented by “explicitly” signaled memory management control operation (MMCO) instructions, is provided for the construction of the DPB. In this process pictures are essentially added in the DPB based on whether the pictures are signaled as references or not, and indicators (FrameNum) are associated with these pictures that relate to their coding order. Although commonly a sliding window (first in – first out) approach is used for controlling which pictures would remain in the DPB, based on the DPB size, the MMCO instructions may also override the sliding window operations by either specifying explicitly the removal of certain frames or by also tagging as certain frames as long-term reference frames. If a frame is tagged as a long-term reference frame, this frame will only be removed through either an explicit MMCO instruction specifying its removal, or through the signaling of an Instantaneous Decoder Refresh (IDR) frame. IDR frames specify that all frames in the DPB should be removed immediately. AVC specified that lists of reference pictures were constructed based on either a predefined process that used the FrameNum information (for P slices) and the Picture Order Count (POC) parameter associated with each reference picture to sort the pictures, or AVC used a process named as the “reference picture list modification” process that explicitly specified the order of the reference pictures in each list. A POC

parameter may be a number uniquely identifying each frame in a coded video sequence, and hence a POC parameter may also uniquely identify each reference frame.

[21] Techniques for encoding video with motion compensation include a compressed bitstream syntax indicating a current list of motion prediction reference frames without distinguishing between short-term reference frame and long-term reference frames. The list of reference frames may be provided, for example, in a slice header and may apply to encoded data video data within the corresponding slice. The list may be prefaced with a single number indicating the total number of reference frames. In an aspect the list may identify reference frame by an absolute POC number associated with each reference frame, or by a delta POC numbers indicating the difference between the absolute POC of a current frame and the absolute POC of each reference frame. In an aspect delta POC reference numbers may be encoded with a flag indicating the sign of the delta POC when the absolute value of the POC is not equal to zero. In an aspect a flag may be encoded for every reference frame indicating if POC information should be used when scaling prediction references, and a weighting parameter may be included when POC information should be used.

[22] In an aspect, the list of all motion prediction references may be used to manage a decoded picture buffer (DPB), for example, by evicting references previously stored in DPB that are no longer included in the current list of references. In another aspect, motion vectors may be predicted using POC information only when a flag for the reference frame indicates POC information should be used. In an aspect a DPB in an encoder may be managed to mimic a DPB in a decoder based on the list of reference frame provided by the encoder in a bitstream syntax that is transmitted to the decoder.

[23] FIG. 1 is a simplified block diagram of an example video coding system 100. The principles of the present disclosure find application in a video coding system that includes a video coder 120 and a video decoder. An embodiment of the video coder 120 is illustrated in FIG. 1 Here, the video coder 120 may receive video data from a video source 110 and code the video data to provide bandwidth compression. The video coder 120 is shown as including a forward coder 121, a decoder 122, a decoded picture buffer (DPB) 123, and a predictor 124. The DPB 123 may store decoded pictures of previously-coded and decoded reference frames that are available for use in inter prediction of a new input frame. The predictor 124 may compare content of new input frames that are to be coded to content of the reference frame(s) stored in the DPB 123, typically at sub-frame granularities called coding units, and may supply content of a prediction reference to the forward coder 121.

- [24] The forward coder 121 may code each coding unit of the new input frame according to a selected coding mode. When inter prediction is selected for coding, the forward coder may code the input coding unit differentially with respect to the prediction coding unit. The forward coder 121 may apply additional coding processes such as quantization and entropy coding. The forward coder 121 may output data representing the coded coding units to a transmitter 140 for transmission to another device.
- [25] When a new input frame is selected to serve as a reference frame, the decoder 122 inverts coding operations applied by the forward coder 121 and generates a decoded reference frame therefrom. The decoded reference frame may be stored in the DPB 123 for use as a candidate prediction reference for coding later-received input video.
- [26] The video coding may include a controller 125. The controller 125 may control which frames are selected to serve as reference frames and may manage the DPB 123 according to the processes described above. When DPB management decisions are made, the controller 125 may cause the transmitter 130 to transmit signaling according to any of the techniques described in the Tables 2 to 7.
- [27] FIG. 2 is a simplified block diagram of an example decoding system 200. Coded video data typically is transmitted to a video decoder (FIG. 2), where it is decoded and consumed. A receiver 210 may receive coded video data from a communication or computer network or from a storage device (not shown), and it may route the coded video data to a video decoder 220 for decoding.
- [28] The video decoder 220 may include a decoder 221, a predictor 222, a decoded picture buffer 223, and a controller 224. The predictor 222 may respond to identifiers of coding decisions applied to the coded video data to identify data from the decoded picture buffer 223 that is a prediction reference for each coded coding unit, and the predictor 222 may supply the reference data to the decoder 221. The decoder 221 may invert coding operations applied by the forward coder 121 (FIG. 1) of the video encoder 120. The decoder 221 may generate decoded frames of video therefrom, and it may output the decoded frames to a video sink device 230 (such as a display, a storage device or an application that processes such video) for consumption.
- [29] Decoded data of frames designated as reference frames may be stored in the decoded picture buffer 223. In the absence of transmission errors between the video coder 120 and the video



