



## (51) International Patent Classification:

H04N 19/11 (2014.01) H04N 19/46 (2014.01)  
H04N 19/593 (2014.01)

## (21) International Application Number:

PCT/EP2020/078460

## (22) International Filing Date:

09 October 2020 (09.10.2020)

## (25) Filing Language:

English

## (26) Publication Language:

English

## (30) Priority Data:

19306330.2 11 October 2019 (11.10.2019) EP

(71) Applicant: **INTERDIGITAL VC HOLDINGS FRANCE, SAS** [FR/FR]; 975 avenue des Champs Blancs, 35576 Cesson Sevigne (FR).

(72) Inventors: **DUMAS, Thierry**; c/o InterDigital R&D France, SAS, 975 Avenue des Champs Blancs, 35576 Cesson Sevigne (FR). **GALPIN, Franck**; c/o InterDigital R&D France, SAS, 975 Avenue des Champs Blancs, 35576 Cesson Sevigne (FR). **BEGAINT, Jean**; c/o InterDigital Communications, Inc, 3000 El Camino Real BLd 4 # 200, Palo Alto, California 94306 (US). **RACAPE, Fabien**; c/o Inter-

Digital Communications, Inc, 3000 El Camino Real BLd 4 # 200, Palo Alto, California 94306 (US).

(74) Agent: **ROLLAND, Sophie** et al.; c/o InterDigital, 20 rue Rouget de Lisle, 92130 Issy-les-Moulineaux (FR).

(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, IT, 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, 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, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

## (54) Title: DEEP INTRA PREDICTOR GENERATING SIDE INFORMATION

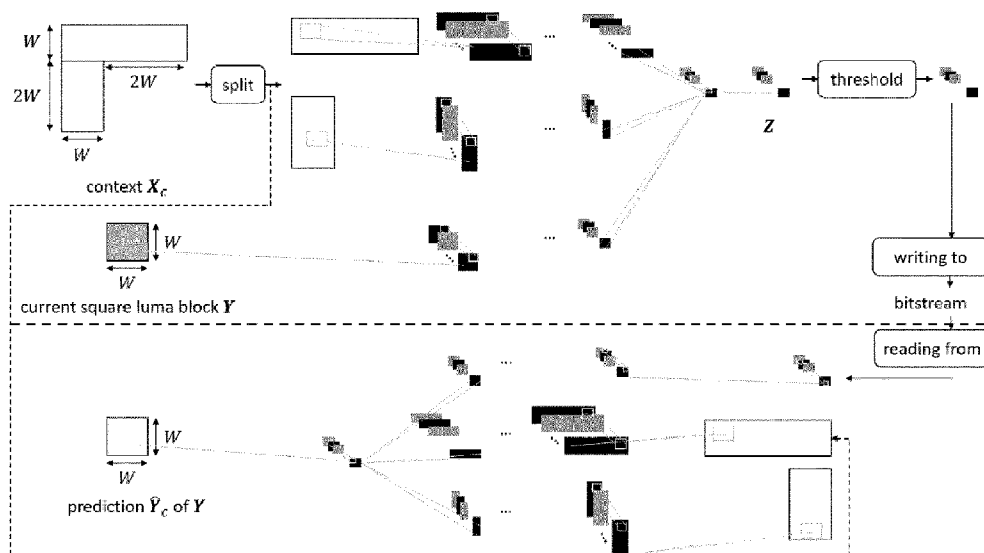


Figure 10

(57) Abstract: At least a method and an apparatus are presented for efficiently encoding or decoding video. For example, an intra prediction of an image block and side information are determined using at least one neural network from a context comprising pixels surrounding the image block. The side information allows a decoder to determine the intra prediction and is signaled for the decoding.

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))*

## Deep intra predictor generating side information

### TECHNICAL FIELD

5 At least one of the present embodiments generally relates to a method or an apparatus for video encoding or decoding, and more particularly, to a method or an apparatus that obtains a neural network intra prediction and side information using a neural network from at least one input data; and encodes or decodes the side information and an image block using the neural network intra prediction.

### BACKGROUND

10 To achieve high compression efficiency, image and video coding schemes usually employ prediction, including motion vector prediction, and transform to leverage spatial and temporal redundancy in the video content. Generally, intra or inter prediction is used to exploit the intra or inter frame correlations, then the difference between an original image block and its prediction, often denoted as prediction error or prediction residual, is transformed, quantized, and entropy coded. To reconstruct the video, the compressed data are decoded by inverse processes corresponding to the entropy coding, quantization, transform, and prediction.

20 Recent additions to video compression technology include various industry standards, versions of the reference software and/or documentations such as Joint Exploration Model (JEM) and later VTM (Versatile Video Coding (VVC) Test Model) being developed by the JVET (Joint Video Exploration Team) group. The aim is to make further improvements to the existing HEVC (High Efficiency Video Coding) standard.

25 Recent works introduce deep neural networks for improving video compression efficiency in terms of bitrate savings. For instance, a deep intra predictor infers from the context surrounding the current block to be predicted a prediction of this block. According to previous works related to the learning of a deep intra predictor for image and video compression, when the learned deep intra predictor is inserted into a video codec, such as HEVC for instance, it is always at least one additional intra prediction mode, in competition with the existing ones. Indeed, the intra prediction component of video codecs cannot rely on the deep intra predictor alone because of the "failure cases". A "failure case" refers to a situation where the learned deep intra predictor infers a prediction of the current block of relatively low quality, compared to the prediction with the best quality among the predictions provided by the regular intra prediction modes in the video codec of interest. In general, a "failure case" occurs when the information

30  
35

in the context is not enough for inferring a prediction of the current block of good quality or, roughly speaking, the context is too decorrelated from the current block.

## SUMMARY

5

The drawbacks and disadvantages of the prior art are solved and addressed by the general aspects described herein, which are directed to a deep intra predictor generating side information. According to at least one embodiment, the deep intra predictor generates side information on the encoder side, the side information is written to the bitstream, and the deep  
10 intra prediction reads the side information on the decoder side. This way, it is possible to transmit from the encoder to the decoder information to supplement the information contained in the context surrounding the current block to be predicted.

15

According to a first aspect, there is provided a method. The method comprises determining, for a block being encoded, a neural network intra prediction and side information using a neural network from at least one input data; encoding the block based on the neural network intra prediction; and encoding the side information.

20

According to another aspect, there is provided a second method. The method comprises obtaining, for a block being decoded, side information relative to a neural network intra prediction; determining, for the block being decoded, a neural network intra prediction using a neural network applied to at least one input data and the side information; and decoding the block using said determined neural network intra prediction.

25

According to another aspect, there is provided an apparatus. The apparatus comprises one or more processors, wherein the one or more processors are configured to: determine, for a block being encoded, a neural network intra prediction and side information using a neural network from at least one input data; encode the block based on the neural network intra prediction; and encode the side information.

30

According to another aspect, there is provided another apparatus. The apparatus comprises one or more processors, wherein the one or more processors are configured to: obtain, for a block being decoded, side information relative to a neural network intra prediction; determining, for the block being decoded, a neural network intra prediction using a neural network applied to at least one input data and the side information; and decoding the block using the determined neural network intra prediction.

35

According to another general aspect of at least one embodiment, the neural network is fully-connected and side information  $Z$  is a vector whose coefficients belong to  $[0,1]$ . In a variant,

the vector  $Z$  is converted into a vector of bits by applying an elementwise threshold of 0.5 to the vector  $Z$ . In another variant, the elementwise threshold value is different from 0.5.

According to another general aspect of at least one embodiment, the neural network is convolutional and wherein side information  $Z$  is a stack of feature maps whose coefficients belong to  $[0, 1]$ . In a variant, the stack of feature maps  $Z$  is converted into a stack of feature maps of bits by applying an elementwise threshold of 0.5 to  $Z$ . In another variant, the elementwise threshold value is different from 0.5.

According to another general aspect of at least one embodiment, side information  $Z$  is a vector or a stack of feature maps whose coefficients belong to  $\mathbb{R}$  and the coefficients of side information are quantized via either scalar quantization or vector quantization. Then, the absolute value of each quantized coefficient is encoded losslessly via arithmetic coding. A CABAC context can be used for the arithmetic coding of the absolute value of each quantized coefficient.

According to another general aspect of at least one embodiment, side information  $Z$  is a vector or a stack of feature maps whose coefficients belong to  $[-1, 1]$  and the coefficients of side information are mapped to  $\{0, 1\}$ .

According to another general aspect of at least one embodiment, for luma block  $Y$ , the input data of the encoding comprises a context  $X_c$  surrounding the current luma block  $Y$  and luma block  $Y$ .

According to another general aspect of at least one embodiment, for luma block  $Y$ , the input data of the decoding comprises a context  $X_c$  surrounding the current luma block  $Y$  and the decoded side information.

According to another general aspect of at least one embodiment, for a chroma block  $Y^{CbCr}$ , the input data of the encoding comprises a context  $X_c^{CbCr}$  surrounding the chroma block  $Y^{CbCr}$ , the context  $X_c^Y$  surrounding the luma block  $Y^Y$  that is collocated with the chroma block, and the current chroma block  $Y^{CbCr}$ .

According to another general aspect of at least one embodiment, for a chroma block  $Y^{CbCr}$ , the input data of the encoding comprises the context  $X_c^Y$  surrounding the luma block  $Y^Y$  that is collocated with the chroma block, and the current chroma block  $Y^{CbCr}$ .

According to another general aspect of at least one embodiment, for a chroma block  $Y^{CbCr}$ , the input data of the encoding comprises a context  $X_c^{CbCr}$  surrounding the chroma block  $Y^{CbCr}$ , the context  $X_c^Y$  surrounding the luma block  $Y^Y$  that is collocated with the chroma block, the current chroma block  $Y^{CbCr}$ , and the reconstructed luma block  $\tilde{Y}^Y$ .

According to another general aspect of at least one embodiment, for a chroma block  $Y^{CbCr}$ , the input data of the decoding comprises a context  $X_c^{CbCr}$  surrounding the chroma block  $Y^{CbCr}$ ,

the context  $X_c^Y$  surrounding the luma block  $Y^Y$  that is collocated with the chroma block and the decoded side information.

According to another general aspect of at least one embodiment, for a chroma block  $Y^{CbCr}$ , the input data of the decoding comprises the context  $X_c^Y$  surrounding the luma block  $Y^Y$  that is collocated with the chroma block and the decoded side information.

According to another general aspect of at least one embodiment, for a chroma block  $Y^{CbCr}$ , the input data of the decoding comprises a context  $X_c^{CbCr}$  surrounding the chroma block  $Y^{CbCr}$ , the context  $X_c^Y$  surrounding the luma block  $Y^Y$  that is collocated with the chroma block, the reconstructed luma block  $\tilde{Y}^Y$  and the decoded side information.

