



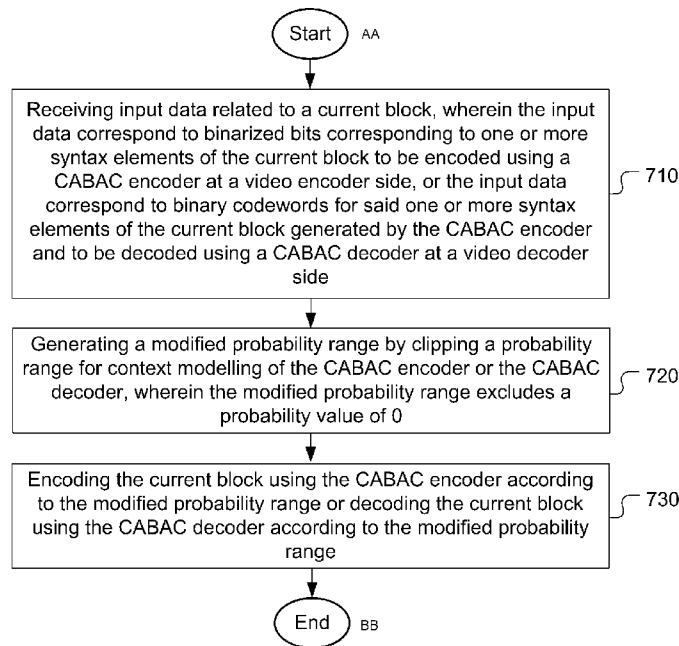
- (51) International Patent Classification:  
*H04N 19/21* (2014.01)
- (21) International Application Number:  
PCT/CN2024/097006
- (22) International Filing Date:  
03 June 2024 (03.06.2024)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:  
63/506197 05 June 2023 (05.06.2023) US
- (71) Applicant: **MEDIATEK INC** [CN/CN]; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN).
- (72) Inventors: **CHEN, Yi-Wen**; 2840 Junction Ave, San Jose, California 95134 (US). **CHUANG, Tzu-Der**; No. 1, Dus-

ing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **CHEN, Ching-Yeh**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN). **HSU, Chih-Wei**; No. 1, Dusing 1st Rd., Hsinchu Science Park, Hsinchu City, Taiwan 30078 (CN).

(74) Agent: **BEIJING SANYOU INTELLECTUAL PROPERTY AGENCY LTD.**; 16th Fl., Block A, Corporate Square, No.35 Jinrong Street, Xicheng District, Beijing 100033 (CN).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO,

(54) Title: METHODS AND APPARATUS OF ADAPTIVE PROBABILITY RANGE FOR CONTEXT MODELS OF CONTEXT ADAPTIVE BINARY ARITHMETIC CODING IN VIDEO CODING



**Fig. 7**

(57) Abstract: A method and apparatus for modifying initial probability range for context modelling associated with CABAC. According to this method, input data of a current block corresponding to binarized bits or binary codewords are received. A modified probability range is generated by clipping a beginning probability or both the beginning probability and an ending probability associated with an initial probability range for context modelling of the CABAC encoder or the CABAC decoder, where the modified probability range excludes a probability value of 0. The current block is encoded using the CABAC encoder according to the modified probability range or the current block is decoded using the CABAC decoder according to the modified probability range.



RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH,  
TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS,  
ZA, ZM, ZW.

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

**Published:**

- *with international search report (Art. 21(3))*

**METHODS AND APPARATUS OF ADAPTIVE PROBABILITY RANGE FOR  
CONTEXT MODELS OF CONTEXT ADAPTIVE BINARY ARITHMETIC CODING  
IN VIDEO CODING**

5 **CROSS REFERENCE TO RELATED APPLICATIONS**

[0001] The present invention is a non-Provisional Application of and claims priority to U.S. Provisional Patent Application No. 63/506,197, filed on June 5, 2023. The U.S. Provisional Patent Application is hereby incorporated by reference in its entirety.

10 **FIELD OF THE INVENTION**

[0002] The present invention relates to CABAC (Context-Adaptive Binary Arithmetic Coding) of syntax elements in a video coding system. In particular, the present invention relates to modifying an initial probability range associated with the context modelling for CABAC in a video coding system.

15

**BACKGROUND**

[0003] Versatile video coding (VVC) is the latest international video coding standard developed by the Joint Video Experts Team (JVET) of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG). The standard has been published as an ISO standard: ISO/IEC 23090-3:2021, Information technology - Coded representation of immersive media - Part 3: Versatile video coding, published Feb. 2021. VVC is developed based on its predecessor HEVC (High Efficiency Video Coding) by adding more coding tools to improve coding efficiency and also to handle various types of video sources including 3-dimensional (3D) video signals.

25 [0004] Fig. 1A illustrates an exemplary adaptive Inter/Intra video encoding system incorporating loop processing. For Intra Prediction 110, the prediction data is derived based on previously coded video data in the current picture. For Inter Prediction 112, Motion Estimation (ME) is performed at the encoder side and Motion Compensation (MC) is performed based on the result of ME to provide prediction data derived from other picture(s) and motion data.  
30 Switch 114 selects Intra Prediction 110 or Inter Prediction 112 and the selected prediction data is supplied to Adder 116 to form prediction errors, also called residues. The prediction error is then processed by Transform (T) 118 followed by Quantization (Q) 120. The transformed and quantized residues are then coded by Entropy Encoder 122 to be included in a video bitstream corresponding to the compressed video data. The bitstream associated with the transform  
35 coefficients is then packed with side information such as motion and coding modes associated

with Intra prediction and Inter prediction, and other information such as parameters associated with loop filters applied to underlying image area. The side information associated with Intra Prediction 110, Inter prediction 112 and in-loop filter 130, is provided to Entropy Encoder 122 as shown in Fig. 1A. When an Inter-prediction mode is used, a reference picture or pictures have to be reconstructed at the encoder end as well. Consequently, the transformed and quantized residues are processed by Inverse Quantization (IQ) 124 and Inverse Transformation (IT) 126 to recover the residues. The residues are then added back to prediction data 136 at Reconstruction (REC) 128 to reconstruct video data. The reconstructed video data may be stored in Reference Picture Buffer 134 and used for prediction of other frames.

10 [0005] As shown in Fig. 1A, incoming video data undergoes a series of processing in the encoding system. The reconstructed video data from REC 128 may be subject to various impairments due to a series of processing. Accordingly, in-loop filter 130 is often applied to the reconstructed video data before the reconstructed video data are stored in the Reference Picture Buffer 134 in order to improve video quality. For example, deblocking filter (DF), Sample Adaptive Offset (SAO) and Adaptive Loop Filter (ALF) may be used. The loop filter information may need to be incorporated in the bitstream so that a decoder can properly recover the required information. Therefore, loop filter information is also provided to Entropy Encoder 122 for incorporation into the bitstream. In Fig. 1A, Loop filter 130 is applied to the reconstructed video before the reconstructed samples are stored in the reference picture buffer 15 134. The system in Fig. 1A is intended to illustrate an exemplary structure of a typical video encoder. It may correspond to the High Efficiency Video Coding (HEVC) system, VP8, VP9, H.264 or VVC.

[0006] The decoder, as shown in Fig. 1B, can use similar or portion of the same functional blocks as the encoder except for Transform 118 and Quantization 120 since the decoder only needs Inverse Quantization 124 and Inverse Transform 126. Instead of Entropy Encoder 122, the decoder uses an Entropy Decoder 140 to decode the video bitstream into quantized transform coefficients and needed coding information (e.g. ILPF information, Intra prediction information and Inter prediction information). The Intra prediction 150 at the decoder side does not need to perform the mode search. Instead, the decoder only needs to generate Intra prediction according to Intra prediction information received from the Entropy Decoder 140. 30 Furthermore, for Inter prediction, the decoder only needs to perform motion compensation (MC 152) according to Inter prediction information received from the Entropy Decoder 140 without the need for motion estimation.

#### [0007] Partitioning of the CTUs Using a Tree Structure

35 [0008] In HEVC, a CTU is split into CUs by using a quaternary-tree (QT) structure

denoted as coding tree to adapt to various local characteristics. The decision whether to code a picture area using inter-picture (temporal) or intra-picture (spatial) prediction is made at the leaf CU level. Each leaf CU can be further split into one, two or four PUs according to the PU splitting type. Inside one PU, the same prediction process is applied and the relevant information is transmitted to the decoder on a PU basis. After obtaining the residual block by applying the prediction process based on the PU splitting type, a leaf CU can be partitioned into transform units (TUs) according to another quaternary-tree structure similar to the coding tree for the CU. One of key feature of the HEVC structure is that it has the multiple partition conceptions including CU, PU, and TU.