- [30] decoder 220 content of the DPBs 123, 223 would remain synchronized with each other.
- [31] The video decoder's controller 224 may perform DPB management operations as described in the foregoing, in response to signaling provided in the channel. Thus, when the controller 224 receives signaling, for example, as shown in any the foregoing Tables 2-7, it may perform corresponding management operations on its DPB 223.
- [32] FIG. 3 depicts a method for management of a decoded picture buffer (DPB). Terminal 302 may encode video, for example using the encoding system 100 of FIG. 1, and terminal 304 may decode video, for example using the decoding system 200 of FIG. 2. In the method of FIG. 3, terminal 302 repeatedly codes a slice of video (box 310) into a bitstream with a bitstream syntax that includes a reference picture list, and then transmits the bitstream of the slice to terminal 304 (message 312). Terminal 304 then decodes the slice, including extracting the reference picture list from the slice bitstream (box 314). If the state of a DPB in terminal 304 does not agree with the reference picture list (box), reference pictures not in the current list may be evicted (box 318), otherwise control may flow to box 320. These methods may be repeated for subsequent slices in both terminals 302 and 304.
- [33] In optional box 320, image data may be predicted based on the extracted current reference picture list. For example, image data may be predicted from one or more pictures in the reference picture list based on motion vectors extracted from the bitstream. Predicted image data may then form the basis for decoded frames, as described above regarding decoder 200 of FIG.2 above, and the decoded frames may be stored for future use as new reference frames in the DPB. The new reference frames themselves may also be evicted from the DPB, for example, based on a later reference picture list specified later in the bitstream.
- [34] The AVC design, as explained in the background section above, had the major flaw that it required knowledge of how reference frames were handled in the past to properly maintain the DPB. Any operations such as splicing, channel change, and editing could result in loss of such information, and therefore impair the decoding process. Therefore, in HEVC a different approach was used where a complete set of the reference pictures potentially used by the current picture as well as any subsequent picture is signaled. This is referred to as the reference picture set (RPS) process. Doing so guarantees that every time the decoder knows which pictures may be utilized for prediction. The HEVC design also includes a list initialization process as well as an optional picture list modification process to generate the prediction lists.

[35] Although the HEVC approach avoided the problems of the AVC design, there were complaints that the process was too complex and difficult to understand. Therefore, for the upcoming VVC/H.266 video coding standard it was proposed that a new approach, that unifies the RPS process and the reference list construction process should be adopted. In particular, it was proposed that instead for signaling the RPS followed by the optional reference list modification process, a reference picture list structure (RPLS) is signaled instead that combines the indication of which references should be in the DPB with the construction of an initial representation of the two reference lists. Essentially, this specifies all reference entries that should remain in the DPB, in the appropriate order for each list. The process is then followed by an indication of how many references should be retained for each list in ascending order. In these systems, short-term and long-term references were still retained, while the number of references that could be used was restricted to the size of the DPB. The signaling of the reference picture list structure for this method is shown below, which includes the number of short-term and long-term references signaled, whether a picture is a long-term picture or not, and information to identify them in the bitstream. Such information can be relative (delta POCs) compared to the current reference for short-term reference pictures, or absolute for long-term reference pictures.

	Descriptor
ref_pic_list_struct( listIdx, rplsIdx, ltrpFlag ) {	
num_strp_entries[ listIdx ][ rplsIdx ]	ue(v)
if( ltrpFlag )	
num_ltrp_entries[ listIdx ][ rplsIdx ]	ue(v)
for( i = 0; i < NumEntriesInList[ listIdx ][ rplsIdx ]; i++ ) {	
if( num_ltrp_entries[ listIdx ][ rplsIdx ] > 0 )	
lt_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]	u(1)
if( !lt_ref_pic_flag[ listIdx ][ rplsIdx ][ i ] ) {	
delta_poc_st[ listIdx ][ rplsIdx ][ i ]	ue(v)
if( delta_poc_st[ listIdx ][ rplsIdx ][ i ] > 0 )	
strp_entry_sign_flag[ listIdx ][ rplsIdx ][ i ]	u(1)
} else	
poc_lsb_lt[ listIdx ][ rplsIdx ][ i ]	u(v)
}	
}	
}	