According to another general aspect of at least one embodiment, the input data further comprises the intra prediction mode  $L$  of the block located on the left side of the block being encoded/decoded and the intra prediction mode  $A$  of the block located above the block being encoded/decoded.

According to another general aspect of at least one embodiment, for intra prediction of a luma block, a syntax element DeepFlag representative of a neural network-based intra prediction mode (i.e. the neural network-based intra prediction mode is selected to predict the current block) is encoded among syntax elements of intra prediction.

According to another general aspect of at least one embodiment, for intra prediction of a luma block, the neural network-based intra prediction mode is always selected. This means that all regular syntax elements of luma intra prediction and DeepFlag do not exist.

According to another general aspect of at least one embodiment, for intra prediction of a luma block, a syntax element DeepFlag representative of a neural network-based intra prediction mode is encoded and if DeepFlag is set to one the luma block is intra predicted using a neural network-based intra prediction mode else if DeepFlag is set to zero, the luma block is intra predicted using planar mode.

According to another general aspect of at least one embodiment, for intra prediction of a chroma block, the neural network-based intra prediction mode is always selected.

According to another general aspect of at least one embodiment, for intra prediction of a chroma block, a syntax element DeepFlag representative of a neural network-based intra prediction mode is encoded among syntax elements of intra prediction.

According to another general aspect of at least one embodiment, there is provided a device comprising an apparatus according to any of the decoding embodiments; and at least one of (i) an antenna configured to receive a signal, the signal including the video block, (ii) a band limiter configured to limit the received signal to a band of frequencies that includes the video block, or (iii) a display configured to display an output representative of the video block.

According to another general aspect of at least one embodiment, there is provided a non-transitory computer readable medium containing data content generated according to any of the described encoding embodiments or variants.

According to another general aspect of at least one embodiment, there is provided a signal comprising video data generated according to any of the described encoding embodiments or variants.

According to another general aspect of at least one embodiment, a bitstream is formatted to include data content generated according to any of the described encoding embodiments or variants.

According to another general aspect of at least one embodiment, there is provided a computer program product comprising instructions which, when the program is executed by a computer, cause the computer to carry out any of the described decoding embodiments or variants.

These and other aspects, features and advantages of the general aspects will become apparent from the following detailed description of exemplary embodiments, which is to be read in connection with the accompanying drawings.

#### BRIEF DESCRIPTION OF THE DRAWINGS

In the drawings, examples of several embodiments are illustrated.

Figure 1 illustrates an example of reference sample for intra prediction in VVC.

Figure 2 shows the prediction directions for a square block in VVC.

Figure 3 shows an example of top and left CU locations for deriving an Intra MPM list for different block shapes in VVC.

Figure 4 represents a decision tree illustrating the intra prediction signaling for luma in VVC.

Figure 5 represents a decision tree illustrating the intra prediction signaling for chroma in VVC.

Figure 6a illustrates an example of context surrounding the current block to be predicted using neural network.

Figure 6b illustrates an example of intra prediction via a fully-connected neural network in which various aspects of the embodiments may be implemented.

Figure 6c illustrates an example of intra prediction via a convolutional neural network in which various aspects of the embodiments may be implemented.

Figure 7 illustrates a generic encoding method according to at least one embodiment.

Figure 8 illustrates a generic decoding method according to at least one embodiment.

Figures 9, 10, 12, 13, 14, 15, 16, 17, 18 illustrate different variants of a neural network-based intra prediction method with side information in a video encoder and in a video decoder according to at least one embodiment.

Figure 11 illustrates a non-limiting example of the conversion operation of the stack of feature maps of bits into a bitstream according to at least one embodiment.

Figure 19 and 20 illustrate different variants of a decision tree illustrating the intra prediction signaling according to at least one embodiment.

5 Figure 21 illustrates a block diagram of an embodiment of video encoder in which various aspects of the embodiments may be implemented.

Figure 22 illustrates a block diagram of an embodiment of video decoder in which various aspects of the embodiments may be implemented.

10 Figure 23 illustrates a block diagram of an example apparatus in which various aspects of the embodiments may be implemented.

### DETAILED DESCRIPTION

15 We firstly introduce the intra prediction component of video codecs, then various embodiments of a deep intra predictor implemented in an encoder or a decoder according to the present principles are disclosed. Finally, various embodiments of an encoder or decoder in which the present principles are implemented are presented.

20 The present principles are described in the context on the video codec H.266/VVC as it is currently viewed as the best video codec in terms of compression performance. However, the present principles are compatible with any codec.

The intra prediction process in H.266/VVC consists of gathering reference samples, processing them, deriving the actual prediction of the samples of the current block, and finally post-processing the predicted samples.

**Figure 1** illustrates reference samples for intra prediction in H.266/VVC in the case of a square current block ( $W = H = N$ ), the pixel values at coordinates  $(x, y)$  being indicated in figure 1 by  $P(x, y)$ . An array of  $2W$  samples on the top is formed from the previously reconstructed top and top-right pixels to the current block,  $W$  denoting the block width. Similarly, a column of  $2H$  samples on the left is formed from the reconstructed left and below left pixels,  $H$  denoting the block height. The corner pixel at the top-left position is also used to fill up the gap between the top row and the left column references. If some of the samples on top or left are not available, because of the corresponding Coding Units (CUs) not being in the same slice or the current CU being at a frame boundary, then a method called reference sample substitution is performed where the missing samples are copied from the available samples in a clock-wise direction. Then, depending on the current CU size and the prediction mode, the reference samples are filtered using a specified filter.

25  
30  
35

H.266/VVC includes a range of prediction models derived from those in H.265/HEVC. Planar and DC prediction modes are used to predict smooth and gradually changing regions, whereas angular prediction modes are used to capture different directional structures. There exist 65 directional prediction modes which are organized differently for each rectangular block shape.

5 **Figure 2** illustrates the prediction direction for square blocks in H.266/VVC. The prediction modes correspond to different prediction directions as shown on Figure 2.

The intra prediction was further expanded with tools such as intra prediction with Multiple Reference Lines (MRL), Intra prediction with Sub-Partitions (ISP), and Matrix Intra-prediction (MIP). MIP is a set of intra prediction modes, each inferring a prediction of the current block  
 10 from reconstructed pixels via a linear transformation. For  $4 \times 4$  blocks, there exist 35 modes. For  $4 \times 8$ ,  $8 \times 4$ , and  $8 \times 8$  blocks, there are 19 modes. For the other blocks, 11 modes are used.

The intra prediction mode is signaled from the encoder to the decoder. For the luma channel,  
 15 the signaling of planar, DC, and the 65 directional modes are firstly described, omitting the signaling of MRL, that of ISP, and that of MIP. These last three will be detailed thereafter.

On the encoder side, the best intra prediction mode, for instance belonging to a set comprising the planar, DC, and the 65 directional modes, according to a rate-distortion criterion is selected, and its index is transmitted from the encoder to the decoder. To perform the signaling of the  
 20 selected mode index via entropy coding, a list of Most Probable Modes (MPMs) is built.

In VTM, an MPM list contains 6 intra prediction modes for signaling the intra prediction mode of the current block. The MPM list is created from the prediction modes of the intra coded CUs on the top and left of the current CU and some default modes. **Figure 3** shows the top and left CUs at the right and bottom edge of the current block, used for deriving the MPM list where:

25 L is the prediction mode of the left CU (value in range [0 – 66])

A is the prediction mode of the above CU (value in range [0 - 66]).

The deriving of the MPM list is as follows (with 2 variables offset = 61 and mod = 64):

Initialization of the MPM list:

MPM[0] = PLANAR\_IDX

30 MPM[1] = DC\_IDX

MPM[2] = VER\_IDX

MPM[3] = HOR\_IDX

MPM[4] = VER\_IDX - 4

MPM[5] = VER\_IDX + 4

35

if (L = A)

if (L > DC\_IDX)

```

    MPM[0] = PLANAR_IDX
    MPM[1] = L
    MPM[2] = ((L + offset) % mod) + 2
    MPM[3] = ((L - 1) % mod) + 2
5    MPM[4] = DC_IDX
    MPM[5] = ((L + offset - 1) % mod) + 2
else
    use initialized values
else
10    if ((L > DC_IDX) && (A > DC_IDX))
        MPM[0] = PLANAR_IDX
        MPM[1] = L
        MPM[2] = A
        MPM[3] = DC_IDX
15    MPM[4] = ((max(L,A) + offset) % mod) + 2 if L and A are not adjacent
            = ((max(L,A) + offset - 1) % mod) + 2, otherwise
    MPM[5] = ((max(L,A) - 1) % mod) + 2 if L and A are not adjacent
            = ((max(L,A) - 0) % mod) + 2 otherwise
    else if (L + A >= 2)
20    MPM[0] = PLANAR_IDX
        MPM[1] = max(L,A)
        MPM[2] = DC_IDX
        MPM[3] = ((max(L,A) + offset) % mod) + 2
        MPM[4] = ((max(L,A) - 1) % mod) + 2
25    MPM[5] = ((max(L,A) + offset - 1) % mod) + 2
    else
        use initialized values

```

Using circular adjacency over the range [2 – 66], it can be equivalently written

```

30    ((L + offset) % mod) + 2 ≡ L - 1
    ((L + offset - 1) % mod) + 2 ≡ L - 2
    ((L - 1) % mod) + 2 ≡ L + 1
    ((L - 0) % mod) + 2 ≡ L + 2

```

35 Using the above relationships, it can be shown that the MPM list derivation is that in Table 1

**Table 1: MPM derivation in VTM. A and L denote the predictions modes of above and left CUs respectively**

| Conditions |                           |            | MPM[0]     | MPM[1]   | MPM[2]  | MPM[3]     | MPM[4]                                              | MPM[5]                                              |
|------------|---------------------------|------------|------------|----------|---------|------------|-----------------------------------------------------|-----------------------------------------------------|
| L=A        | L≠PLANAR_IDX and L≠DC_IDX |            | PLANAR_IDX | L        | L-1     | L+1        | DC_IDX                                              | L-2                                                 |
|            | otherwise                 |            | PLANAR_IDX | DC_IDX   | VER_IDX | HOR_IDX    | VER_IDX-4                                           | VER_IDX+4                                           |
| L≠A        | L > DC_IDX and A > DC_IDX |            | PLANAR_IDX | L        | A       | DC_IDX     | max(L,A)-2, if L and A are adjacent else max(L,A)-1 | max(L,A)+2, if L and A are adjacent else max(L,A)+1 |
|            | otherwise                 | L + A >= 2 | PLANAR_IDX | max(L,A) | DC_IDX  | max(L,A)-1 | max(L,A)+1                                          | max(L,A)- 2                                         |
|            |                           | otherwise  | PLANAR_IDX | DC_IDX   | VER_IDX | HOR_IDX    | VER_IDX-4                                           | VER_IDX+4                                           |

If the selected intra prediction mode for predicting the current block corresponds to one of the six MPM modes, this is signaled via the *mpmFlag* with value 1 and then by signaling the candidate mode from the MPM list using the variable length coding scheme shown in Table 2.