10 [0009] In VVC, a quadtree with nested multi-type tree using binary and ternary splits segmentation structure replaces the concepts of multiple partition unit types, i.e. it removes the separation of the CU, PU and TU concepts except as needed for CUs that have a size too large for the maximum transform length, and supports more flexibility for CU partition shapes. In the coding tree structure, a CU can have either a square or rectangular shape. A coding tree unit (CTU) is first partitioned by a quaternary tree (a.k.a. quadtree) structure. Then the quaternary tree leaf nodes can be further partitioned by a multi-type tree structure. As shown in Fig. 2, there are four splitting types in multi-type tree structure, vertical binary splitting (SPLIT\_BT\_VER 210), horizontal binary splitting (SPLIT\_BT\_HOR 220), vertical ternary splitting (SPLIT\_TT\_VER 230), and horizontal ternary splitting (SPLIT\_TT\_HOR 240). The multi-type tree leaf nodes are called coding units (CUs), and unless the CU is too large for the maximum transform length, this segmentation is used for prediction and transform processing without any further partitioning. This means that, in most cases, the CU, PU and TU have the same block size in the quadtree with nested multi-type tree coding block structure. The exception occurs when maximum supported transform length is smaller than the width or height of the colour component of the CU.

25 [0010] Fig. 3 shows a CTU divided into multiple CUs with a quadtree and nested multi-type tree coding block structure, where the bold block edges represent quadtree partitioning and the remaining edges represent multi-type tree partitioning. The quadtree with nested multi-type tree partition provides a content-adaptive coding tree structure comprised of CUs. The size of the CU may be as large as the CTU or as small as  $4 \times 4$  in units of luma samples. For the case of the 4:2:0 chroma format, the maximum chroma CB size is  $64 \times 64$  and the minimum size chroma CB consist of 16 chroma samples.

30 [0011] In VVC, the maximum supported luma transform size is  $64 \times 64$  and the maximum supported chroma transform size is  $32 \times 32$ . When the width or height of the CB is larger the maximum transform width or height, the CB is automatically split in the horizontal and/or

vertical direction to meet the transform size restriction in that direction.

[0012] In VVC, the coding tree scheme supports the ability for the luma and chroma to have separate block tree structures. For P and B slices, the luma and chroma CTBs in one CTU have to share the same coding tree structure. However, for I slices, the luma and chroma can have separate block tree structures. When the separate block tree mode is applied, luma CTB is partitioned into CUs by one coding tree structure, and the chroma CTBs are partitioned into chroma CUs by another coding tree structure. This means that a CU in an I slice may consist of a coding block of the luma component or coding blocks of two chroma components, and a CU in a P or B slice always consists of coding blocks of all three colour components unless the video is monochrome.

[0013] For each inter-predicted CU, motion parameters consisting of motion vectors, reference picture indices and reference picture list usage index, and additional information needed for the new coding feature of VVC to be used for inter-predicted sample generation. The motion parameter can be signalled in an explicit or implicit manner. When a CU is coded with skip mode, the CU is associated with one PU and has no significant residual coefficients, no coded motion vector delta or reference picture index. A merge mode is specified whereby the motion parameters for the current CU are obtained from neighbouring CUs, including spatial and temporal candidates, and additional candidates introduced in VVC. The merge mode can be applied to any inter-predicted CU, not only for skip mode. The alternative to merge mode is the explicit transmission of motion parameters, where motion vector, corresponding reference picture index for each reference picture list, reference picture list usage flag and other needed information are signalled explicitly per each CU.

[0014] ITU-T VCEG (Q6/16) and ISO/IEC MPEG (JTC 1/SC 29/WG 5) are studying the potential need for standardization of future video coding technology with a compression capability significantly exceeding that of the current VVC standard. The Enhanced Compression Model (ECM) reference software is provided to demonstrate a reference implementation of encoding techniques and the decoding process for JVET enhanced compression beyond VVC capability exploration work. The reference software can be accessed via:

<https://vcgit.hhi.fraunhofer.de/ecm/ECM.git>. E

[http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/12\\_Geneva/wg11/JCTVC-L1002-v3.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1002-v3.zip)

<https://www.itu.int/rec/T-REC-H.266-202204-I>

[https://jvet-experts.org/doc\\_end\\_user/documents/19\\_Teleconference/wg11/JVET-S2002-v2.zip](https://jvet-experts.org/doc_end_user/documents/19_Teleconference/wg11/JVET-S2002-v2.zip)

[0015] CM basically is the successor to VVC and thus it shares many common parts as VVC.

[0016] The following text provides the details on some Context-Adaptive Binary Arithmetic Coding (CABAC) related methods specified in HEVC, VVC and ECM.

[0017] **Fundamental Concept of CABAC**

[0018] CABAC is a lossless entropy coding technique used in video compression standards such as H.264/AVC, H.265/HEVC, H.266/VVC and ECM. It is used to compress the syntax elements of video data, which represent various parameters used to encode the video, such as motion vectors, quantization parameters, and prediction modes.

[0019] As illustrated in Figs. 4A-B, CABAC works by assigning binary codewords to each syntax element, with more probable syntax elements assigned shorter codewords. The probability of a CABAC context model represents the likelihood of a particular symbol (0 or 1) occurring in a particular context. The probability is a value between 0 and 1, and it is usually represented using a fixed-point arithmetic notation. The probability of a particular syntax element is estimated based on the context of the element, which includes the neighbouring syntax elements and the previously coded syntax elements. The context is used to determine the likelihood of a particular value for the syntax element.

[0020] The probability estimates are updated dynamically as each syntax element is coded, allowing the coding to adapt to the statistical properties of the video data. This adaptive context modelling is a key feature of CABAC, as it allows for more accurate probability estimates and improved compression efficiency.

[0021] The binary codewords for each syntax element are generated using binary arithmetic coding, which is a technique for encoding data using a series of binary digits. Binary arithmetic coding is a more efficient method of encoding data than fixed-length coding, as it can generate codewords of varying length depending on the probability of the syntax element.

[0022] The initial probabilities for different context models in CABAC can be different, based on the characteristics of the syntax elements being encoded. The initial probability estimates for each context model can be initialized to specific values. In general, the most effective initial probability estimates for each context model are determined through an iterative training process that involves encoding sample video data and measuring the compression efficiency and video quality achieved. In the existing video coding standards including H.264/AVC, H.265/HEVC, H.266/VVC and ECM, the initial probabilities for all the CABAC context models are pre-trained and specified in the specifications of the corresponding video coding standards. A CABAC encoder 410 is shown in Fig. 4A, where the input syntax element is converted to a binary string by using Binarization unit 412. At the same time, the syntax element is also provided to Context Modelling unit 414 for adaptively updating the model. Side information relate to the context model is also provided to Context Modelling unit 414. The

binary codewords for each syntax element are generated using Binary Arithmetic Coding unit 416. Fig. 4B illustrates an example of CABAC decoder 420, which comprises Context Modelling unit 422, DeBinarization unit 424 and Context Modelling unit 426.

[0023] In summary, CABAC is an important component of video compression standards, as it provides efficient compression of the video data while maintaining lossless quality. Its adaptive context modelling and binary arithmetic coding techniques allow for more accurate probability estimates and improved compression efficiency, leading to higher-quality video with reduced bit rates.

[0024] **CABAC in H.265/HEVC**

[0025] To achieve multiplication-free binary arithmetic coding scheme, H.265/HEVC relies on the assumption that the estimated probabilities of each context model can be represented by a sufficiently limited set of representative values. In H.265/HEVC, each probability model related to a given context index is determined by a pair of two values, a 6-bit probability state index and the (binary) value of the most probable symbol (MPS), hence the models themselves can be efficiently represented by 7-bit unsigned integer values. As a result of this design, each context model in CABAC can be completely determined by two parameters: its current estimate of the low-probability symbol (LPS) probability, which in turn is characterized by an index between 0 and 63, and its value of being either 0 or 1. Thus, probability estimation in CABAC is performed by using a total number of 128 different probability states, each of them efficiently represented by a 7-bit integer value.

[0026] For a given probability state, the update depends on the state index and the value of the encoded symbol identified either as a LPS or a most-probability symbol (MPS). As a result of the updating process, a new probability state is derived, which consists of a potentially modified LPS probability estimate and, if necessary, a modified MPS value. Fig. 5 illustrates the probability values for the LPS estimates together with their corresponding transition rules for updating the state indices. In the event of a MPS, a given state index is simply incremented by 1, unless a MPS occurs at state index 62, where the LPS probability is already at its minimum, or equivalently, the maximum MPS probability is reached. In the latter case, the state index 62 remains fixed until a LPS is seen, in which case the state index is changed by decrementing the state index by an amount illustrated by the dashed line in Fig. 5.