Table 1

[36] In this disclosure, the methodology proposed to the VVC specification is improved by, first, removing the concept of long-term references. In particular, there may be limited value

in separating the references in short-term and long-term references. The proposed method already has the ability to indicate which frames should be retained for the prediction of the current or potentially future pictures, and the classification of whether a picture is a short-term or a long-term does not seem to provide any benefits in this context. Instead, the encoder can signal which past pictures should be used as reference and make no explicit identification of short-term or long-term references. In an example, this could be done with the following syntax:

ref_pic_list_struct( listIdx, rplsIdx) {	<b>Descriptor</b>
num_ref_entries[ listIdx ][ rplsIdx ]	ue(v)
for( i = 0; i < num_ref_entries[ listIdx ][ rplsIdx ]; i++) {	
abs_delta_poc_ref[ listIdx ][ rplsIdx ][ i ]	ue(v)
if(abs_delta_poc_ref[ listIdx ][ rplsIdx ][ i ] > 0 )	
sign_delta_poc_ref[ listIdx ][ rplsIdx ][ i ]	u(1)
}	
}	

**Table 2**

Or:

ref_pic_list_struct( listIdx, rplsIdx) {	<b>Descriptor</b>
num_ref_entries[ listIdx ][ rplsIdx ]	ue(v)
for( i = 0; i < num_ref_entries[ listIdx ][ rplsIdx ]; i++) {	
delta_poc_ref[ listIdx ][ rplsIdx ][ i ]	se(v)
}	
}	

**Table 3**

[37] In these examples, the relative distances in terms of POC between the current picture and the references are indicated. For index  $i = 0$ , the distance between the current picture and the first reference is used. For indices  $i > 0$ , the distances versus the current picture can be computed, or, in an alternative embodiment, the POC distance between the reference of index  $i$  and reference of index  $i - 1$ . If the codec does not permit prediction of the current picture using itself, i.e. availability of the “current picture reference” method, the value of 0 for the reference with index  $i = 0$  would not be possible. In that case, the first case can be modified to the following:

ref_pic_list_struct( listIdx, rplsIdx) {	<b>Descriptor</b>
<b>num_ref_entries</b> [ listIdx ][ rplsIdx ]	ue(v)
if(num_ref_entries[ listIdx ][ rplsIdx ] > 0) {	
<b>abs_delta_poc_ref_minus1</b> [ listIdx ][ rplsIdx ][ 0 ]	ue(v)
<b>sign_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ 0 ]	u(1)
}	
for( i = 1; i < num_ref_entries[ listIdx ][ rplsIdx ]; i++) {	
<b>abs_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	ue(v)
if(abs_delta_poc_ref[ listIdx ][ rplsIdx ][ i ] > 0 )	
<b>sign_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	u(1)
}	
}	

**Table 4**

[38] Another alternative would be to permit indicating instead of the delta POC for some pictures their absolute POC number. This could be done as follows:

ref_pic_list_struct( listIdx, rplsIdx) {	<b>Descriptor</b>
<b>num_ref_entries</b> [ listIdx ][ rplsIdx ]	ue(v)
for( i = 0; i < num_ref_entries[ listIdx ][ rplsIdx ]; i++) {	
<b>absolute_ref_pic_flag</b> [ listIdx ][ rplsIdx ][ i ]	u(1)
if( !absolute_ref_pic_flag[ listIdx ][ rplsIdx ][ i ] ) {	
<b>abs_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	ue(v)
if(abs_delta_poc_ref[ listIdx ][ rplsIdx ][ i ] > 0 )	
<b>sign_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	u(1)
} else	
<b>poc_lsb_ref</b> [ listIdx ][ rplsIdx ][ i ]	u(v)
}	
}	

**Table 5**

[39] Note that poc\_lsb\_ref needs not be of the same size as the slice\_pic\_order\_cnt\_lsb that is used to signal the picture order count modulo MaxPicOrderCntLsb of the current picture. It could also be larger and its size could be indicated with an additional parameter that is signaled at the sequence or picture parameter set level. That can allow signaling of references that are further away from the current picture than a distance of MaxPicOrderCntLsb, similar to what is done for long-term references. However, those references may still not be considered long-term

references. Alternatively, the encoder can optionally also signal an additional parameter either for each reference or for the entire reference group that would specify an additional fixed distance for those references versus the current reference. Absolute position references could be considered in the delta distance computation or could be excluded from it.

- [40] Long-term references in AVC and HEVC also impact the computation of weighted prediction (AVC) and motion vector prediction (AVC and HEVC). This is because for long-term references it is assumed that using the temporal relationship between pictures, hinted by POC parameters, may be unreliable. Since in the proposed system, it is not explicitly indicated whether a picture is long-term or short-term, that capability may be lost. Instead, and if such capability is retained, a flag can be indicated that would specify whether the POC information for a particular reference should be used for such computations or not. This could be done, for example as follows, by introducing a new flag, such as, for example, `consider_poc_for_derivations_flag`. This new flag could be signaled for every reference, could be signaled equally for all references, or could also be signaled only for certain types of references, e.g. for the references that signaled in AVC or HEVC with the absolute POC lsb. An example of signaling this is as follows:

	Descriptor
<code>ref_pic_list_struct( listIdx, rplsIdx) {</code>	
<b>num_ref_entries</b> [ listIdx ][ rplsIdx ]	ue(v)
for( i = 0; i < num_ref_entries[ listIdx ][ rplsIdx ]; i++) {	
<b>consider_poc_for_derivations_flag</b> [ listIdx ][ rplsIdx ][ i ]	u(1)
<b>abs_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	ue(v)
if(abs_delta_poc_ref[ listIdx ][ rplsIdx ][ i ] > 0 )	
<b>sign_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	u(1)
}	
}	

**Table 6**

- [41] The bitstream syntax embodiments of Tables 5 and 6 can be combined to provide flags indicating absolute POC or delta POC, and for indicating if reference frame POC are to be used as hints for motion vector prediction:

	<b>Descriptor</b>
ref_pic_list_struct( listIdx, rplsIdx ) {	
<b>num_ref_entries</b> [ listIdx ][ rplsIdx ]	ue(v)
for( i = 0; i < num_ref_entries[ listIdx ][ rplsIdx ]; i++) {	
<b>consider_poc_for_derivations_flag</b> [ listIdx ][ rplsIdx ][ i ]	u(1)
if( !absolute_ref_pic_flag[ listIdx ][ rplsIdx ][ i ] ) {	
<b>abs_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	ue(v)
if( abs_delta_poc_ref[ listIdx ][ rplsIdx ][ i ] > 0 )	
<b>sign_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	u(1)
} else	
<b>poc_lsb_lt</b> [ listIdx ][ rplsIdx ][ i ]	u(v)
}	
}	
}	

**Table 6.5**

[42] In an alternative embodiment, the POC may be associated only with the construction of the DPB and signal an additional parameter that will dictate the scaling parameter that should be used during prediction and decoding operations (such as for scaling motion vectors). Such scaling parameter could be specified during the decoding of the current picture and could persist through the entire presence of this picture in the DPB but could also be signaled explicitly and dynamically inside the ref pic list struct. The value that should be used for the current picture, when combined with the references in this list, similar to what is done now for POC, can then also be signaled (current pic\_weight[ listIdx ] below). By using this method different scaling of any related parameters could be achieved for different pictures or even subgroups. All these parameters and their presence could also be controlled through parameters at higher level syntax elements.

ref_pic_list_struct( listIdx, rplsIdx) {	<b>Descriptor</b>
<b>num_ref_entries</b> [ listIdx ][ rplsIdx ]	ue(v)
refDerivationCount = 0	
for( i = 0; i < num_ref_entries[ listIdx ][ rplsIdx ]; i++) {	
<b>consider_ref_for_derivations_flag</b> [ listIdx ][ rplsIdx ][ i ]	u(1)
<b>abs_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	ue(v)
if(abs_delta_poc_ref[ listIdx ][ rplsIdx ][ i ] > 0 )	
<b>sign_delta_poc_ref</b> [ listIdx ][ rplsIdx ][ i ]	u(1)
if(consider_ref_for_derivations_flag[ listIdx ][ rplsIdx ][ i ] == TRUE) {	
<b>ref_weight_param</b> [ listIdx ][ rplsIdx ][ i ]	se(v)
refDerivationCount ++	
}	
}	
if (refDerivationCount > 0 )	
<b>current_pic_weight</b> [ listIdx ]	se(v)
}	

**Table 7**

[43] The above structures permit replication of references, especially if such replication is to be used to allow multi-hypothesis prediction using the same reference, including multi-hypothesis prediction with more than two (2) defined reference lists, or for weighted prediction purposes. For weighted prediction, the same “physical reference” may be associated with different weighted prediction parameters, but the codec only permits weighted prediction parameter signaling at the slice or higher level and not at the block level (as in AVC and HEVC). In that case, commonly the same reference is specified multiple times in a reference list, and each instance of that reference is associated with a different weight. HEVC, for example, permits up to 15 such references for each list. However, such replicated references do not require any additional memory space and therefore should not be considered as taking up space in the DPB. The same is also true for references that may be present in multiple lists. Only “unique” references, and essentially the union of all unique references across all prediction lists should be considered for determining the pictures that should be retained in the DPB.

[44] Codecs like HEVC and AVC utilize also parallelization “grouping” concepts of blocks such as tiles or slices. For these groups it is important that the complete DPB and reference list information are also available. In AVC and HEVC it was mandated that the DPB information

was identical across all slices, and this was handled appropriately by replicating the needed information for each codec across all slices (i.e. MMCO + sliding window operations in AVC and RPS process in HEVC). Similar considerations need to be made in VVC. However here there are the following options:

- a) Repeat exactly the same RPLS for all slices/tiles/groups. This will guarantee that the DPB can be constructed identically for all such groups, however it has the limitation that the construction of the reference lists is less flexible and likely less efficient. This could still be combined with different weighted prediction parameters since those could be signaled with a different set. However, weighted parameters could also be restricted to be the same.
- b) Permit the RPLS to be different in terms of picture ordering and replication of references, however, guarantee that the same pictures are present at least once in the RPLS for every slice/tile/group, even if not used for prediction, so as to guarantee that the DPB is constructed the same for all such groups.
- c) Allow the RPLS to differ for each group. In this case, however, one should consider the “union” of all RPLS to construct the DPB. This can provide coding benefits, but requires parsing of all RPLS in all groups and may complicate also the design of parallel decoding systems or systems that may wish to perform partial image decoding based on sub-groups (e.g. using motion constraint tiled principles used for example in 360 degree applications).
- d) Signal the full DPB in the first RPLS (first group/tile/slice), while permitting all other groups to use a reduced RPLS version with only the necessary pictures for performing prediction in the current group. The benefit of this would be that a system that needs to know which references are used would only need to parse the first group while ignoring the other groups. This saves bits, however, it may be more error prone and would have the same issues as case (c) in terms of parallel decoding. In this case, if so desired, the full DPB could be signaled optionally in other groups as well but will not be enforced. Enforcement however could also be done by application or usage, and there could even be signals in the bitstream that say that such enforcement is used or not.

[45] The above concepts could be used on their own, in combination with other mechanisms for managing DPB and references, including sliding window or some other predefined



mechanism that specifies how pictures are added and removed from the DPB, signaling of reference lists/DPB at sequence or picture parameter sets, use of memory management operations for some pictures and so on.

[46] As discussed above, FIGS. 1 and 2 illustrate functional block diagrams of terminals. In implementation, the terminals may be embodied as hardware systems, in which case, the illustrated blocks may correspond to circuit sub-systems. Alternatively, the terminals may be embodied as software systems, in which case, the blocks illustrated may correspond to program modules within software programs executed by a computer processor. In yet another embodiment, the terminals may be hybrid systems involving both hardware circuit systems and software programs. Moreover, not all of the functional blocks described herein need be provided or need be provided as separate units. For example, although FIG. 1 illustrates the components of an exemplary encoder, including components such as the forward coder 121 and decoder 122, as separate units. In one or more embodiments, some components may be integrated. Such implementation details are immaterial to the operation of the present invention unless otherwise noted above. Similarly, the encoding, decoding and buffer management operations described with relation to FIG. 3 may be performed continuously as data is input into the encoder / decoder. The order of the steps as described above does not limit the order of operations.

[47] Some embodiments may be implemented, for example, using a non-transitory computer-readable storage medium or article which may store an instruction or a set of instructions that, if executed by a processor, may cause the processor to perform a method in accordance with the disclosed embodiments. The exemplary methods and computer program instructions may be embodied on a non-transitory machine-readable storage medium. In addition, a server or database server may include machine readable media configured to store machine executable program instructions. The features of the embodiments of the present invention may be implemented in hardware, software, firmware, or a combination thereof and utilized in systems, subsystems, components or subcomponents thereof. The “machine readable storage media” may include any medium that can store information. Examples of a machine-readable storage medium include electronic circuits, semiconductor memory device, ROM, flash memory, erasable ROM (EROM), floppy diskette, CD-ROM, optical disk, hard disk, fiber optic medium, or any electromagnetic or optical storage device.

[48] While the invention has been described in detail above with reference to some embodiments, variations within the scope and spirit of the invention will be apparent to those

of ordinary skill in the art. Thus, the invention should be considered as limited only by the scope of the appended claims.

**WE CLAIM:**

1. A method for managing a decoded picture buffer (DPB) comprising:  
responsive to a bitstream of a coded sequence of pictures, deriving a list of all reference pictures used in decoding of a portion of bitstream syntax essentially consisting of a single list of reference pictures including single number indicating the total number of all reference frames; and  
evicting decoded pictures from a DPB that are not in the list.
2. The method of claim 1, wherein reference frames in the single list are identified by a difference in a picture order count (POC) between a current frame and an identified reference frame.
3. The method of claim 2, wherein the reference pictures are defined by a communication syntax according to Table 3.
4. The method of claim 2, wherein the bitstream syntax for the difference in POC includes a magnitude of the difference, and, if the magnitude is not zero, a flag indicating a sign of the difference.
5. The method of claim 4, wherein the reference pictures are defined by a communication syntax according to Table 2.
6. The method of claim 2, wherein the bitstream syntax for the difference in POC includes:  
for the first reference frame in the list, a magnitude of the difference, and always includes a flag indicating a sign of the difference; and  
for subsequent reference frames in the list, a magnitude of the difference, and, only if the magnitude is not zero, a flag indicating a sign of the difference.
7. The method of claim 6, wherein the reference pictures are defined by a communication syntax according to Table 4.
8. The method of claim 1, wherein the bitstream syntax for each entry in the list of reference frames includes a flag indicating if an absolute POC is used and an indicator of the reference frame, wherein:  
if the flag indicates an absolute reference is used, the indicator of the reference frame includes least significant bits of the POC of the reference frame; and

if the flag indicates an absolute reference frame is not used, the indicator of the reference frame includes a difference in a (POC) between a current frame and the reference frame.