- 5 Otherwise, the *mpmFlag* is equal to 0 and the candidate index in the set of remaining 61 modes is truncated binary encoded with either 5 or 6 bits.

**Table 2: MPM signaling in VTM**

| Candidate Index | Code  |
|-----------------|-------|
| MPM[0]          | 0     |
| MPM[1]          | 10    |
| MPM[2]          | 110   |
| MPM[3]          | 1110  |
| MPM[4]          | 11110 |
| MPM[5]          | 11111 |

For intra prediction with MRL, the reference line used for the prediction is signaled with a flag *multiRefIdx*. The valid values of *multiRefIdx* are 0, 1, and 3, which signal the first reference line, the second reference line, and the fourth reference line respectively. When *multiRefIdx* is non-zero, meaning either the second or the fourth reference line is used, the prediction mode always belongs to the MPM list. Thus, the *mpmFlag* is not signaled. Furthermore, planar is excluded from the list. This means that, when *multiRefIdx* is non-zero, only five prediction modes are available as possible candidates. When *multiRefIdx* is non-zero, the prediction mode is signaled as shown in Table 3.

**Table 3: MPM signaling when multiRefIdx > 0 in VTM**

| Candidate index | Code |
|-----------------|------|
| MPM[1]          | 0    |
| MPM[2]          | 10   |
| MPM[3]          | 110  |
| MPM[4]          | 1110 |
| MPM[5]          | 1111 |

For ISP, the type of partitioning used for the CU is signaled with a flag called *ispMode*. *ispMode* is encoded only when *multiRefIdx* is equal to 0. The valid values of *ispMode* are 0, 1, and 2, which signal no partitioning, horizontal partitioning, and vertical partitioning respectively.

Now regarding MIP, a MIP mode is first signaled with a flag called *mipFlag*, a value of 1 meaning that a MIP mode is used for predicting the current block, and 0 meaning that one of the 67 intra prediction modes is used. When *mipFlag* is equal to 1, *multiRefIdx* is necessarily equal to 0, meaning that the first reference line is used, and *ispMode* is equal to 0, i.e. there is no target CU partition. Therefore, when *mipFlag* is equal to 1, *multiRefIdx* and *ispMode* are not written to the bitstream. If *mipFlag* is equal to 1, the index of the selected MIP mode is then truncated binary encoded in the latest versions of VTM.

To handle the case where the intra prediction mode for predicting the current block is one of the 67 intra prediction modes and the selected mode for predicting the top CU or the one for predicting the left CU is a MIP mode, a mapping between each MIP mode and one of the conventional modes enables to substitute this MIP mode with its mapped conventional mode. In the latest versions of VTM, any MIP mode is mapped to planar.

**Figure 4** represents a decision tree illustrating the intra prediction signaling for luma in VVC.

In Figure 4, a flag in light gray indicates that the value of the flag is deduced from the value of the previous flags written to the bitstream on the encoder side and read from the bitstream on the decoder side. This means that the flags in light gray are not written to the bitstream on the encoder side; they are not read from the bitstream on the decoder side.

It is essential to mention that some flags are CU-level whereas the others are PU-level. For a CU-level flag, the value of the flag applies to the PU(s) in the CU. For a PU-level flag, each PU in the CU has its own flag value. If a flag is PU-level, in Figure 4, when the decision tree reaches the binary decision associated to this flag, the value of this flag for each PU is written to the bitstream before moving on to the subsequent binary decision. Table 4 shows the level of each flag represented in Figure 4.

**Table 4: level of each flag involved in the intra prediction signaling in VTM**

| Flag | Level |
|------|-------|
|------|-------|

|                      |    |
|----------------------|----|
| <i>mipFlag</i>       | CU |
| <i>multiRefIndex</i> | PU |
| <i>ispMode</i>       | CU |
| <i>mpmFlag</i>       | PU |

For the two chroma channels, neither MRL nor ISP nor MIP is used. However, two specific tools are used: the direct mode and Cross-Component Linear Model (CCLM).

**Figure 5** represents a decision tree illustrating the intra prediction signaling for chroma in VVC.

- 5 The direct mode corresponds to the application of the selected mode for predicting the collocated luma block to the prediction of the current chroma block. If the *directFlag* is equal to 1, the direct mode is selected for predicting the current chroma block. Otherwise, the *directFlag* is equal to 0, and one mode in the list  $L = [\text{planar}, \text{vertical}, \text{horizontal}, \text{DC}]$  is selected. If a mode in  $L$  is equal to the direct mode, this mode is replaced by the mode of index 66. In
- 10 CCLM, a linear model predicts the current chroma block from the reconstructed luma reference samples surrounding the collocated luma block. The parameters of the linear model are derived from the reconstructed reference samples. There exist three CCLM modes, each associated to a different derivation of the parameters. If the *cclmFlag* is equal to 1, one of the three CCLM mode is selected. In this case, the *directFlag* is not written to the bitstream.
- 15 Otherwise, the *cclmFlag* is equal to 0, and either the direct mode or one of the modes in  $L$  is selected.

As previously explained, recent video codec introduces neural network-based intra prediction.

- 20 A deep intra predictor infers a prediction from the context surrounding the current block to be predicted. **Figure 6a** illustrates an example of context surrounding a current block to be predicted using neural network. The context  $X_c$  is composed of reconstructed pixels on top and top-right of the current block  $Y$  and on its left side and below-left side, similarly to the set of reconstructed reference samples for the intra prediction in H.266/VVC. But, unlike it, the context  $X_c$  is extended towards the left and the top as shown on Figure 6a. Thanks to this
- 25 extension, the deep intra predictor can learn a relationship between the spatial correlations in its input context and the prediction it gives. Note that the subscript “c” in  $X_c$  indicates that the reconstructed pixels in the context have already been preprocessed, as detailed later.

- Figure 6b** illustrates an example of intra prediction via a fully-connected neural network in which various aspects of the embodiments may be implemented. If the deep intra predictor is
- 30 fully-connected, the context is typically flattened into a vector, and the resulting vector is fed into the neural network. Then, the vector provided by the neural network is reshaped to the

shape of the current block, yielding the prediction  $\hat{Y}_c$  as shown on Figure 6b. Note that the subscript “c” in  $\hat{Y}_c$  indicates that the predicted pixels have not been post-processed yet, which will be explained later.

**Figure 6c** illustrates an example of intra prediction via a convolutional neural network in which various aspects of the embodiments may be implemented. If the deep intra predictor is convolutional, the context can be split into two portions. Then, each portion is fed into a stack of convolutional layers. The two stacks of feature maps at the output of the two stacks of convolutional layers are merged via full connectivity. Finally, the result of the merge is inserted into a stack of transpose convolutional layers, yielding the prediction  $\hat{Y}_c$ . During the preprocessing step, the context  $X_c$  can be obtained by subtracting from the original context  $X$  of reconstructed pixels its mean pixel intensity  $\alpha$ ,

$$X_c = X - \alpha$$

Then, during the post-processing step, the post-processed prediction  $\hat{Y}$  of the current block  $Y$  is computed by adding to the prediction  $\hat{Y}_c$  the mean pixel intensity and clipping,

$$\hat{Y} = \min(\max(\hat{Y}_c + \alpha, 0), (1 \ll b) - 1)$$

$b$  denotes the pixel bit depth. As an alternative,  $\alpha$  can also be the mean pixel intensity over a large set of training images.

In video codecs such as H.265/HEVC and H.266/VVC, the image is split into Coding Tree Units (CTUs). The CTUs are processed one at a time, in raster-scan order. Each CTU can be split hierarchically into Coding Units (CUs). The CUs in a CTU are processed in Z-scan order. Consequently, in H.265/HEVC for instance, the size of a block to be predicted can be either  $64 \times 64$ ,  $32 \times 32$ ,  $16 \times 16$ ,  $8 \times 8$  or  $4 \times 4$ . This means that 5 neural networks are needed, one for each size of block to be predicted. The deep intra predictor mode is thus made of the 5 neural networks. In H.266/VVC, as the hierarchical splitting is more sophisticated, a block to be predicted can be of size either  $128 \times 128$ ,  $64 \times 64$ ,  $32 \times 32$ ,  $16 \times 16$ ,  $8 \times 8$  or  $4 \times 4$ . Besides, it can also be rectangular, e.g. of size  $4 \times 8$ . In this case, a solution is to assign one neural network per block size to build the deep neural network mode.

In the different works integrating a deep intra prediction mode into a video codec, usually H.265/HEVC, the deep intra prediction mode is systematically in competition with the existing ones. For the current block to be predicted, a flag is written to the bitstream before all the other flags for intra prediction. The value 1 indicates that the deep intra prediction mode is selected

for predicting the current block. In this case, no other flag for intra prediction is written to the bitstream. The value 0 means that one of the regular intra prediction is selected. In this case, the regular flags for intra prediction are then written to the bitstream.

Note that, the above-mentioned signaling has been implemented in H.265/HEVC. No approach  
5 has been proposed yet in H.266/VVC. Notably, it is not clear yet how to handle the flags *mipFlag*, *multiRefIdx*, and *ispMode* when the deep neural network mode is selected.

According to at least one generic embodiment of the present principles, instead of having a deep neural network mode competing with the existing intra prediction modes, the deep neural  
10 network mode alone takes over the intra prediction component of the video codec. According to a particular feature, the deep neural networks generates side information which is transmitted from the encoder to the decoder. According to another particular feature, on the decoder side, the deep intra predictor infers a prediction of the current block from the context surrounding the current block to be predicted and the side information.

15 **Figure 7** illustrates a generic encoding method according to at least one embodiment. As detailed hereafter with the various implementations, the encoding method 10 comprises, once the picture of a video is partitioned into block to encode, determining 12 a neural network intra prediction and side information using a neural network from at least one input data. At the encoding, the neural network uses as input not only the context surrounding the block being  
20 encoded, but also to the block itself. Then, based on the neural network intra prediction, the block is encoded 16 wherein the encoding further comprises, as non-limiting examples of steps, obtaining a block of residuals that is then transformed and quantized. In a step 14, the side information of the intra prediction neural network is also encoded and transmitted to the decoder thus advantageously allowing the decoder to avoid the "failure case" in the neural  
25 network intra prediction. Steps 14 and 16 are performed in any order or in parallel.