[0027] The CABAC engine in HEVC uses a table-based probability transition process between 64 different representative probability states. In HEVC, the range `ivlCurrRange` representing the state of the coding engine is quantized to a set of 4 values prior to the calculation of the new interval range. The HEVC state transition can be implemented using a table containing all 64x4 8-bit pre-computed values to approximate the values of

$ivlCurrRange * pLPS( pStateIdx )$ , where  $pLPS$  is the probability of the least probable symbol (LPS) and  $pStateIdx$  is the index of the current state. Also, a decode decision can be implemented using the pre-computed LUT. First  $ivlLpsRange$  is obtained using the LUT as follows. Then,  $ivlLpsRange$  is used to update  $ivlCurrRange$  and calculate the output  $binVal$ .

5  $ivlLpsRange = rangeTabLps[ pStateIdx ][ qRangeIdx ]$ .

[0028] **CABAC in VVC**

[0029] CABAC is also used in Versatile Video Coding (VVC), which is the successor to HEVC. However, the CABAC used in VVC has some differences compared to the one used in HEVC.

10 [0030] In VVC, CABAC is designed to be more efficient and to support a wider range of bit depths and chroma sampling formats. One of the main improvements in CABAC in VVC is the use of more flexible and efficient context modelling techniques, which allow for more accurate probability estimates and improved compression efficiency.

[0031] VVC also introduces a new adaptive scanning order for CABAC, which is  
15 designed to improve the efficiency of the binary arithmetic coding process. This new scanning order is optimized for the distribution of syntax elements in VVC, and it allows for faster and more accurate probability estimation.

[0032] Another key difference in CABAC in VVC is the use of a new bypass mode, which  
20 allows for direct mapping of syntax elements to binary codewords without the need for probability estimation. This can improve coding efficiency for syntax elements that have a high probability of occurrence.

[0033] Overall, CABAC in VVC is designed to be more efficient and flexible than the one used in HEVC, which helps to improve compression efficiency and reduce bit rates for high-quality video content.

25 [0034] In VVC, the probability is linearly expressed by the probability index  $pStateIdx$ . Therefore, all the calculation can be done with equations without LUT operation. To improve the accuracy of probability estimation, a multi-hypothesis probability update model is applied. The  $pStateIdx$  used in the interval subdivision in the binary arithmetic coder is a combination of two probabilities  $pStateIdx0$  and  $pStateIdx1$ . The two probabilities are associated with each  
30 context model and are updated independently with different adaptation rates. The adaptation rates of  $pStateIdx0$  and  $pStateIdx1$  for each context model are pre-trained based on the statistics of the associated bins. The probability estimate  $pStateIdx$  is the average of the estimates from the two hypotheses.

[0035] Fig. 6 shows the flowchart for decoding a single binary decision in VVC. As  
35 shown Fig. 6, the flowchart starts with function call, `DecodeDecision(ctxTable, ctxIdx)` 610. In

step 620, a set of operations related interval range is performed. In step 630, the offset to the initial probability of the context model (i.e.,  $ivlOffset$ ) is compared with the interval range (i.e.,  $ivlCurrRange$ ). If  $ivlOffset$  is greater than or equal to  $ivlCurrRange$ , step 640 is performed. Otherwise step 650 is performed. In step 640, the decoded bin value ( $binVal$ ) is set to  $!valMps$  and the  $ivlCurrRange$  is adjusted. In step 650, the decoded bin value ( $binVal$ ) is set to  $valMps$ . In step 660, a short and a long window size (i.e.,  $shift0$  and  $shift1$ ) and the index of the current state (i.e.,  $pStateIdx0$  and  $pStateIdx1$ ) are updated. In step 670, the interval range is remormalized. As done in HEVC, CABAC for VVC also has a QP dependent initialization process invoked at the beginning of each slice. Given the initial value of luma QP for the slice, the initial probability state of a context model, denoted as  $preCtxState$ , is derived as follows:

$$m = slopeIdx \times 5 - 45,$$

$$n = (offsetIdx \ll 3) + 7, \text{ and}$$

$$preCtxState = Clip3(1, 127, ((m \times (QP - 32)) \gg 4) + n).$$

where  $slopeIdx$  and  $offsetIdx$  are restricted to 3 bits, and total initialization values are represented by 6-bit precision. The probability state  $preCtxState$  represents the probability in the linear domain directly. Hence,  $preCtxState$  only needs proper shifting operations before input to arithmetic coding engine, and the logarithmic to linear domain mapping as well as the 256-byte table is saved.

$$pStateIdx0 = preCtxState \ll 3$$

$$pStateIdx1 = preCtxState \ll 7$$

#### [0036] CABAC in ECM

#### [0037] Extended precision

[0038] The intermediate precision used in the arithmetic coding engine is increased, including three elements. First, the precisions for two probability states are both increased to 15 bits, in comparison to 10 bits and 14 bits in VVC. Second, the LPS range update process is modified as below:

$$\text{if } q \geq 16384$$

$$q = 2^{15} - 1 - q$$

$$R_{LPS} = ((range * (q \gg 6)) \gg 9) + 1,$$

where  $range$  is a 9-bit variable representing the width of the current interval,  $q$  is a 15-bit variable representing the probability state of the current context model, and  $R_{LPS}$  is the updated range for LPS. This operation can also be realized by looking up a  $512 \times 256$ -entry in 9-bit look-up table. Third, at the encoder side, the 256-entry look-up table used for bits estimation in VTM is extended to 512 entries.

#### [0039] Slice-type-based window size

[0040] Since statistics are different with different slice types, it is beneficial to have a context's probability state updated at a rate that is optimal under the given slice type. Therefore, for each context model, three window sizes are pre-defined for I-, B-, and P-slices, respectively, like the initialization parameters.

5 [0041] The context initialization parameters and window sizes are retrained.

[0042] **Multi-hypothesis probability estimation with adaptive weight**

[0043] The multi-hypothesis-based probability is estimated based on adaptive weights (MHP-AW). Specifically, two separate probability estimates  $p_0$  and  $p_1$  are maintained for each context and updated according to their own adaptation rates. However, instead of using simple  
10 average, multiple weights are introduced to derive the resulting probability  $p$  used for the binary arithmetic coding, as illustrated as follows:

$$p = (\omega_0 \cdot p_0 + \omega_1 \cdot p_1) \gg s,$$

where  $\omega_0$  and  $\omega_1$  are the weights selected from a pre-defined set {10, 12, 16, 20, 22};  $s$  is the  
15 bitwise right-shift value, which is equal to 5 when  $(\omega_0 + \omega_1) \leq 32$  and 6 otherwise. Three different sets of weights are pre-determined for each context model at I-, B- and P-slice types. The weights of I-slice type are only allowed for intra slices while the weights of B- and P-slice types are allowed to be switched for inter slices at slice level.

[0044] **CABAC initialization from previous inter slice and windows adjustment**

[0045] Context initialization stored at previously coded picture after coding the last CTU  
20 can be used to initialize an inter slice having the same slice type, QP, and temporal ID. The buffer size for storing previous initializations is set equal to 5 for each slice type, when the buffer is full, the entry with the smallest QP and temporal ID is removed first before storing the initialization.

[0046] The CABAC employs two probability states that are updated with a short and a  
25 long window size, respectively. The window sizes, predefined for each context model, are not optimal for varying statistics in different regions, hence window sizes are adjusted according to the previously coded bin of each context.

[0047] The short and long window sizes used in CABAC update are adjusted by two delta  
30 parameters stored in a look-up table per context and retrieved by a previous coded bin used as an index. The previous coded bin is used as an index to get the adjustment parameters from a look-up table: delta0 for the short window and delta1 for the long window. Denote the original short and long window sizes stored in the existed initialization tables and defined for the context model as shift0 and shift1, respectively. The actual window sizes used to code the current bin after adjustment are (shift0+delta0) and (shift1+delta1), where shift0 and shift1 are  
35 existing predefined windows sizes stored in the context initialization tables.

[0048] In the present invention, methods and apparatus to modify the initial probability range for context modelling of CABAC encoder or decoder. In particular, the beginning probability or both the beginning probability and ending probability are modified so that to improve the coding efficiency for CABAC.