9. The method of claim 8, wherein the reference pictures are defined by a communication syntax according to Table 5.

10. The method of claim 1, wherein the bitstream syntax for each entry in the list of reference frames includes a flag indicating if prediction of motion vectors from the respective reference picture are to be based on POC information or not based on POC information.

11. The method of claim 10, wherein the reference pictures are defined by a communication syntax according to Table 6.

12. The method of claim 1, wherein the bitstream syntax for each entry in the list of reference frames includes a `consider_POC` flag indicating if prediction of motion vectors from the respective reference picture are to be based on POC information or not, an `absolute_ref` flag indicating if an absolute POC is used, and an indicator of the reference frame, wherein:

if the `absolute_ref` flag indicates an absolute reference is used, the indicator of the reference frame includes least significant bits of the POC of the reference frame;

if the `absolute_ref` flag indicates an absolute reference frame is not used, the indicator of the reference frame includes a difference in a (POC) between a current frame and the reference frame.

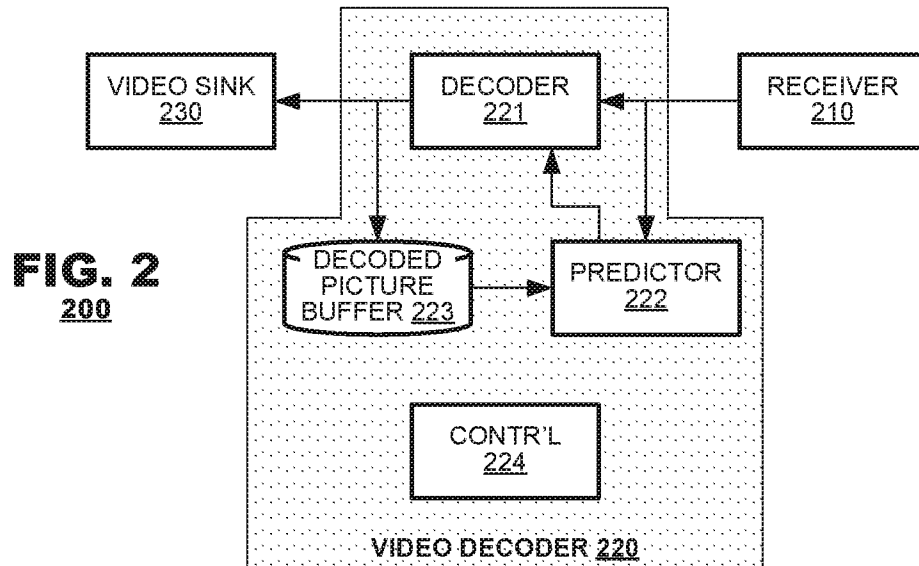
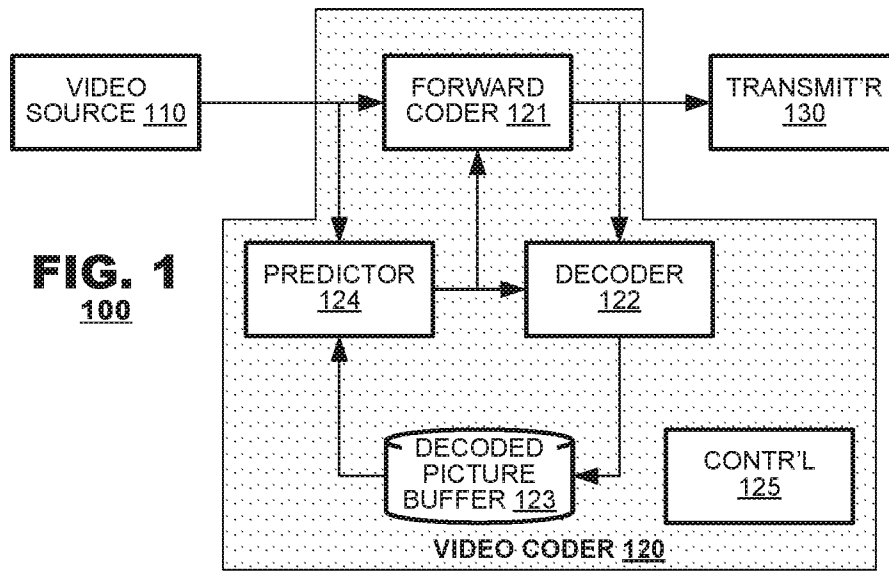
13. The method of claim 12, wherein the reference pictures are defined by a communication syntax according to Table 6.5.