**Figure 8** illustrates a generic decoding method according to at least one embodiment. As detailed hereafter with the various implementations, the decoding method 20 comprises, obtaining 22, for a block being decoded in a picture of a video, a side information for a neural network intra prediction. According to a non-limiting example, the side information is received  
30 and decoded using optionally transform and quantization as block data. Then, in a step 24, a neural network intra prediction is determined using a neural network which uses as input the side information along with other input data, such as, the context surrounding the block being decoded wherein the context comprises available pixels being previously reconstructed in the video picture. As with classic decoding method, the block is reconstructed 26 based on the  
35 neural network intra prediction.

In the following different variants of a neural network-based intra prediction method based on side information in a video encoder and in a video decoder according to at least one

embodiment is presented. Accordingly, different variants of the encoding method of Figure 7 and decoding method of Figure 8 are thus illustrated in the following. The neural network architecture is split into two parts: an encoder part and a decoder part. In the encoder part, the neural network has access not only to the context  $X_c$  surrounding the current luma block  $Y$ , but also to  $Y$ . That is why the encoder architecture takes them as input to generate side information. The side information is written to the bitstream. In the decoder part, the neural network takes the side information read from the bitstream and the context  $X_c$  to provide a prediction  $\hat{Y}_c$  of the current luma block. Figure 9 and Figure 10 illustrate two non-limiting examples of instantiations of the above-mentioned proposition for fully-connected and convolutional architectures respectively. In both cases, the last layer of the encoder architecture contains the sigmoid non-linearity.

**Figure 9** illustrates deep intra predictor generating side information with fully-connected architecture for predicting luma blocks. The horizontal dashed line delineates the encoder and decoder parts of the architecture. The broken dashed line crosses the frontier between the encoder side and the decoder side as the context is available on both sides. For the fully-connected architecture, the side information is thus a vector  $Z$  whose coefficients belong to  $[0, 1]$ . By applying an elementwise threshold of 0.5 to  $Z$ , this vector is converted into a vector of bits, which is written to the bitstream

**Figure 10** illustrates a deep intra predictor generating side information with convolutional architecture for predicting luma blocks. The horizontal dashed line delineates the encoder and decoder parts of the architecture. The broken dashed line crosses the frontier between the encoder side and the decoder side as the context is available on both sides. For the convolutional architecture,  $Z$  is a stack of feature maps whose coefficients belong to  $[0, 1]$ . The same threshold enables to convert it into a stack of feature maps of bits, which is written to the bitstream in raster-scan order. **Figure 11** illustrates a non-limiting example of the conversion operation of the stack of feature maps of bits into a bitstream wherein the bits are written into or read from the bitstream in raster scan order. In this case, on the decoder side, as the bitstream is read, the stack of feature maps of bits is reconstructed in raster-scan order.

According to a first variant, the values of the output vector  $Z$  are unrestricted and belong to  $\mathbb{R}$ .

**Figure 12** illustrates an example of deep intra predictor generating side information with fully-connected architecture in the case of the first variant. **Figure 13** illustrates an example of deep intra predictor generating side information with convolutional architecture in the case of the first variant. In this variant, the sigmoid non-linearity is removed from the last layer of the encoder part of the deep intra predictor generating side information. As this layer has no non-linearity, the coefficients in its output  $Z$  belong to  $\mathbb{R}$ . As for the transform coefficients in

H.265/HEVC and H.266/VVC, the coefficients in  $Z$  are quantized via a uniform scalar quantization as shown on Figures 12 and 13. Then, the sign of each quantized coefficient is encoded losslessly and the absolute value is encoded losslessly via binary arithmetic coding. A CABAC context model can be used for the binary arithmetic coding of the absolute value of the quantized coefficients.

According to a second variant, the values of the output vector  $Z$  are binarized and belong to  $[-1, 1]$ . In this second variant, the sigmoid non-linearity in the last layer of the encoder part of the architecture is replaced by a tangent-hyperbolic non-linearity. The coefficients in its output  $Z$  thus belong to  $[-1, 1]$ . According to yet another second variant, the output vector is converted into a vector of bits by applying an element wise threshold of 0.5 to the vector. In this case, an elementwise threshold at 0 is applied to  $Z$ , yielding a vector of coefficients in  $\{-1, 1\}$  for the fully-connected architecture and a stack of feature maps whose coefficients belong to  $\{-1, 1\}$  for the convolutional architecture. When writing to the bitstream and reading from it, each coefficient equal to  $-1$  is mapped to 0 and conversely.

**Figure 14** illustrates a deep intra predictor generating side information with fully-connected architecture for predicting chroma blocks. **Figure 15** illustrates a deep intra predictor generating side information with convolutional architecture for predicting chroma blocks. The notations are the same as for the previous embodiments. The encoder part of the deep neural network is fed with the context  $X_c^{CbCr}$  surrounding the current chroma block  $Y^{CbCr}$ , the context  $X_c^Y$  surrounding the luma block  $Y^Y$  that is collocated with the current chroma block, and the current chroma block  $Y^{CbCr}$ , to generate side information. The decoder part of the deep neural network takes the side information read from the bitstream,  $X_c^{CbCr}$ , and  $X_c^Y$  to provide a prediction  $\hat{Y}_c^{CbCr}$  of  $Y^{CbCr}$ . The adaptation of Figure 9 to the case of a given chroma block to be predicted in 4:2:0 is shown in Figure 14. The adaptation of Figure 10 to the case of a given chroma block to be predicted in 4:2:0 is depicted in Figure 15.

According to a first variant of the prediction of chroma blocks, the prediction of the chroma blocks relies on the luma context exclusively. In the first variant, only the context  $X_c^Y$  surrounding the luma block  $Y^Y$  that is collocated with the current chroma block and the current chroma block  $Y^{CbCr}$  are fed into the encoder part of the deep neural network. The decoder part of the deep neural network takes the side information read from the bitstream and  $X_c^Y$  to give a prediction  $\hat{Y}_c^{CbCr}$  of  $Y^{CbCr}$ .

According to a second variant of the prediction of chroma blocks, the prediction of the chroma blocks relies on reconstructed luma block. When predicting the current chroma block, the luma block that is collocated with the current chroma block is already encoded and decoded. Figure 16 illustrates an example of a deep intra predictor generating side information with fully-connected architecture for predicting the current chroma block when the reconstruction of the luma block that is collocated with the current chroma block is fed into the deep intra predictor on both the encoder and decoder sides. In this second variant, the encoder part of the deep intra predictor takes the context  $X_c^{CbCr}$  surrounding the current chroma block, the context  $X_c^Y$  surrounding the luma block that is collocated with the current chroma block, the current chroma block  $Y^{CbCr}$ , and the reconstruction  $\tilde{Y}^Y$  of this luma block to generate side information. The decoder part of the deep neural network takes the side information read from the bitstream,  $X_c^{CbCr}$ ,  $X_c^Y$ , and  $\tilde{Y}^Y$  to provide a prediction  $\hat{Y}_c^{CbCr}$  of  $Y^{CbCr}$ .

According to yet another variant, the intra prediction modes of the top and left CUs is also used as input data for predicting luma blocks or chroma blocks. In this variant, the intra prediction mode ( $L$ ) of the block located on the left side of the block being encoded or decoded and the intra prediction mode ( $A$ ) of the block located above the block being encoded or decoded are used as input data to the neural network. It is possible to insert into the architectures presented in Figures 9 and 10, the intra prediction mode of the CU located on the left side of the current CU and the intra prediction mode of the CU located above the current CU, which are denoted  $L$  and  $A$  in Figure 3.

**Figure 17** illustrates an example of a deep intra predictor generating side information with fully-connected architecture for predicting luma blocks when the intra prediction mode of the CU located on the left side of the current CU and the intra prediction mode of the CU located above the current CU are fed into the deep intra predictor.

**Figure 18** illustrates an example of a deep intra predictor generating side information with convolutional architecture for predicting luma blocks when the intra prediction mode of the CU located on the left side of the current CU and the intra prediction mode of the CU located above the current CU are fed into the deep intra predictor.

These two intra prediction modes  $L$  and  $A$  are fed into the architectures on both the encoder and decoder sides. This way, on the encoder side, the deep neural network can generate more compressible side information by removing from it some redundancies between the side information it is computing and  $\{A, L\}$ .

According to yet another advantageous characteristic, the enabling/disabling of a deep intra predictor generating side information is signaled from the encoder to the decoder. As the deep

intra predictor generating side information advantageously allows to handle alone the intra prediction, it replaces the 67 intra prediction modes, MRL, ISP, and MIP. This implies that, on the encoder side, there is no more selection of the best intra prediction mode according to a rate-distortion criterion. In this variant, the prediction is always determined according to the deep intra prediction mode. At a given step of the hierarchical partitioning in H.266/VVC, for a given luma block to be predicted, the deep intra predictor is systematically selected for predicting the current luma block. The signaling cost of intra prediction comes from the side information the deep intra predictor generates exclusively.

In a first variant of the above approach, the deep intra prediction mode is still in competition with intra prediction modes in H.266/VVC. It replaces MIP in H.266/VVC. This means that MIP is suppressed and the *mipFlag* becomes the *deepFlag*. **Figure 19** illustrates a non-limiting example of a decision tree illustrating the intra prediction signaling according to at least one embodiment. If the *deepFlag* is equal to 1, the deep intra predictor is selected for predicting the current luma block. Otherwise, the *deepFlag* is equal to 0, and one of the 67 intra prediction modes is used. When the *deepFlag* is equal to 1, *multiRefIdx* is necessarily equal to 0, meaning that the first reference line is used, and *ispMode* is equal to 0, i.e. there is no target CU partition. Therefore, when the *deepFlag* is equal to 1, *multiRefIdx* and *ispMode* are not written to the bitstream. The *deepFlag* has a CABAC context model with two neighbors: the intra prediction mode selected for predicting the left PU and the intra prediction mode for selecting the above PU. **Figure 19** summarizes the sequence of decisions for the intra prediction signaling via the decision tree. Here, the *deepFlag* is CU-level.

In a second variant of the above approach, the intra prediction in H.266/VVC is based on the deep intra predictor and planar exclusively. **Figure 20** illustrates a non-limiting example of a decision tree illustrating the intra prediction signaling according to this variant. This time, the rate-distortion optimization is kept, but its candidates are limited to planar and the deep intra prediction mode. During the rate-distortion optimization, the fast pass is suppressed, meaning that the two candidates are systematically tested for the current luma block to be predicted. The *deepFlag* is written to the bitstream to indicate the selection of the deep intra prediction mode for predicting the current luma block. If the *deepFlag* is equal 1, the deep intra predictor is selected. Then, the side information it generates is written to the bitstream. Otherwise, planar is selected as shown on **Figure 20**. Note that, in this second variant, MRL and ISP are removed. As with the first variant, the *deepFlag* has a CABAC context model with two neighbors.