5

### BRIEF SUMMARY OF THE INVENTION

[0049] A method and apparatus for modifying initial probability range for context modelling associated with CABAC are disclosed. According to this method, input data related to a current block are received, wherein the input data correspond to binarized bits  
10 corresponding to one or more syntax elements of the current block to be encoded using a CABAC encoder at a video encoder side, or the input data correspond to binary codewords for said one or more syntax elements of the current block generated by the CABAC encoder and to be decoded using a CABAC decoder at a video decoder side. A modified probability range is generated by clipping a probability range for context modelling of the CABAC encoder or the  
15 CABAC decoder, wherein the modified probability range excludes a probability value of 0. The current block is encoded using the CABAC encoder according to the modified probability range or the current block is decoded using the CABAC decoder according to the modified probability range.

[0050] In one embodiment, the modified probability range is generated by clipping a  
20 beginning probability or both the beginning probability and an ending probability associated with an initial probability range for the context modelling of the CABAC encoder or the CABAC decoder. In one embodiment, a minimum probability is used to clip the beginning probability, or both the minimum probability and a maximum probability are used to clip both the beginning probability and the ending probability. In one embodiment, the minimum  
25 probability is represented by a first index, or both the minimum probability and the maximum probability are represented by the first index and a second index respectively. In one embodiment, when 15 bits are used to represent indexes for corresponding probabilities in the initial probability range, the first index corresponds to 16, or the first index and the second index correspond to 16 and (32767-16) respectively.

[0051] In one embodiment, the first index or both the first index and the second index are  
30 both predefined. In another embodiment, the first index or the first index and the second index are signalled in or parsed from one or more bitstream levels. In one embodiment, said one or more bitstream levels comprise video level, sequence level, picture level, slice level, block level, or a combination thereof.

[0052] In one embodiment, the minimum probability or the maximum probability is  
35

selected from a set of predefined values. In one embodiment, the minimum probability or the maximum probability is selected from the set of predefined values by using an index.

[0053] In one embodiment, the minimum probability is determined by subtracting an offset value from or adding the offset to the beginning probability associated with the initial probability range. In one embodiment, the offset value corresponds to a non-negative number.

[0054] In one embodiment, part or all of coded or parsed bins for each context model or part of context models are stored for generating a further modified probability range according to a modified minimum probability or both the modified minimum probability and modified maximum probability derived based on stored part or all of coded or parsed bins. In one embodiment, the modified minimum probability or both the modified minimum probability and the modified maximum probability are inherited by other slices, pictures, tiles or any other data being encoded or decoded after a current slice.

[0055] In one embodiment, a control syntax element indicating whether to use the modified probability range is signalled in a bitstream or parsed from the bitstream. In one embodiment, the control syntax element is signalled or parsed in video level, sequence level, picture level, slice level, block level, or a combination thereof.

### **BRIEF DESCRIPTION OF THE DRAWINGS**

[0056] Fig. 1A illustrates an exemplary adaptive Inter/Intra video coding system incorporating loop processing.

[0057] Fig. 1B illustrates a corresponding decoder for the encoder in Fig. 1A.

[0058] Fig. 2 illustrates an example of multi-type tree splitting modes.

[0059] Fig. 3 illustrates an example of quadtree with nested multi-type tree coding block structure.

[0060] Fig. 4A illustrates an example of CABAC encoder.

[0061] Fig. 4B illustrates an example of CABAC decoder.

[0062] Fig. 5 illustrates an example of CABAC probability transition rules.

[0063] Fig. 6 illustrates an exemplary flowchart of CABAC decoder for decoding a bin.

[0064] Fig. 7 illustrates a flowchart of an exemplary video coding system that uses a modified probability range for context modelling used by a CABAC encoder or decoder according to an embodiment of the present invention.

### **DETAILED DESCRIPTION OF THE INVENTION**

[0065] It will be readily understood that the components of the present invention, as generally described and illustrated in the figures herein, may be arranged and designed in a

wide variety of different configurations. Thus, the following more detailed description of the embodiments of the systems and methods of the present invention, as represented in the figures, is not intended to limit the scope of the invention, as claimed, but is merely representative of selected embodiments of the invention. References throughout this specification to “one  
5 embodiment,” “an embodiment,” or similar language mean that a particular feature, structure, or characteristic described in connection with the embodiment may be included in at least one embodiment of the present invention. Thus, appearances of the phrases “in one embodiment” or “in an embodiment” in various places throughout this specification are not necessarily all referring to the same embodiment.

10 [0066] Furthermore, the described features, structures, or characteristics may be combined in any suitable manner in one or more embodiments. One skilled in the relevant art will recognize, however, that the invention can be practiced without one or more of the specific details, or with other methods, components, etc. In other instances, well-known structures, or operations are not shown or described in detail to avoid obscuring aspects of the invention. The  
15 illustrated embodiments of the invention will be best understood by reference to the drawings, wherein like parts are designated by like numerals throughout. The following description is intended only by way of example, and simply illustrates certain selected embodiments of apparatus and methods that are consistent with the invention as claimed herein.

[0067] In ECM, the probability of a CABAC context model is represented by 15 bits (the  
20 largest integer number represented by 15 bits is 32767) and thus the probability ranges from 0 to 32767/32768. The granularity of the probability representation is 1/32768 and there are  $2^{15}$  probability states to represent the probability of the CABAC context model. The probability of each CABAC context model ranges from 0 to 32767/32768 which equally ranges from 0 to 99.9969% with index granularity 1/32768 (which is 0.003%). The issue here is that the  
25 probability context model may not need extremely low and/or high probability (e.g. <1% or >99%).

[0068] In this invention, it is proposed to clip the range of the probability of a CABAC model within a predefined range.

[0069] In the first embodiment of this invention, a minimum probability and a maximum  
30 probability are used to clip the range of the probability of all CABAC context models. The minimum and maximum probability can be expressed as an index (or termed as state) of the probability (or a probability value). In one example that utilizes the ECM CABAC probability representation, the minimum probability is index 16 while the maximum probability is index 32767-16 (which is 32751). The index (or termed as state) of the minimum and maximum  
35 probability can then be predefined at the encoder and decoder. Alternatively, the index (or

termed as state) of the minimum and maximum probability can then be signalled in the bitstream at different levels such as video level (e.g. video parameters set, VPS), sequence level (e.g. sequence parameter set, SPS), picture level (e.g. picture parameters set, PPS; picture header), slice level (e.g. slice header), and/or block level (e.g. CTU, CU, CTB, CB).

5 [0070] In the second embodiment of this invention, a minimum probability and a maximum probability are used to clip the range of the probability of one specific CABAC context model. The minimum and maximum probability could be expressed as index (or termed as state) of the probability (or a probability value). In one example which utilizes the ECM  
10 CABAC probability representation, the minimum probability is index 16 while the maximum probability is index 32767-16 (which is 32751). The index (or termed as state) of the minimum and maximum probability could then be predefined at the encoder and decoder. Alternatively, the index (or termed as state) of the minimum and maximum probability could then be signaled into the bitstream at different levels such as video level (e.g. video parameters set, VPS),  
15 sequence level (e.g. sequence parameter set, SPS), picture level (e.g. picture parameters set, PPS; picture header), slice level (e.g. slice header), and/or block level (e.g. CTU, CU, CTB, CB).

[0071] In the third embodiment, a minimum probability and a maximum probability are used to clip the range of the probability of one set of at least one CABAC context models. The minimum and maximum probability could be expressed as index (or termed as state) of the  
20 probability (or a probability value). In one example which utilizes the ECM CABAC probability representation, the minimum probability is index 16 while the maximum probability is index 32767-16 (which is 32751). The index (or termed as state) of the minimum and maximum probability could then be predefined at the encoder and decoder. Alternatively, the index (or termed as state) of the minimum and maximum probability could then be signaled  
25 into the bitstream at different levels such as video level (e.g. video parameters set, VPS), sequence level (e.g. sequence parameter set, SPS), picture level (e.g. picture parameters set, PPS; picture header), slice level (e.g. slice header), and/or block level (e.g. CTU, CU, CTB, CB). The set of CABAC context models can be CU partition related context set, CU/PU/TU mode related context set, coefficient coding related context set, loop filtering related context  
30 set.

[0072] In the fourth embodiment, a minimum probability and a maximum probability are used to clip the range of the probability of the CABAC context models. The minimum and maximum probability could be expressed as index (or termed as state) of the probability or a probability value. The minimum and/or maximum probability could be selected as one  
35 probability value from a set of predefined probability indices or probability values. For

example, a set for minimum probabilities is defined as  $\{16/32768, 32/32768, 64/32768, 128/32768\}$ ; the minimum probability is then specified by the index to the minimum probability set. Similarly, a set for maximum probabilities is defined as  $\{32752/32768, 32736/32768, 32720/32768, 32704/32768\}$ ; the maximum probability is then specified by the index to the maximum probability set.