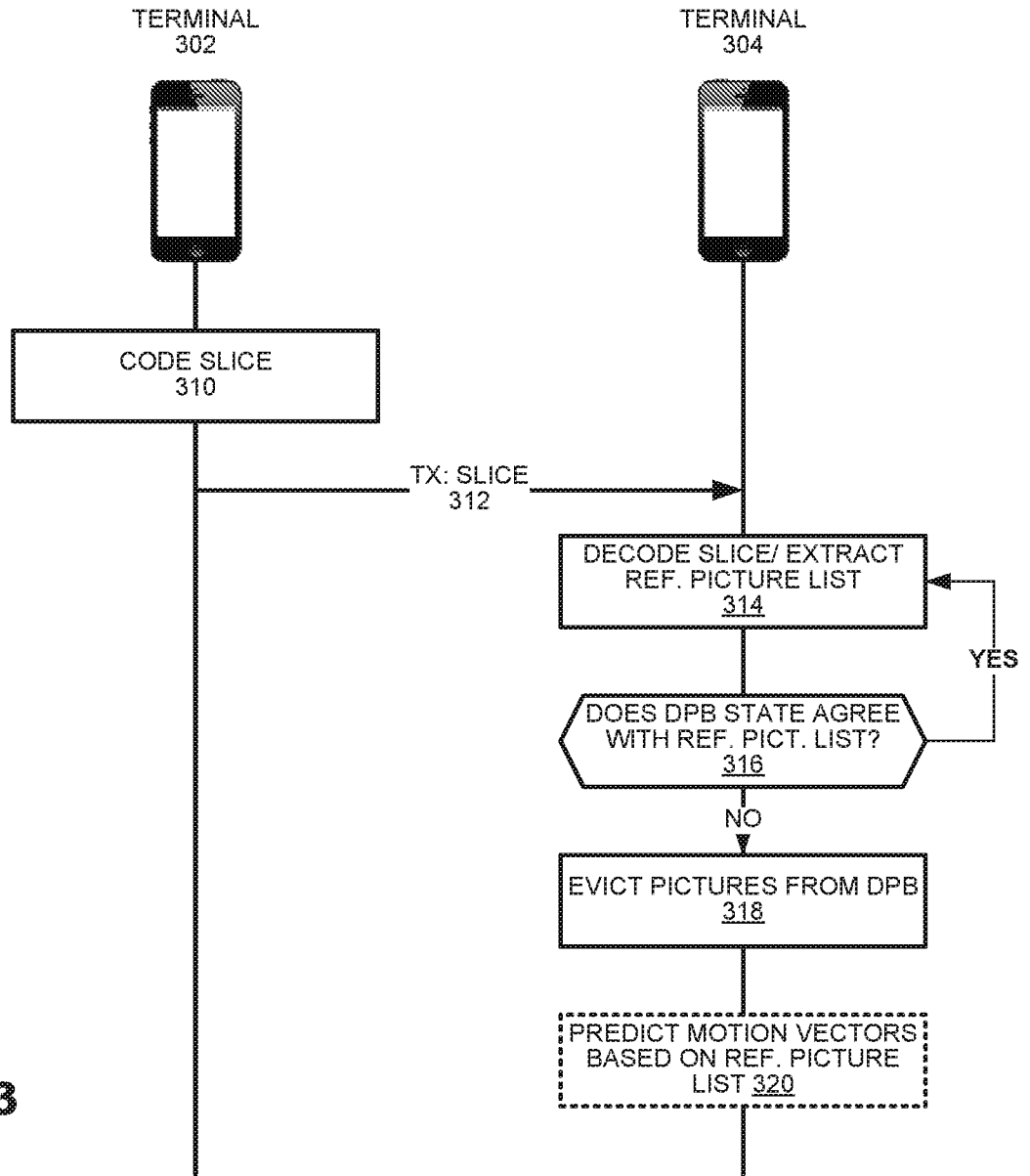
14. The method of claim 10, wherein, if the `consider_POC` flag indicates prediction is based on POC information, the bitstream syntax for each entry further includes a weighting parameter.

15. The method of claim 14, wherein the reference pictures are defined by a communication syntax according to Table 7.

16. The method of claims 1-15, wherein the list of all reference pictures is included in a slice header, and the portion of bitstream syntax in which the list of reference pictures is used corresponds to a slice.

17. A computer readable medium containing instructions that, when executed on a processor, cause the method of claims 1-16 to be performed by the processor.
18. A video decoder comprising:
  - a decoded picture buffer (DPB);
  - a decoder for decoding compressed video and storing reference pictures in the DPB;
  - a predictor for predicting from reference pictures in the DPB; and
  - a controller for performing the method of claims 1-16.