As for luma, as the deep intra predictor generating side information advantageously allows to handle alone the intra prediction, it replaces CCLM, the direct mode, and the four other modes.

At a given step of the hierarchical partitioning in H.266/VVC, for a given chroma block to be predicted, the deep intra predictor is systematically selected for predicting the current chroma block. The signaling cost of intra prediction comes from the side information the deep intra predictor generates exclusively.

5

In a first variant, the deep intra prediction mode is still in competition with the other tools for chroma intra prediction in H.266/VVC. The intra prediction signaling for chroma is the one in Figure 5. In the list of four modes for chroma prediction  $L$ , the horizontal mode is replaced by the deep neural network mode. If the direct mode is equal to one of the four modes in  $L$ , this mode is replaced by the horizontal mode.

10

This application describes a variety of aspects, including tools, features, embodiments, models, approaches, etc. Many of these aspects are described with specificity and, at least to show the individual characteristics, are often described in a manner that may sound limiting. However, this is for purposes of clarity in description, and does not limit the application or scope of those aspects. Indeed, all of the different aspects can be combined and interchanged to provide further aspects. Moreover, the aspects can be combined and interchanged with aspects described in earlier filings as well.

15

The aspects described and contemplated in this application can be implemented in many different forms. **Figures 21, 22 and 23** below provide some embodiments, but other embodiments are contemplated and the discussion of Figures 21, 22 and 23 does not limit the breadth of the implementations. At least one of the aspects generally relates to video encoding and decoding, and at least one other aspect generally relates to transmitting a bitstream generated or encoded. These and other aspects can be implemented as a method, an apparatus, a computer readable storage medium having stored thereon instructions for encoding or decoding video data according to any of the methods described, and/or a computer readable storage medium having stored thereon a bitstream generated according to any of the methods described.

20

25

In the present application, the terms “reconstructed” and “decoded” may be used interchangeably, the terms “pixel” and “sample” may be used interchangeably, the terms “image,” “picture” and “frame” may be used interchangeably. Usually, but not necessarily, the term “reconstructed” is used at the encoder side while “decoded” is used at the decoder side.

30

Various methods are described herein, and each of the methods comprises one or more steps or actions for achieving the described method. Unless a specific order of steps or actions is

required for proper operation of the method, the order and/or use of specific steps and/or actions may be modified or combined.

Various methods and other aspects described in this application can be used to modify modules, for example, the intra prediction, entropy coding, and/or decoding modules (160, 360, 145, 330), of a video encoder 100 and decoder 200 as shown in FIG. 21 and FIG. 22. Moreover, the present aspects are not limited to VVC or HEVC, and can be applied, for example, to other standards and recommendations, whether pre-existing or future-developed, and extensions of any such standards and recommendations (including VVC and HEVC). Unless indicated otherwise, or technically precluded, the aspects described in this application can be used individually or in combination.

Various numeric values are used in the present application, for example, number of intra prediction mode, parameters of the neural networks (layer, output range, threshold). The specific values are for example purposes and the aspects described are not limited to these specific values.

FIG. 21 illustrates an encoder 100. Variations of this encoder 100 are contemplated, but the encoder 100 is described below for purposes of clarity without describing all expected variations.

Before being encoded, the video sequence may go through pre-encoding processing (101), for example, applying a color transform to the input color picture (e.g., conversion from RGB 4:4:4 to YCbCr 4:2:0), or performing a remapping of the input picture components in order to get a signal distribution more resilient to compression (for instance using a histogram equalization of one of the color components). Metadata can be associated with the pre-processing, and attached to the bitstream.

In the encoder 100, a picture is encoded by the encoder elements as described below. The picture to be encoded is partitioned (102) and processed in units of, for example, CUs. Each unit is encoded using, for example, either an intra or inter mode. When a unit is encoded in an intra mode, it performs intra prediction (160). In an inter mode, motion estimation (175) and compensation (170) are performed. The encoder decides (105) which one of the intra mode or inter mode to use for encoding the unit, and indicates the intra/inter decision by, for example, a prediction mode flag. Prediction residuals are calculated, for example, by subtracting (110) the predicted block from the original image block.

The prediction residuals are then transformed (125) and quantized (130). The quantized

transform coefficients, as well as motion vectors and other syntax elements, are entropy coded (145) to output a bitstream. The encoder can skip the transform and apply quantization directly to the non-transformed residual signal. The encoder can bypass both transform and quantization, i.e., the residual is coded directly without the application of the transform or quantization processes.

The encoder decodes an encoded block to provide a reference for further predictions. The quantized transform coefficients are de-quantized (140) and inverse transformed (150) to decode prediction residuals. Combining (155) the decoded prediction residuals and the predicted block, an image block is reconstructed. In-loop filters (165) are applied to the reconstructed picture to perform, for example, deblocking/SAO (Sample Adaptive Offset) filtering to reduce encoding artifacts. The filtered image is stored at a reference picture buffer (180).

FIG. 22 illustrates a block diagram of a video decoder 200. In the decoder 200, a bitstream is decoded by the decoder elements as described below. Video decoder 200 generally performs a decoding pass reciprocal to the encoding pass as described in FIG. 21. The encoder 100 also generally performs video decoding as part of encoding video data.

In particular, the input of the decoder includes a video bitstream, which can be generated by video encoder 100. The bitstream is first entropy decoded (230) to obtain transform coefficients, motion vectors, and other coded information. The picture partition information indicates how the picture is partitioned. The decoder may therefore divide (235) the picture according to the decoded picture partitioning information. The transform coefficients are de-quantized (240) and inverse transformed (250) to decode the prediction residuals. Combining (255) the decoded prediction residuals and the predicted block, an image block is reconstructed. The predicted block can be obtained (270) from intra prediction (260) or motion-compensated prediction (i.e., inter prediction) (275). In-loop filters (265) are applied to the reconstructed image. The filtered image is stored at a reference picture buffer (280).

The decoded picture can further go through post-decoding processing (285), for example, an inverse color transform (e.g. conversion from YCbCr 4:2:0 to RGB 4:4:4) or an inverse remapping performing the inverse of the remapping process performed in the pre-encoding processing (101). The post-decoding processing can use metadata derived in the pre-encoding processing and signaled in the bitstream.

FIG. 23 illustrates a block diagram of an example of a system in which various aspects and embodiments are implemented. System 1000 can be embodied as a device including the various components described below and is configured to perform one or more of the aspects

described in this document. Examples of such devices, include, but are not limited to, various electronic devices such as personal computers, laptop computers, smartphones, tablet computers, digital multimedia set top boxes, digital television receivers, personal video recording systems, connected home appliances, and servers. Elements of system 1000, singly or in combination, can be embodied in a single integrated circuit (IC), multiple ICs, and/or discrete components. For example, in at least one embodiment, the processing and encoder/decoder elements of system 1000 are distributed across multiple ICs and/or discrete components. In various embodiments, the system 1000 is communicatively coupled to one or more other systems, or other electronic devices, via, for example, a communications bus or through dedicated input and/or output ports. In various embodiments, the system 1000 is configured to implement one or more of the aspects described in this document.

The system 1000 includes at least one processor 1010 configured to execute instructions loaded therein for implementing, for example, the various aspects described in this document. Processor 1010 can include embedded memory, input output interface, and various other circuitries as known in the art. The system 1000 includes at least one memory 1020 (e.g., a volatile memory device, and/or a non-volatile memory device). System 1000 includes a storage device 1040, which can include non-volatile memory and/or volatile memory, including, but not limited to, Electrically Erasable Programmable Read-Only Memory (EEPROM), Read-Only Memory (ROM), Programmable Read-Only Memory (PROM), Random Access Memory (RAM), Dynamic Random Access Memory (DRAM), Static Random Access Memory (SRAM), flash, magnetic disk drive, and/or optical disk drive. The storage device 1040 can include an internal storage device, an attached storage device (including detachable and non-detachable storage devices), and/or a network accessible storage device, as non-limiting examples.

System 1000 includes an encoder/decoder module 1030 configured, for example, to process data to provide an encoded video or decoded video, and the encoder/decoder module 1030 can include its own processor and memory. The encoder/decoder module 1030 represents module(s) that can be included in a device to perform the encoding and/or decoding functions. As is known, a device can include one or both of the encoding and decoding modules. Additionally, encoder/decoder module 1030 can be implemented as a separate element of system 1000 or can be incorporated within processor 1010 as a combination of hardware and software as known to those skilled in the art.

Program code to be loaded onto processor 1010 or encoder/decoder 1030 to perform the various aspects described in this document can be stored in storage device 1040 and subsequently loaded onto memory 1020 for execution by processor 1010. In accordance with various embodiments, one or more of processor 1010, memory 1020, storage device 1040,

and encoder/decoder module 1030 can store one or more of various items during the performance of the processes described in this document. Such stored items can include, but are not limited to, the input video, the decoded video or portions of the decoded video, the bitstream, matrices, variables, and intermediate or final results from the processing of equations, formulas, operations, and operational logic.

In some embodiments, memory inside of the processor 1010 and/or the encoder/decoder module 1030 is used to store instructions and to provide working memory for processing that is needed during encoding or decoding. In other embodiments, however, a memory external to the processing device (for example, the processing device can be either the processor 1010 or the encoder/decoder module 1030) is used for one or more of these functions. The external memory can be the memory 1020 and/or the storage device 1040, for example, a dynamic volatile memory and/or a non-volatile flash memory. In several embodiments, an external non-volatile flash memory is used to store the operating system of, for example, a television. In at least one embodiment, a fast external dynamic volatile memory such as a RAM is used as working memory for video coding and decoding operations, such as for MPEG-2 (MPEG refers to the Moving Picture Experts Group, MPEG-2 is also referred to as ISO/IEC 13818, and 13818-1 is also known as H.222, and 13818-2 is also known as H.262), HEVC (HEVC refers to High Efficiency Video Coding, also known as H.265 and MPEG-H Part 2), or VVC (Versatile Video Coding, a new standard being developed by JVET, the Joint Video Experts Team).

The input to the elements of system 1000 can be provided through various input devices as indicated in block 1130. Such input devices include, but are not limited to, (i) a radio frequency (RF) portion that receives an RF signal transmitted, for example, over the air by a broadcaster, (ii) a Component (COMP) input terminal (or a set of COMP input terminals), (iii) a Universal Serial Bus (USB) input terminal, and/or (iv) a High Definition Multimedia Interface (HDMI) input terminal. Other examples, not shown in FIG. 23, include composite video.