[0073] In the fifth embodiment, the minimum probability is determined by subtracting an offset to the initial probability of the context model. The offset value could be any non-negative number. For example, if the initial probability of a context model is “8192/32768” and the offset is “4096/32768”, the minimum probability is then determined as “(8192-4096)/32768” which is “4096/32768”.

[0074] In the sixth embodiment, the maximum probability is determined by adding an offset to the initial probability of the context model. The offset value could be any non-negative number. For example, if the initial probability of a context model is “8192/32768” and the offset is “4096/32768”, the minimum probability is then determined as “(8192+4096)/32768” which is “12288/32768”.

[0075] In the seventh embodiment, part of coded/parsed bins or all of coded/parsed bin for each context model or part of context models can be stored. The encoder and the decoder can calculate the best probability limitation (e.g. best maximum and best minimum probabilities) for each context model or part of context models by using the stored bins. In one example, codec can re-encode the CABAC for one certain context model with the stored bins with different probability limitation and see how many bits are required with this probability limitation. From the result, one best probability limitation can be determined. The information of the best probability limitation can be inherited by other slices/pictures/tiles or any other data that encoded/decoded after the current slice. For example, the best probability limitation can be inherited by the following coded slice with same QP or with same temporal ID. In this embodiment, in one example, each context model can have different probability limitation. In another example, all the context models can have the same probability limitation. In another example, the context models can be classified into one or more sets. Each context model set can be one probability limitation for the context models in this set.

[0076] In the eighth embodiment, the syntax element is signalled in the bitstream to indicate the enabling or disabling of the foregoing proposed methods. The syntax element can be signalled in different level such as video level (e.g. video parameters set, VPS), sequence level (e.g. sequence parameter set, SPS), picture level (e.g. picture parameters set, PPS; picture header), slice level (e.g. slice header), and/or block level (e.g. CTU, CU, CTB, CB).

[0077] Any of the foregoing proposed CABAC methods can be applied independently or

jointly. Moreover, any of the foregoing CABAC proposed methods can be implemented in encoders and/or decoders. For example, any of the proposed CABAC methods can be implemented in entropy coding (e.g. CABAC) module of an encoder and/or a decoder. Alternatively, any of the proposed methods can be implemented as a circuit coupled to entropy coding (e.g. CABAC) module of the encoder and/or the decoder. With respect to the reference video encoder in Fig. 1A and the reference video decoder in Fig. 1B, any of the proposed CABAC methods can be implemented in the Entropy Encoder module 122 in the encoder side or the Entropy Decoder module 140 in the decoder side. The required processing for the present invention may be implemented according to executable software or firmware codes stored on a media, such as hard disk or flash memory, for a CPU (Central Processing Unit) or programmable devices (e.g. DSP (Digital Signal Processor) or FPGA (Field Programmable Gate Array)).

[0078] Fig. 7 illustrates a flowchart of an exemplary video coding system that uses a modified probability range for context modelling used by a CABAC encoder or decoder according to an embodiment of the present invention. The steps shown in the flowchart may be implemented as program codes executable on one or more processors (e.g., one or more CPUs) at the encoder side. The steps shown in the flowchart may also be implemented based hardware such as one or more electronic devices or processors arranged to perform the steps in the flowchart. A method and apparatus for modifying initial probability range for context modelling associated with CABAC are disclosed. According to this method, input data related to a current block are received in step 710, wherein the input data correspond to binarized bits corresponding to one or more syntax elements of the current block to be encoded using a CABAC encoder at a video encoder side, or the input data correspond to binary codewords for said one or more syntax elements of the current block generated by the CABAC encoder and to be decoded using a CABAC decoder at a video decoder side. A modified probability range is generated in step 720 by clipping a probability range for context modelling of the CABAC encoder or the CABAC decoder, wherein the modified probability range excludes a probability value of 0. The current block is encoded using the CABAC encoder according to the modified probability range or the current block is decoded using the CABAC decoder according to the modified probability range in step 730.

[0079] The flowcharts shown are intended to illustrate an example of video coding according to the present invention. A person skilled in the art may modify each step, rearranges the steps, split a step, or combine steps to practice the present invention without departing from the spirit of the present invention. In the disclosure, specific syntax and semantics have been used to illustrate examples to implement embodiments of the present

invention. A skilled person may practice the present invention by substituting the syntax and semantics with equivalent syntax and semantics without departing from the spirit of the present invention.

[0080] The above description is presented to enable a person of ordinary skill in the art to practice the present invention as provided in the context of a particular application and its requirement. Various modifications to the described embodiments will be apparent to those with skill in the art, and the general principles defined herein may be applied to other embodiments. Therefore, the present invention is not intended to be limited to the particular embodiments shown and described, but is to be accorded the widest scope consistent with the principles and novel features herein disclosed. In the above detailed description, various specific details are illustrated in order to provide a thorough understanding of the present invention. Nevertheless, it will be understood by those skilled in the art that the present invention may be practiced.

[0081] Embodiment of the present invention as described above may be implemented in various hardware, software codes, or a combination of both. For example, an embodiment of the present invention can be one or more circuit circuits integrated into a video compression chip or program code integrated into video compression software to perform the processing described herein. An embodiment of the present invention may also be program code to be executed on a Digital Signal Processor (DSP) to perform the processing described herein. The invention may also involve a number of functions to be performed by a computer processor, a digital signal processor, a microprocessor, or field programmable gate array (FPGA). These processors can be configured to perform particular tasks according to the invention, by executing machine-readable software code or firmware code that defines the particular methods embodied by the invention. The software code or firmware code may be developed in different programming languages and different formats or styles. The software code may also be compiled for different target platforms. However, different code formats, styles and languages of software codes and other means of configuring code to perform the tasks in accordance with the invention will not depart from the spirit and scope of the invention.

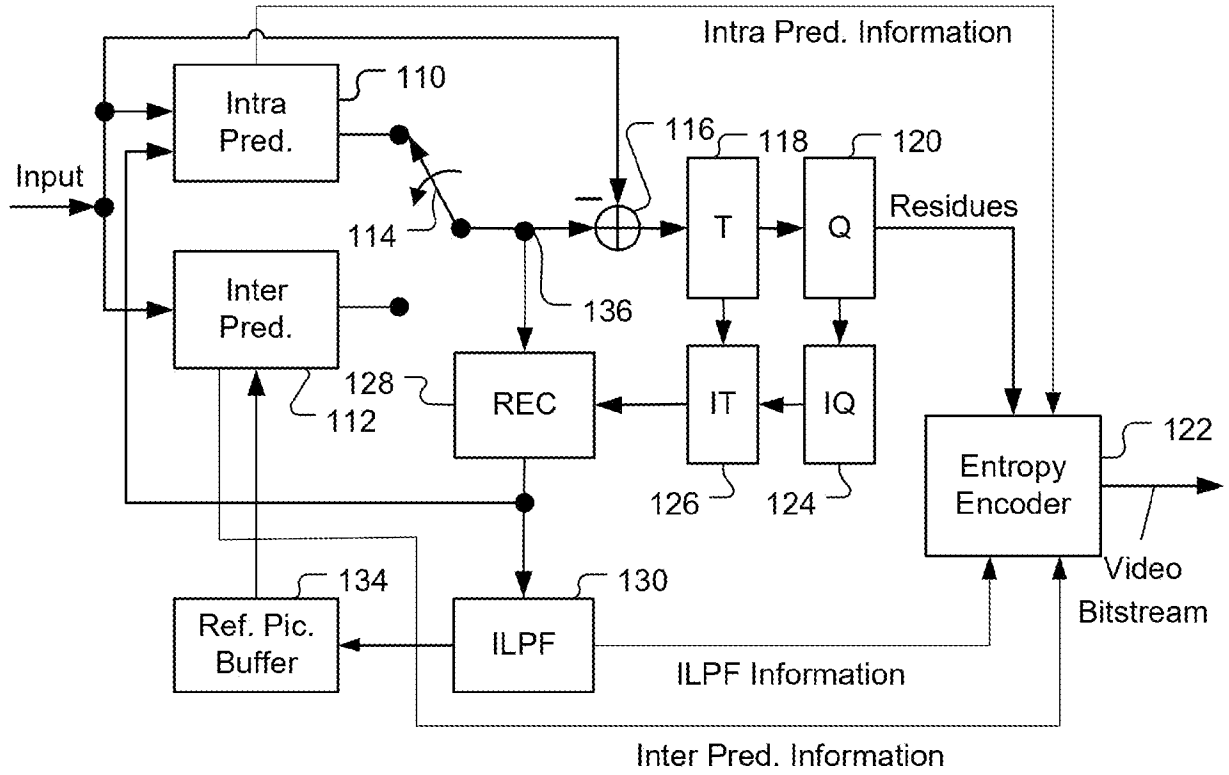
[0082] The invention may be embodied in other specific forms without departing from its spirit or essential characteristics. The described examples are to be considered in all respects only as illustrative and not restrictive. The scope of the invention is therefore, indicated by the appended claims rather than by the foregoing description. All changes which come within the meaning and range of equivalency of the claims are to be embraced within their scope.