**FIG. 3**  
300

INTERNATIONAL SEARCH REPORT

International application No  
PCT/US2019/059429

A. CLASSIFICATION OF SUBJECT MATTER  
INV. H04N19/70 H04N19/105 H04N19/58 H04N19/573 H04N19/172  
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, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X Y A	US 2012/189058 A1 (CHEN PEISONG [US] ET AL) 26 July 2012 (2012-07-26) abstract paragraph [0007] paragraph [0009] - paragraph [0012] paragraph [0024] - paragraph [0026] paragraph [0030] -----	1,8,9, 16-18 2-5 6,7
X Y A	US 2012/250765 A1 (WU ZHUANGFEI [SE] ET AL) 4 October 2012 (2012-10-04) abstract figure 1 paragraph [0038] - paragraph [0039] paragraph [0041] ----- -/--	1,8,9, 17,18 2-5 6,7

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  22 January 2020	Date of mailing of the international search report  06/04/2020
--	--

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  Wedi, Thomas
--	--



## INTERNATIONAL SEARCH REPORT

International application No  
PCT/US2019/059429

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	HANNUKSELA M M ET AL: "AHG21: Removal of reference picture list modification", 8. JCT-VC MEETING; 99. MPEG MEETING; 1-2-2012 - 10-2-2012; SAN JOSE; (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-H0426, 20 January 2012 (2012-01-20), XP030111453, section 2  -----	2-5
Y	US 2013/101034 A1 (WAHADANIAH VIKTOR [SG] ET AL) 25 April 2013 (2013-04-25) abstract figure 13  -----	2-5
Y	BROSS B ET AL: "Versatile Video Coding (Draft 3)", 12. JVET MEETING; 20181003 - 20181012; MACAO; (THE JOINT VIDEO EXPLORATION TEAM OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16 ), , no. JVET-L1001 14 December 2018 (2018-12-14), XP030198630, Retrieved from the Internet: URL:http://phenix.int-evry.fr/jvet/doc_end_user/documents/12_Macao/wg11/JVET-L1001-v10.zip JVET-L1001-v6.docx [retrieved on 2018-12-14] page 29 page 38  -----	4,5
X,P	Y-K WANG (HUAWEI) ET AL: "AHG17: On reference picture management for VVC", 13. JVET MEETING; 20190109 - 20190118; MARRAKECH; (THE JOINT VIDEO EXPLORATION TEAM OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16 ), , no. JVET-M0128 2 January 2019 (2019-01-02), XP030200192, Retrieved from the Internet: URL:http://phenix.int-evry.fr/jvet/doc_end_user/documents/13_Marrakech/wg11/JVET-M0128-v1.zip JVET-M0128-v1_SpecTextNoLTRP.docx [retrieved on 2019-01-02] section 7.3.4  -----	1,8,9, 17,18

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No  
PCT/US2019/059429

Patent document cited in search report	Publication date	Patent family member(s)	Publication date	
US 2012189058	A1	26-07-2012	BR 112013018734 A2	25-10-2016
			CN 103339936 A	02-10-2013
			EP 2668781 A1	04-12-2013
			JP 5784752 B2	24-09-2015
			JP 2014506084 A	06-03-2014
			JP 2015188231 A	29-10-2015
			KR 20130118381 A	29-10-2013
			US 2012189058 A1	26-07-2012
			WO 2012102973 A1	02-08-2012
-----				
US 2012250765	A1	04-10-2012	BR 112012015001 A2	29-11-2016
			CA 2784515 A1	23-06-2011
			CA 2784517 A1	23-06-2011
			CN 102656890 A	05-09-2012
			CN 102668561 A	12-09-2012
			EP 2514209 A1	24-10-2012
			EP 2514210 A1	24-10-2012
			JP 5855013 B2	09-02-2016
			JP 2013514718 A	25-04-2013
			JP 2013514719 A	25-04-2013
			RU 2012130147 A	20-03-2014
			US 2012250765 A1	04-10-2012
			US 2012250766 A1	04-10-2012
			US 2015264393 A1	17-09-2015
			WO 2011075071 A1	23-06-2011
WO 2011075072 A1	23-06-2011			
-----				
US 2013101034	A1	25-04-2013	CA 2850598 A1	25-04-2013
			CN 104025598 A	03-09-2014
			CN 107770548 A	06-03-2018
			EP 2770734 A1	27-08-2014
			JP 5995111 B2	21-09-2016
			JP 6183630 B2	23-08-2017
			JP 2016220237 A	22-12-2016
			JP WO2013057884 A1	02-04-2015
			KR 20140085442 A	07-07-2014
			RU 2014113564 A	27-11-2015
			US 2013101034 A1	25-04-2013
			US 2017041634 A1	09-02-2017
US 2018199059 A1	12-07-2018			
US 2019132610 A1	02-05-2019			
WO 2013057884 A1	25-04-2013			
-----				

# INTERNATIONAL SEARCH REPORT

International application No.  
PCT/US2019/059429

## Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1.  Claims Nos.:  
because they relate to subject matter not required to be searched by this Authority, namely:
  
2.  Claims Nos.:  
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
  
3.  Claims Nos.:  
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

## Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

1.  As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
  
2.  As all searchable claims could be searched without effort justifying an additional fees, this Authority did not invite payment of additional fees.
  
3.  As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
  
4.  No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:  
  
1-9, 16-18

### Remark on Protest

- The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- No protest accompanied the payment of additional search fees.

**FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210**

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

1. claims: 1-9, 16-18

Identifying reference frames in the list

---

2. claims: 10-15

Signaling usage for prediction

---