In various embodiments, the input devices of block 1130 have associated respective input processing elements as known in the art. For example, the RF portion can be associated with elements suitable for (i) selecting a desired frequency (also referred to as selecting a signal, or band-limiting a signal to a band of frequencies), (ii) downconverting the selected signal, (iii) band-limiting again to a narrower band of frequencies to select (for example) a signal frequency band which can be referred to as a channel in certain embodiments, (iv) demodulating the downconverted and band-limited signal, (v) performing error correction, and (vi) demultiplexing to select the desired stream of data packets. The RF portion of various embodiments includes one or more elements to perform these functions, for example, frequency selectors, signal selectors, band-limiters, channel selectors, filters, downconverters, demodulators, error

correctors, and demultiplexers. The RF portion can include a tuner that performs various of these functions, including, for example, downconverting the received signal to a lower frequency (for example, an intermediate frequency or a near-baseband frequency) or to baseband. In one set-top box embodiment, the RF portion and its associated input processing  
5 element receives an RF signal transmitted over a wired (for example, cable) medium, and performs frequency selection by filtering, downconverting, and filtering again to a desired frequency band. Various embodiments rearrange the order of the above-described (and other) elements, remove some of these elements, and/or add other elements performing similar or different functions. Adding elements can include inserting elements in between existing  
10 elements, such as, for example, inserting amplifiers and an analog-to-digital converter. In various embodiments, the RF portion includes an antenna.

Additionally, the USB and/or HDMI terminals can include respective interface processors for connecting system 1000 to other electronic devices across USB and/or HDMI connections. It is to be understood that various aspects of input processing, for example, Reed-Solomon error  
15 correction, can be implemented, for example, within a separate input processing IC or within processor 1010 as necessary. Similarly, aspects of USB or HDMI interface processing can be implemented within separate interface ICs or within processor 1010 as necessary. The demodulated, error corrected, and demultiplexed stream is provided to various processing elements, including, for example, processor 1010, and encoder/decoder 1030 operating in  
20 combination with the memory and storage elements to process the datastream as necessary for presentation on an output device.

Various elements of system 1000 can be provided within an integrated housing. Within the integrated housing, the various elements can be interconnected and transmit data therebetween using suitable connection arrangement 1140, for example, an internal bus as  
25 known in the art, including the Inter-IC (I2C) bus, wiring, and printed circuit boards.

The system 1000 includes communication interface 1050 that enables communication with other devices via communication channel 1060. The communication interface 1050 can include, but is not limited to, a transceiver configured to transmit and to receive data over communication channel 1060. The communication interface 1050 can include, but is not  
30 limited to, a modem or network card and the communication channel 1060 can be implemented, for example, within a wired and/or a wireless medium.

Data is streamed, or otherwise provided, to the system 1000, in various embodiments, using a wireless network such as a Wi-Fi network, for example IEEE 802.11 (IEEE refers to the Institute of Electrical and Electronics Engineers). The Wi-Fi signal of these embodiments is

received over the communications channel 1060 and the communications interface 1050 which are adapted for Wi-Fi communications. The communications channel 1060 of these embodiments is typically connected to an access point or router that provides access to external networks including the Internet for allowing streaming applications and other over-the-top communications. Other embodiments provide streamed data to the system 1000 using a set-top box that delivers the data over the HDMI connection of the input block 1130. Still other embodiments provide streamed data to the system 1000 using the RF connection of the input block 1130. As indicated above, various embodiments provide data in a non-streaming manner. Additionally, various embodiments use wireless networks other than Wi-Fi, for example a cellular network or a Bluetooth network.

The system 1000 can provide an output signal to various output devices, including a display 1100, speakers 1110, and other peripheral devices 1120. The display 1100 of various embodiments includes one or more of, for example, a touchscreen display, an organic light-emitting diode (OLED) display, a curved display, and/or a foldable display. The display 1100 can be for a television, a tablet, a laptop, a cell phone (mobile phone), or other device. The display 1100 can also be integrated with other components (for example, as in a smart phone), or separate (for example, an external monitor for a laptop). The other peripheral devices 1120 include, in various examples of embodiments, one or more of a stand-alone digital video disc (or digital versatile disc) (DVR, for both terms), a disk player, a stereo system, and/or a lighting system. Various embodiments use one or more peripheral devices 1120 that provide a function based on the output of the system 1000. For example, a disk player performs the function of playing the output of the system 1000.

In various embodiments, control signals are communicated between the system 1000 and the display 1100, speakers 1110, or other peripheral devices 1120 using signaling such as AV.Link, Consumer Electronics Control (CEC), or other communications protocols that enable device-to-device control with or without user intervention. The output devices can be communicatively coupled to system 1000 via dedicated connections through respective interfaces 1070, 1080, and 1090. Alternatively, the output devices can be connected to system 1000 using the communications channel 1060 via the communications interface 1050. The display 1100 and speakers 1110 can be integrated in a single unit with the other components of system 1000 in an electronic device such as, for example, a television. In various embodiments, the display interface 1070 includes a display driver, such as, for example, a timing controller (T Con) chip.

The display 1100 and speaker 1110 can alternatively be separate from one or more of the other components, for example, if the RF portion of input 1130 is part of a separate set-top

box. In various embodiments in which the display 1100 and speakers 1110 are external components, the output signal can be provided via dedicated output connections, including, for example, HDMI ports, USB ports, or COMP outputs.

The embodiments can be carried out by computer software implemented by the processor 1010 or by hardware, or by a combination of hardware and software. As a non-limiting example, the embodiments can be implemented by one or more integrated circuits. The memory 1020 can be of any type appropriate to the technical environment and can be implemented using any appropriate data storage technology, such as optical memory devices, magnetic memory devices, semiconductor-based memory devices, fixed memory, and removable memory, as non-limiting examples. The processor 1010 can be of any type appropriate to the technical environment, and can encompass one or more of microprocessors, general purpose computers, special purpose computers, and processors based on a multi-core architecture, as non-limiting examples.

Various implementations involve decoding. “Decoding”, as used in this application, can encompass all or part of the processes performed, for example, on a received encoded sequence in order to produce a final output suitable for display. In various embodiments, such processes include one or more of the processes typically performed by a decoder, for example, entropy decoding, inverse quantization, inverse transformation, and differential decoding. In various embodiments, such processes also, or alternatively, include processes performed by a decoder of various implementations described in this application, for example, determining for a current block a neural network intra prediction and side information using a neural network from at least one input data, encoding the current block based on the neural network intra prediction; and encoding the side information.

As further examples, in one embodiment “decoding” refers only to entropy decoding, in another embodiment “decoding” refers only to differential decoding, and in another embodiment “decoding” refers to a combination of entropy decoding and differential decoding. Whether the phrase “decoding process” is intended to refer specifically to a subset of operations or generally to the broader decoding process will be clear based on the context of the specific descriptions and is believed to be well understood by those skilled in the art.

Various implementations involve encoding. In an analogous way to the above discussion about “decoding”, “encoding” as used in this application can encompass all or part of the processes performed, for example, on an input video sequence in order to produce an encoded bitstream. In various embodiments, such processes include one or more of the processes typically performed by an encoder, for example, partitioning, differential encoding,

transformation, quantization, and entropy encoding. In various embodiments, such processes also, or alternatively, include processes performed by an encoder of various implementations described in this application, for example, receiving for a current block an information relative to a side information of a neural network intra prediction; determining a neural network intra prediction for the current block using a neural network applied to at least one input data and side information; and decoding the block using the determined neural network intra prediction.

As further examples, in one embodiment “encoding” refers only to entropy encoding, in another embodiment “encoding” refers only to differential encoding, and in another embodiment “encoding” refers to a combination of differential encoding and entropy encoding. Whether the phrase “encoding process” is intended to refer specifically to a subset of operations or generally to the broader encoding process will be clear based on the context of the specific descriptions and is believed to be well understood by those skilled in the art.

Note that the syntax elements as used herein, for example, *deepFlag*, are descriptive terms. As such, they do not preclude the use of other syntax element names.

When a figure is presented as a flow diagram, it should be understood that it also provides a block diagram of a corresponding apparatus. Similarly, when a figure is presented as a block diagram, it should be understood that it also provides a flow diagram of a corresponding method/process.

Various embodiments refer to rate distortion optimization. In particular, during the encoding process, the balance or trade-off between the rate and distortion is usually considered, often given the constraints of computational complexity. The rate distortion optimization is usually formulated as minimizing a rate distortion function, which is a weighted sum of the rate and of the distortion. There are different approaches to solve the rate distortion optimization problem. For example, the approaches may be based on an extensive testing of all encoding options, including all considered modes or coding parameters values, with a complete evaluation of their coding cost and related distortion of the reconstructed signal after coding and decoding. Faster approaches may also be used, to save encoding complexity, in particular with computation of an approximated distortion based on the prediction or the prediction residual signal, not the reconstructed one. Mix of these two approaches can also be used, such as by using an approximated distortion for only some of the possible encoding options, and a complete distortion for other encoding options. Other approaches only evaluate a subset of the possible encoding options. More generally, many approaches employ any of a variety of techniques to perform the optimization, but the optimization is not necessarily a complete

evaluation of both the coding cost and related distortion. However, according to at least one embodiment, the rate distortion optimization in the encoding is removed as the deep intra prediction mode is the only available mode for intra prediction.

The implementations and aspects described herein can be implemented in, for example, a method or a process, an apparatus, a software program, a data stream, or a signal. Even if only discussed in the context of a single form of implementation (for example, discussed only as a method), the implementation of features discussed can also be implemented in other forms (for example, an apparatus or program). An apparatus can be implemented in, for example, appropriate hardware, software, and firmware. The methods can be implemented in, for example, a processor, which refers to processing devices in general, including, for example, a computer, a microprocessor, an integrated circuit, or a programmable logic device. Processors also include communication devices, such as, for example, computers, cell phones, portable/personal digital assistants ("PDAs"), and other devices that facilitate communication of information between end-users.

Reference to "one embodiment" or "an embodiment" or "one implementation" or "an implementation", as well as other variations thereof, means that a particular feature, structure, characteristic, and so forth described in connection with the embodiment is included in at least one embodiment. Thus, the appearances of the phrase "in one embodiment" or "in an embodiment" or "in one implementation" or "in an implementation", as well as any other variations, appearing in various places throughout this application are not necessarily all referring to the same embodiment.

Additionally, this application may refer to "determining" various pieces of information. Determining the information can include one or more of, for example, estimating the information, calculating the information, predicting the information, or retrieving the information from memory.

Further, this application may refer to "accessing" various pieces of information. Accessing the information can include one or more of, for example, receiving the information, retrieving the information (for example, from memory), storing the information, moving the information, copying the information, calculating the information, determining the information, predicting the information, or estimating the information.