**CLAIMS**

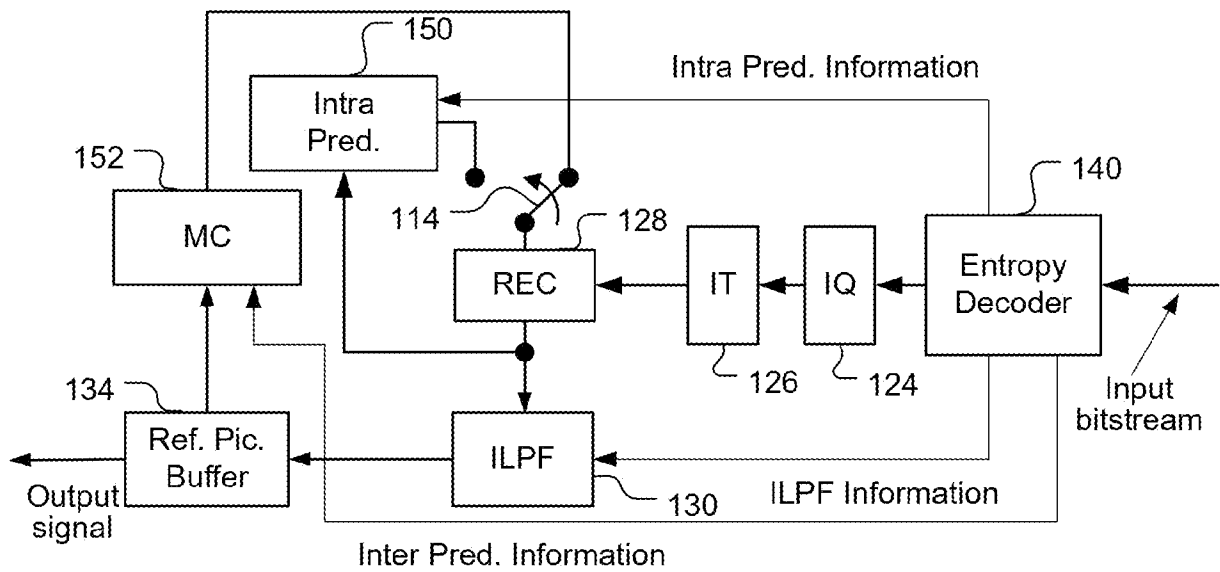
1. A method of video coding, the method comprising:
  - receiving input data related to a current block, wherein the input data correspond to binarized bits corresponding to one or more syntax elements of the current block to be
  - 5 encoded using a CABAC encoder at a video encoder side, or the input data correspond to binary codewords for said one or more syntax elements of the current block generated by the CABAC encoder and to be decoded using a CABAC decoder at a video decoder side;
  - generating a modified probability range by clipping a probability range for context modelling of the CABAC encoder or the CABAC decoder, wherein the modified
  - 10 probability range excludes a probability value of 0; and
  - encoding the current block using the CABAC encoder according to the modified probability range or decoding the current block using the CABAC decoder according to the modified probability range.
2. The method of Claim 1, wherein the modified probability range is generated by clipping a
- 15 beginning probability or both the beginning probability and an ending probability associated with an initial probability range for the context modelling of the CABAC encoder or the CABAC decoder.
3. The method of Claim 2, wherein a minimum probability is used to clip the beginning probability, or both the minimum probability and a maximum probability are used to clip both
- 20 the beginning probability and the ending probability.
4. The method of Claim 3, wherein the minimum probability is represented by a first index, or both the minimum probability and the maximum probability are represented by the first index and a second index respectively.
5. The method of Claim 4, wherein when 15 bits are used to represent indexes for
- 25 corresponding probabilities in the initial probability range, the first index corresponds to 16, or the first index and the second index correspond to 16 and (32767-16) respectively.
6. The method of Claim 4, wherein the first index or both the first index and the second index are both predefined.
7. The method of Claim 4, wherein the first index or the first index and the second index are
- 30 signalled in or parsed from one or more bitstream levels.
8. The method of Claim 7, wherein said one or more bitstream levels comprise video level, sequence level, picture level, slice level, block level, or a combination thereof.
9. The method of Claim 3, wherein the minimum probability or the maximum probability is selected from a set of predefined values.
- 35 10. The method of Claim 9, wherein the minimum probability or the maximum probability is

selected from the set of predefined values by using an index.

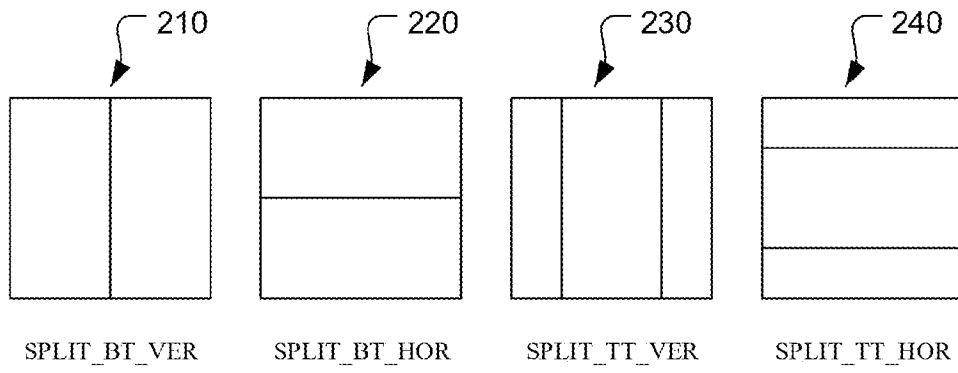
11. The method of Claim 3, wherein the minimum probability is determined by subtracting an offset value from or adding the offset to the beginning probability associated with the initial probability range.
- 5 12. The method of Claim 11, wherein the offset value corresponds to a non-negative number.
13. The method of Claim 1, wherein part or all of coded or parsed bins for each context model or part of context models are stored for generating a further modified probability range according to a modified minimum probability or both the modified minimum probability and modified maximum probability derived based on stored part or all of coded or parsed bins.
- 10 14. The method of Claim 13, wherein the further modified probability range is used by the CABAC encoder for re-encoding the current block.
15. The method of Claim 13, wherein the modified minimum probability or both the modified minimum probability and the modified maximum probability are inherited by other slices, pictures, tiles or any other data being encoded or decoded after a current slice.
- 15 16. The method of Claim 1, wherein a control syntax element indicating whether to use the modified probability range is signalled in a bitstream or parsed from the bitstream.
17. The method of Claim 16, wherein the control syntax element is signalled or parsed in video level, sequence level, picture level, slice level, block level, or a combination thereof.
18. An apparatus of video coding, the apparatus comprising one or more electronic circuits or  
20 processors arranged to:
  - receive input data related to a current block, wherein the input data correspond to binarized bits corresponding to one or more syntax elements of the current block to be encoded using a CABAC encoder at a video encoder side, or the input data correspond to binary codewords for said one or more syntax elements of the current block generated by  
25 the CABAC encoder and to be decoded using a CABAC decoder at a video decoder side;
  - generate a modified probability range by clipping a probability range for context modelling of the CABAC encoder or the CABAC decoder, wherein the modified probability range excludes a probability value of 0; and
  - 30 encode the current block using the CABAC encoder according to the modified probability range or decode the current block using the CABAC decoder according to the modified probability range.