Additionally, this application may refer to "receiving" various pieces of information. Receiving is, as with "accessing", intended to be a broad term. Receiving the information can include one or more of, for example, accessing the information, or retrieving the information (for example, from memory). Further, "receiving" is typically involved, in one way or another, during

operations such as, for example, storing the information, processing the information, transmitting the information, moving the information, copying the information, erasing the information, calculating the information, determining the information, predicting the information, or estimating the information.

- 5 It is to be appreciated that the use of any of the following “/”, “and/or”, and “at least one of”, for example, in the cases of “A/B”, “A and/or B” and “at least one of A and B”, is intended to encompass the selection of the first listed option (A) only, or the selection of the second listed option (B) only, or the selection of both options (A and B). As a further example, in the cases of “A, B, and/or C” and “at least one of A, B, and C”, such phrasing is intended to encompass
- 10 the selection of the first listed option (A) only, or the selection of the second listed option (B) only, or the selection of the third listed option (C) only, or the selection of the first and the second listed options (A and B) only, or the selection of the first and third listed options (A and C) only, or the selection of the second and third listed options (B and C) only, or the selection of all three options (A and B and C). This may be extended, as is clear to one of ordinary skill
- 15 in this and related arts, for as many items as are listed.

- Also, as used herein, the word “signal” refers to, among other things, indicating something to a corresponding decoder. For example, in certain embodiments the encoder signals a particular one of a plurality of parameters for deep intra prediction. In this way, in an embodiment the same parameter is used at both the encoder side and the decoder side. Thus,
- 20 for example, an encoder can transmit (explicit signaling) a particular parameter to the decoder so that the decoder can use the same particular parameter. Conversely, if the decoder already has the particular parameter as well as others, then signaling can be used without transmitting (implicit signaling) to simply allow the decoder to know and select the particular parameter. By avoiding transmission of any actual functions, a bit savings is realized in various embodiments.
- 25 It is to be appreciated that signaling can be accomplished in a variety of ways. For example, one or more syntax elements, flags, and so forth are used to signal information to a corresponding decoder in various embodiments. While the preceding relates to the verb form of the word “signal”, the word “signal” can also be used herein as a noun.

- As will be evident to one of ordinary skill in the art, implementations can produce a variety of
- 30 signals formatted to carry information that can be, for example, stored or transmitted. The information can include, for example, instructions for performing a method, or data produced by one of the described implementations. For example, a signal can be formatted to carry the bitstream of a described embodiment. Such a signal can be formatted, for example, as an electromagnetic wave (for example, using a radio frequency portion of spectrum) or as a
- 35 baseband signal. The formatting can include, for example, encoding a data stream and

modulating a carrier with the encoded data stream. The information that the signal carries can be, for example, analog or digital information. The signal can be transmitted over a variety of different wired or wireless links, as is known. The signal can be stored on a processor-readable medium.

5 We describe a number of embodiments. Features of these embodiments can be provided alone or in any combination, across various claim categories and types. Further, embodiments can include one or more of the following features, devices, or aspects, alone or in any combination, across various claim categories and types:

- Modifying the intra prediction process applied in the decoder and/or encoder.
- 10 • Using a neural network in the intra prediction process applied in the decoder and/or encoder.
- Determining a neural network intra prediction and side information using a neural network from at least one input data in the encoder;
- Determining a neural network intra prediction using a neural network applied to at least  
15 one input data and received side information in the decoder;
- Encoding side information generated by the intra deep predictor in the encoder;
- Decoding side information to be used by the intra deep predictor in the decoder;
- For luma block, the input data of the neural network intra predictor in the encoder comprises a context surrounding the luma block and luma block;
- 20 • For luma block, the input data of the neural network intra predictor in the decoder comprises a context surrounding the luma block and decoded side information;
- For chroma block, the input data of the neural network intra predictor in the encoder comprises a context surrounding the chroma block, the context surrounding the luma block that is collocated with the chroma block, and the chroma block.
- 25 • For chroma block, the input data of the neural network intra predictor in the decoder comprises a context surrounding the chroma block, the context surrounding the luma block that is collocated with the chroma block, and the generated side information.
- For chroma block, the input data of the neural network intra predictor in the encoder comprises the context surrounding the luma block that is collocated with the chroma  
30 block, and the chroma block.
- For chroma block, the input data of the neural network intra predictor in the decoder comprises the context surrounding the luma block that is collocated with the chroma block, and the generated side information.
- For chroma block, the input data of the neural network intra predictor in the encoder  
35 comprises a context surrounding the chroma block, the context surrounding the luma

block that is collocated with the chroma block, the chroma block and the reconstructed luma block.

- For chroma block, the input data of the neural network intra predictor in the decoder comprises a context surrounding the chroma block, the context surrounding the luma block that is collocated with the chroma block, the reconstructed luma block, and the generated side information.

- Any of the variant of input data for luma and chroma block further comprising as input data the intra prediction mode L of the block located on the left side of the block and the intra prediction mode A of the block located above the block.

- Inserting in the signaling syntax elements that enable the decoder to identify the intra prediction method to use.
- Selecting, based on these syntax elements, the intra prediction method to apply at the decoder.

- Inserting in the signaling a syntax element DeepFlag representative of the neural network-based intra prediction mode among syntax elements of intra prediction modes.

- Enabling, from the syntax element, only one of neural network-based intra prediction mode and planar intra prediction mode for luma intra prediction.

- Enabling only neural network-based intra prediction mode.

- A bitstream or signal that includes one or more of the described syntax elements, or variations thereof.

- A bitstream or signal that includes syntax conveying information generated according to any of the embodiments described.

- Inserting in the signaling syntax elements that enable the decoder to determine deep intra prediction used in the encoder.

- Creating and/or transmitting and/or receiving and/or decoding a bitstream or signal that includes one or more of the described syntax elements, or variations thereof.

- Creating and/or transmitting and/or receiving and/or decoding according to any of the embodiments described.

- A method, process, apparatus, medium storing instructions, medium storing data, or signal according to any of the embodiments described.

- A TV, set-top box, cell phone, tablet, or other electronic device that performs intra prediction according to any of the embodiments described.

- A TV, set-top box, cell phone, tablet, or other electronic device that performs intra prediction according to any of the embodiments described, and that displays (e.g. using a monitor, screen, or other type of display) a resulting image.

- A TV, set-top box, cell phone, tablet, or other electronic device that selects (e.g. using a tuner) a channel to receive a signal including an encoded image, and performs intra prediction according to any of the embodiments described.
- A TV, set-top box, cell phone, tablet, or other electronic device that receives (e.g. using an antenna) a signal over the air that includes an encoded image, and performs intra prediction according to any of the embodiments described.

CLAIMS

1. A method for video decoding, comprising:
  - obtaining, for a block being decoded in a picture of a video, a side information for a neural network intra prediction;
  - determining, for said block being decoded, a neural network intra prediction using a neural network applied to at least one input data and said side information;
  - decoding the block using said determined neural network intra prediction.
2. An apparatus for video decoding, comprising one or more processors, wherein the one or more processors are configured to:
  - obtain, for a block being decoded in a picture of a video, a side information for a neural network intra prediction;
  - determine, for said block being decoded, a neural network intra prediction using a neural network applied to at least one input data and said side information;
  - decoding the block using said determined neural network intra prediction.
3. A method for video encoding, comprising:
  - determining, for a block being encoded in a picture of a video, a neural network intra prediction and side information using a neural network from at least one input data;
  - encoding the block based on neural network intra prediction; and
  - encoding said side information.
4. An apparatus for video encoding, comprising one or more processors, wherein the one or more processors are configured to:
  - determine, for a block being encoded in a picture of a video, a neural network intra prediction and side information using a neural network from at least one input data;
  - encode the block based on a neural network intra prediction; and
  - encode said side information.
5. The method of claim 1 or 3 or the apparatus of claim 2 or 4, wherein the neural network is fully-connected and said side information is a vector ( $Z$ ) whose coefficients belong to  $[0,1]$ .
6. The method of claim 5 or the apparatus of claim 5, wherein the vector is converted into a vector of bits by applying an elementwise threshold of 0.5 to the vector.

7. The method of any one of claims 1, or 3 or the apparatus of claim 2 or 4, wherein the neural network is convolutional and wherein said side information is a stack of feature maps ( $Z$ ) whose coefficients belong to  $[0, 1]$ .

5 8. The method of claim 7 or the apparatus of claim 7, wherein the stack of feature maps is converted into a stack of feature maps of bits by applying an elementwise threshold of 0.5 to the coefficients of the stack of feature maps.

10 9. The method of any one of claims 1, 3 or the apparatus of any one of claims 2, 4, wherein side information ( $Z$ ) is a vector or a a stack of feature maps whose coefficients belong to  $\mathbb{R}$  and wherein the coefficients of side information are quantized via either scalar quantization or vector quantization, and wherein the absolute value of the quantized coefficients is losslessly encoded via arithmetic coding.

15 10. The method of any one of claims 1, 3 or the apparatus of any one of claims 2, 4, wherein side information is a vector or a stack of feature maps whose coefficients belong to  $[-1, 1]$ , and wherein the coefficients of side information are mapped to  $\{0, 1\}$ .

20 11. The method of any one of claims 3 and 5-10 or the apparatus of any one of claims 4 and 5-10, wherein for luma block ( $Y$ ), the input data of the encoding comprises a context ( $X_c$ ) surrounding the luma block ( $Y$ ) and luma block ( $Y$ ).

25 12. The method of any one of claims 1 and 5-11 or the apparatus of any one of claims 2 and 5-11, wherein for a luma block ( $Y$ ), the input data of the decoding comprises a context ( $X_c$ ) surrounding the luma block ( $Y$ ).

30 13. The method of any one of claims 3 and 5-12 or the apparatus of any one of claims 4 and 5-12, wherein for a chroma block ( $Y^{CbCr}$ ), the input data of the encoding comprises a context ( $X_c^{CbCr}$ ) surrounding the chroma block ( $Y^{CbCr}$ ), the context ( $X_c^Y$ ) surrounding the luma block ( $Y^Y$ ) that is collocated with the chroma block, and the chroma block ( $Y^{CbCr}$ ).

35 14. The method of any one of claims 3 and 5-12 or the apparatus of any one of claims 4 and 5-12, wherein for a chroma block ( $Y^{CbCr}$ ), the input data of the encoding comprises the context ( $X_c^Y$ ) surrounding the luma block ( $Y^Y$ ) that is collocated with the chroma block, and the chroma block ( $Y^{CbCr}$ ).

15. The method of any one of claims 3 and 5-12 or the apparatus of any one of claims 4 and 5-12, wherein for a chroma block ( $Y^{CbCr}$ ), the input data of the encoding comprises a context ( $X_c^{CbCr}$ ) surrounding the chroma block ( $Y^{CbCr}$ ), the context ( $X_c^Y$ ) surrounding the luma block ( $Y^Y$ ) that is collocated with the chroma block, the chroma block ( $Y^{CbCr}$ ), and the reconstructed luma block ( $\tilde{Y}^Y$ ).