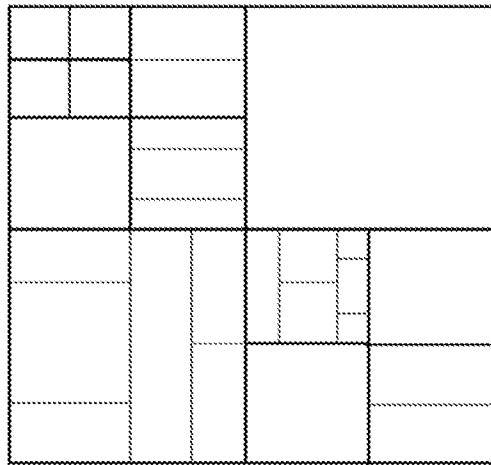
**Fig. 1A**



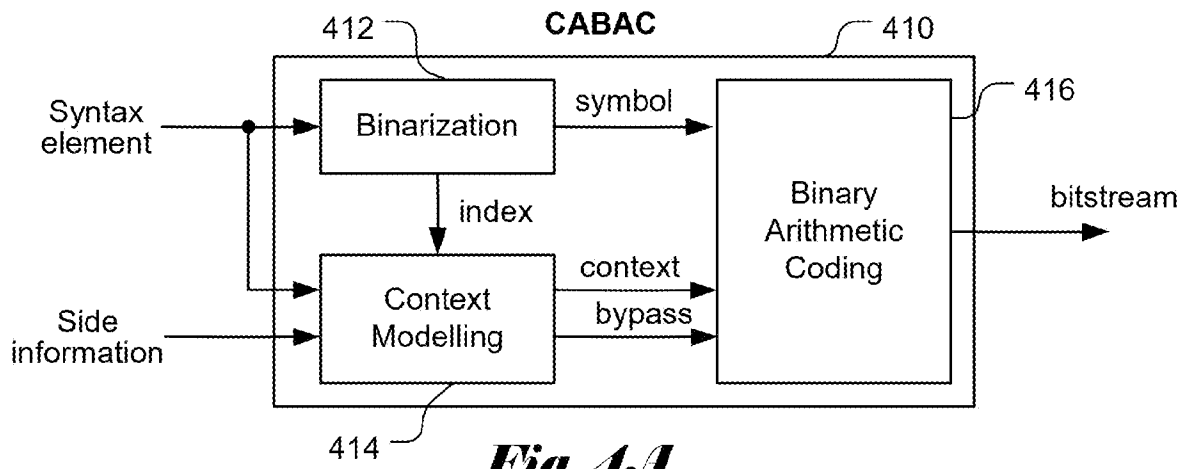
**Fig. 1B**



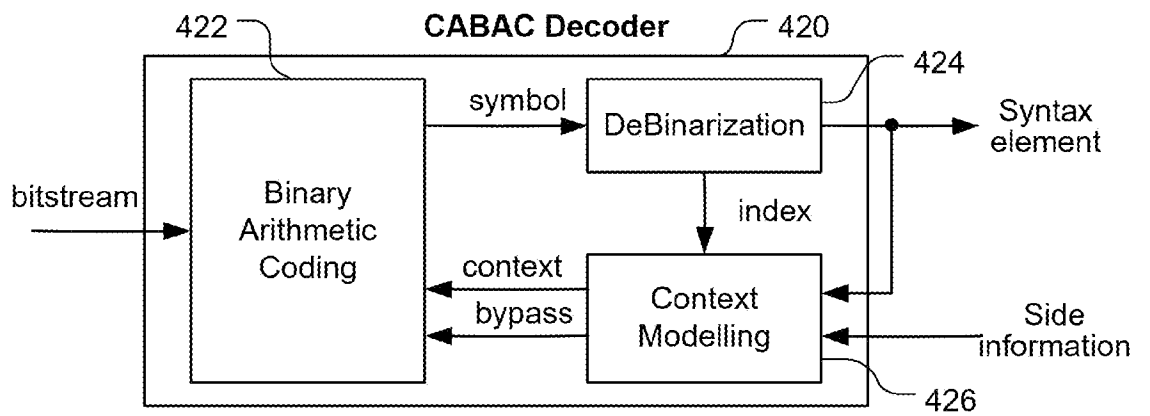
**Fig. 2**



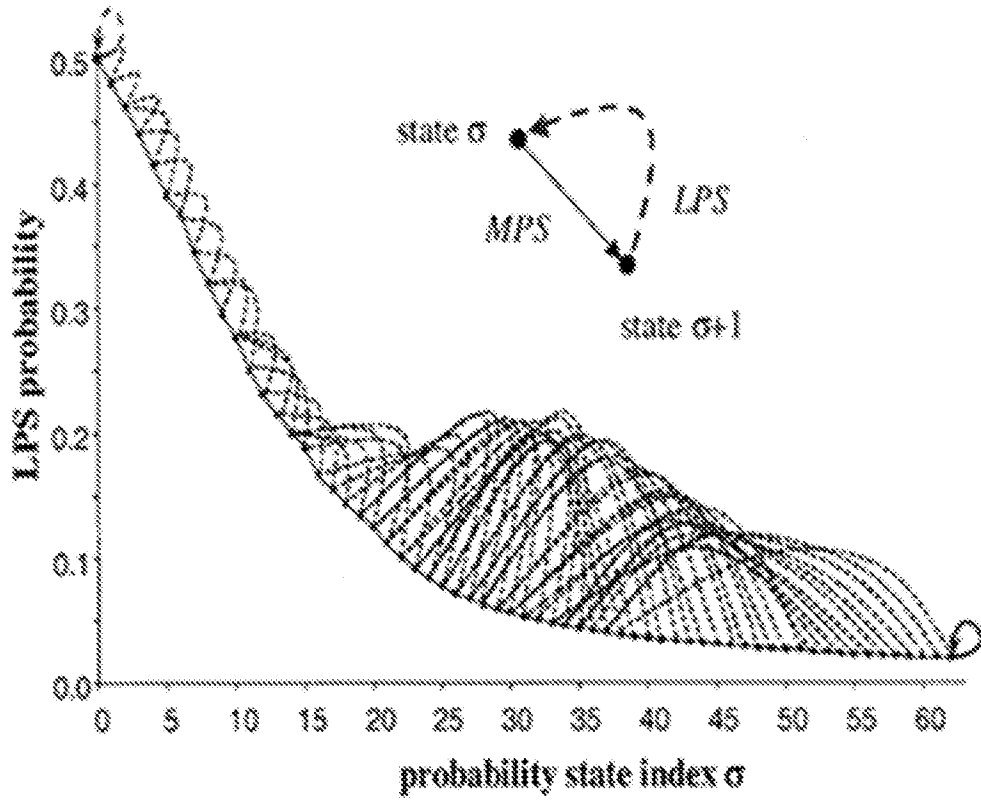
**Fig. 3**



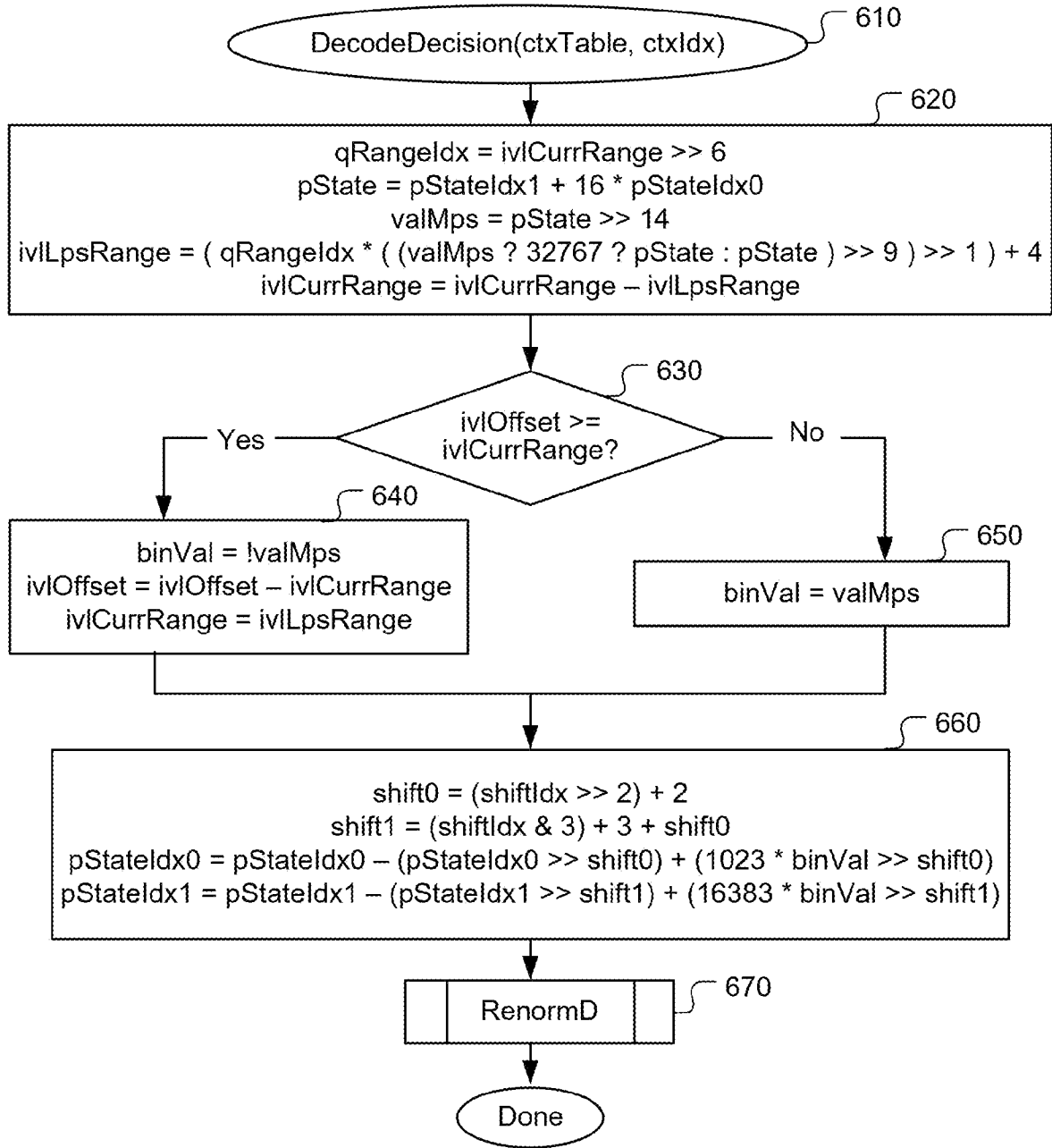
**Fig. 4A**



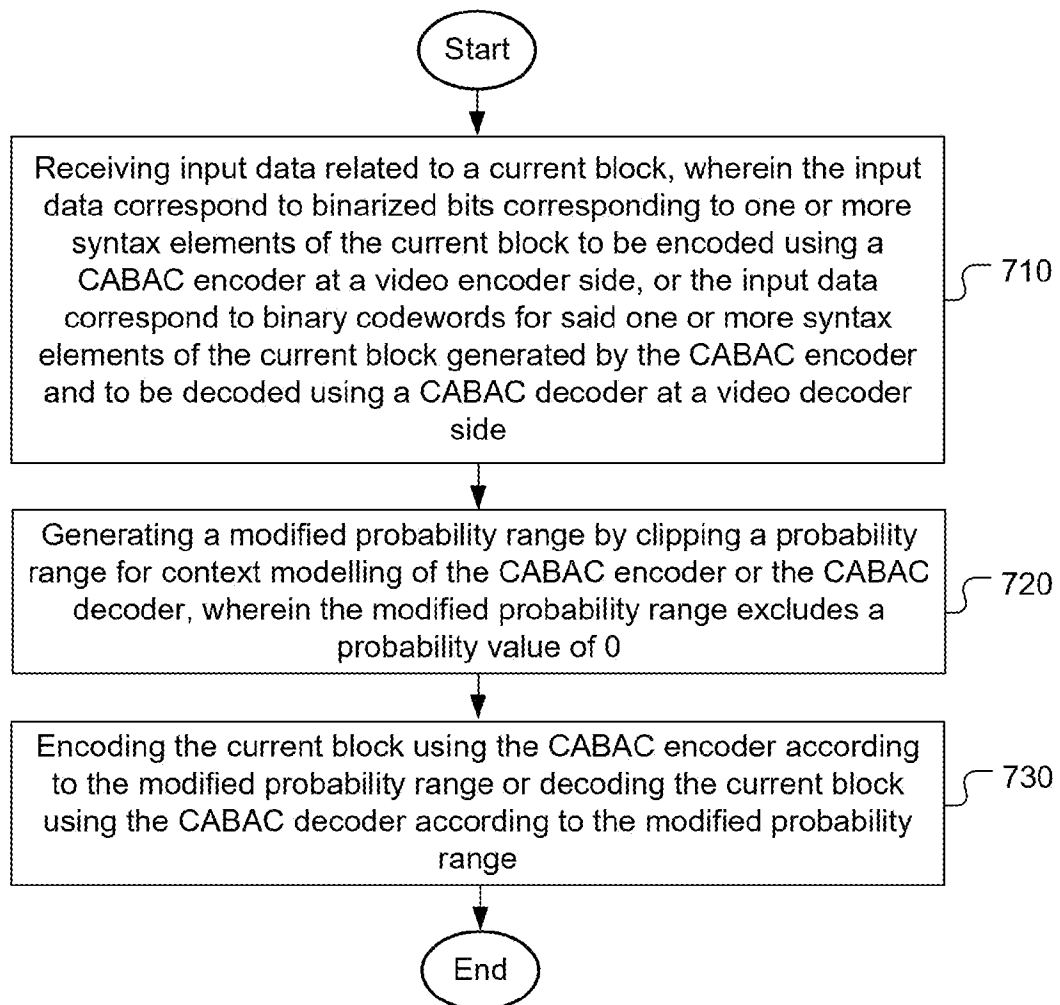
**Fig. 4B**



**Fig. 5**



**Fig. 6**

**Fig. 7**

## INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2024/097006

<b>A. CLASSIFICATION OF SUBJECT MATTER</b>		
H04N 19/21(2014.01)i		
According to International Patent Classification (IPC) or to both national classification and IPC		
<b>B. FIELDS SEARCHED</b>		
Minimum documentation searched (classification system followed by classification symbols)		
IPC: H04N		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)		
CNTXT, CNABS, WPABS, WPABSC, ENTXT, ENTXTC, DWPI, CNKI, IEEE, JNET: video, decod+, coding, block, CABAC, binary, probability range, beginning probability, ending probability, modif+, clip+, correct+, minimum probability, maximum probability, index, offset		
<b>C. DOCUMENTS CONSIDERED TO BE RELEVANT</b>		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2020186802 A1 (MEDIATEK INC.) 11 June 2020 (2020-06-11) description, paragraphs [0027]-[0032]	1-18
A	CN 103797803 A (SAMSUNG ELECTRONICS CO., LTD.) 14 May 2014 (2014-05-14) the whole document	1-18
A	CN 107667530 A (QUALCOMM INC.) 06 February 2018 (2018-02-06) the whole document	1-18
A	CN 110291793 A (MEDIATEK INC.) 27 September 2019 (2019-09-27) the whole document	1-18
A	US 2022132173 A1 (HYUNDAI MOTOR COMPANY et al.) 28 April 2022 (2022-04-28) the whole document	1-18
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "D" document cited by the applicant in the international application "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family		
Date of the actual completion of the international search		Date of mailing of the international search report
01 August 2024		19 August 2024
Name and mailing address of the ISA/CN		Authorized officer
<b>CHINA NATIONAL INTELLECTUAL PROPERTY ADMINISTRATION</b> <b>6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing 100088, China</b>		<b>WANG, CongLei</b>  Telephone No. (+86) 010-53961717

**INTERNATIONAL SEARCH REPORT**  
**Information on patent family members**

International application No.

**PCT/CN2024/097006**

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2020186802	A1	11 June 2020	US	2021266555	A1	26 August 2021
				EP	3652937	A1	20 May 2020
				TW	201909633	A	01 March 2019
				CA	3069635	A1	17 January 2019
				WO	2019011292	A1	17 January 2019
				CN	110915213	A	24 March 2020
-----							
CN	103797803	A	14 May 2014	EP	2760201	A2	30 July 2014
				JP	2014523186	A	08 September 2014
				WO	2013002585	A2	03 January 2013
				US	2014177708	A1	26 June 2014
				KR	20130002298	A	07 January 2013
-----							
CN	107667530	A	06 February 2018	TW	201701664	A	01 January 2017
				TW	201705760	A	01 February 2017
				JP	2018521555	A	02 August 2018
				WO	2016196307	A1	08 December 2016
				BR	112017025605	A2	07 August 2018
				EP	3304909	A1	11 April 2018
				KR	20180013928	A	07 February 2018
				US	2016353110	A1	01 December 2016
				AU	2016270616	A1	09 November 2017
				WO	2016196287	A1	08 December 2016
				BR	112017025612	A2	07 August 2018
				JP	2018521556	A	02 August 2018
				EP	3304910	A1	11 April 2018
				KR	20180013927	A	07 February 2018
				AU	2016270634	A1	09 November 2017
				US	2016353108	A1	01 December 2016
-----							
CN	110291793	A	27 September 2019	MX	2019008186	A	09 September 2019
				US	2018199046	A1	12 July 2018
				WO	2018127167	A1	12 July 2018
				TW	201832554	A	01 September 2018
				PL	3560201	T3	28 August 2023
				BR	112019013916	A2	04 February 2020
				EP	3560201	A1	30 October 2019
				ZA	201905142	A	21 December 2022
-----							
US	2022132173	A1	28 April 2022	WO	2021025485	A1	11 February 2021
				KR	20210018140	A	17 February 2021
				CN	114556927	A	27 May 2022
-----							