16. The method of any one of claims 1 and 5-12 or the apparatus of any one of claims 3 and 5-12, wherein for a chroma block ( $Y^{CbCr}$ ), the input data of the decoding comprises a context ( $X_c^{CbCr}$ ) surrounding the chroma block ( $Y^{CbCr}$ ), the context ( $X_c^Y$ ) surrounding the luma block ( $Y^Y$ ) that is collocated with the chroma block.

17. The method of any one of claims 1 and 5-12 or the apparatus of any one of claims 3 and 5-12, wherein for a chroma block ( $Y^{CbCr}$ ), the input data of the decoding comprises the context ( $X_c^Y$ ) surrounding the luma block ( $Y^Y$ ) that is collocated with the chroma block.

18. The method of any one of claims 1 and 5-12 or the apparatus of any one of claims 3 and 5-12, wherein for a chroma block ( $Y^{CbCr}$ ), the input data of the decoding comprises a context ( $X_c^{CbCr}$ ) surrounding the chroma block ( $Y^{CbCr}$ ), the context ( $X_c^Y$ ) surrounding the luma block ( $Y^Y$ ) that is collocated with the chroma block, and the reconstructed luma block ( $\tilde{Y}^Y$ ).

19. The method of any one of claims 1, 3 or 5-18 or the apparatus of any one of claims 2, 4-18, wherein the input data further comprises the intra prediction mode ( $L$ ) of the block located on the left side of the block being encoded or decoded and the intra prediction mode ( $A$ ) of the block located above the block being encoded or decoded.

20. The method of any one of claims 1, 3 or 5-19 or the apparatus of any one of claims 2, 4-19, wherein for intra prediction of a luma block, a syntax element DeepFlag representative of a neural network-based intra prediction mode is encoded among syntax elements of intra prediction.

21. The method of any one of claims 1, 3 or 5-19 or the apparatus of any one of claims 3, 4-19, wherein for intra prediction of a luma block, a neural network-based intra prediction mode is always selected.

22. The method of any one of claims 1, 3 or 5-19 or the apparatus of any one of claims 2, 4-19, wherein for intra prediction of a luma block, a syntax element DeepFlag representative

of a neural network-based intra prediction is encoded and if DeepFlag is set to one, the luma block is intra predicted using a neural network-based intra prediction else if DeepFlag is set to zero the luma block is intra predicted using planar mode.

5 23. The method of any one of claims 1, 3 or 5-19 or the apparatus of any one of claims 2, 4-19, a neural network-based intra prediction mode is always selected.

24. The method of any one of claims 1, 3 or 5-15 or the apparatus of any one of claims 2, 4-15, wherein for intra prediction of a chroma block, a syntax element DeepFlag representative  
10 of a neural network-based intra prediction mode is encoded among syntax elements of intra prediction.

25. A non-transitory computer readable medium containing data content generated according to the method of any one of claims 3 and 5-24, or the apparatus of any one of claims  
15 4 and 5-24.

26. A computer program product comprising instructions for performing the method of any of any one of claims 1, 2 and 5-24 when executed by one of more processors.

20 27. A signal comprising video data generated according to the method of any one of claims 3 and 5-24, or the apparatus of any one of claims 4 and 5-24.

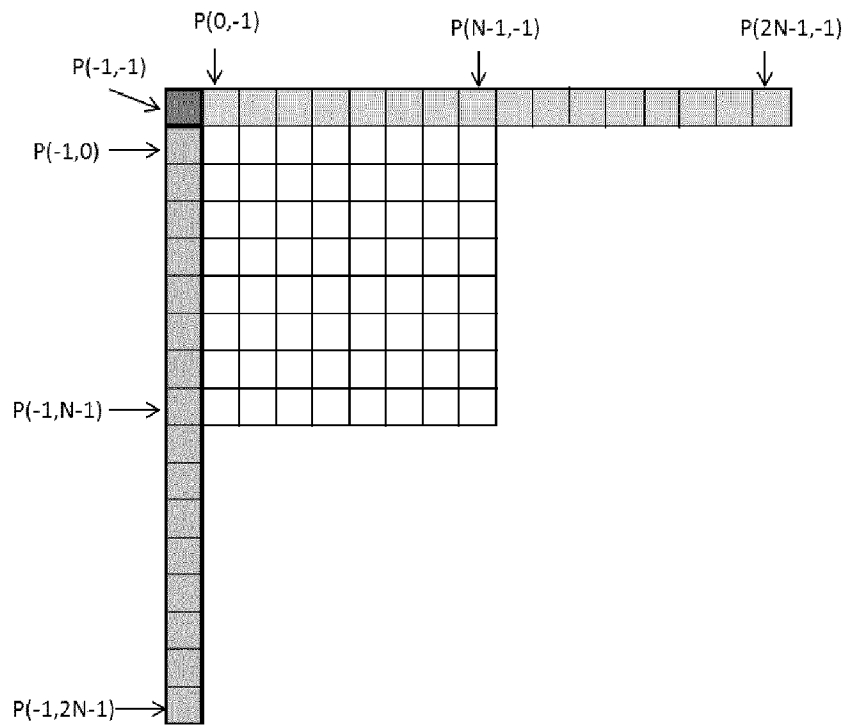


Figure 1

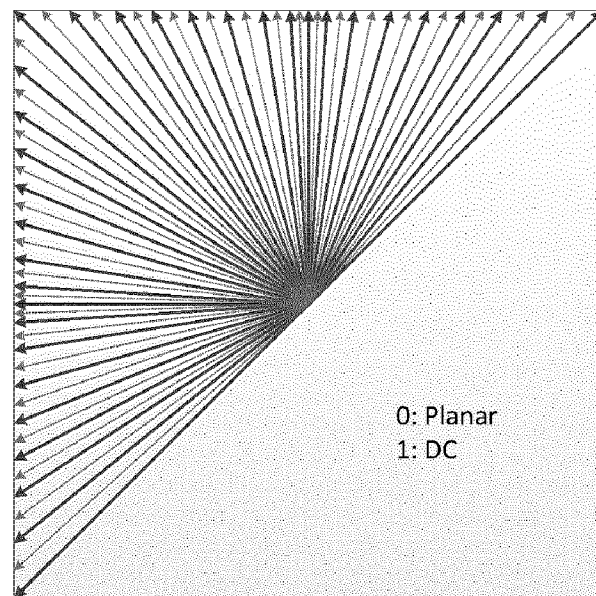


Figure 2

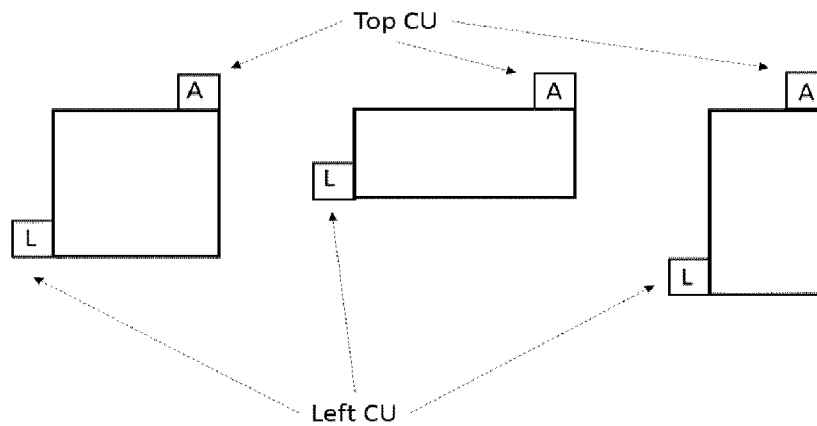


Figure 3

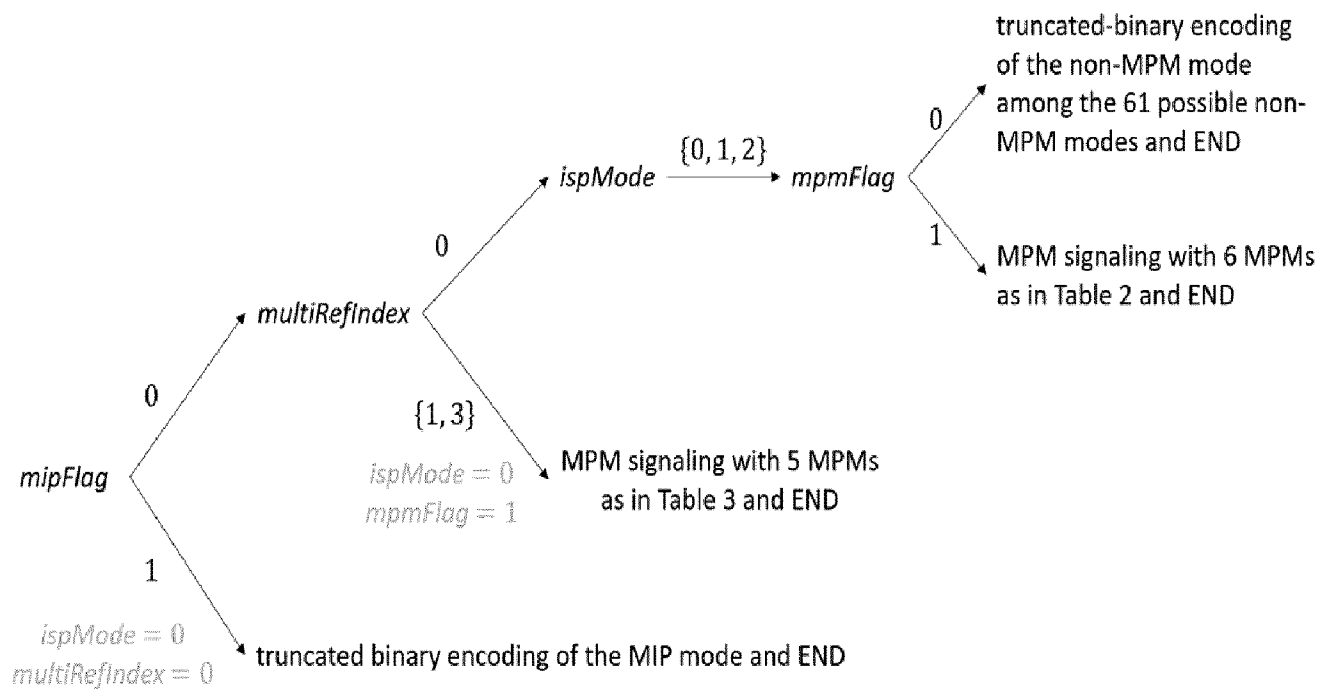


Figure 4

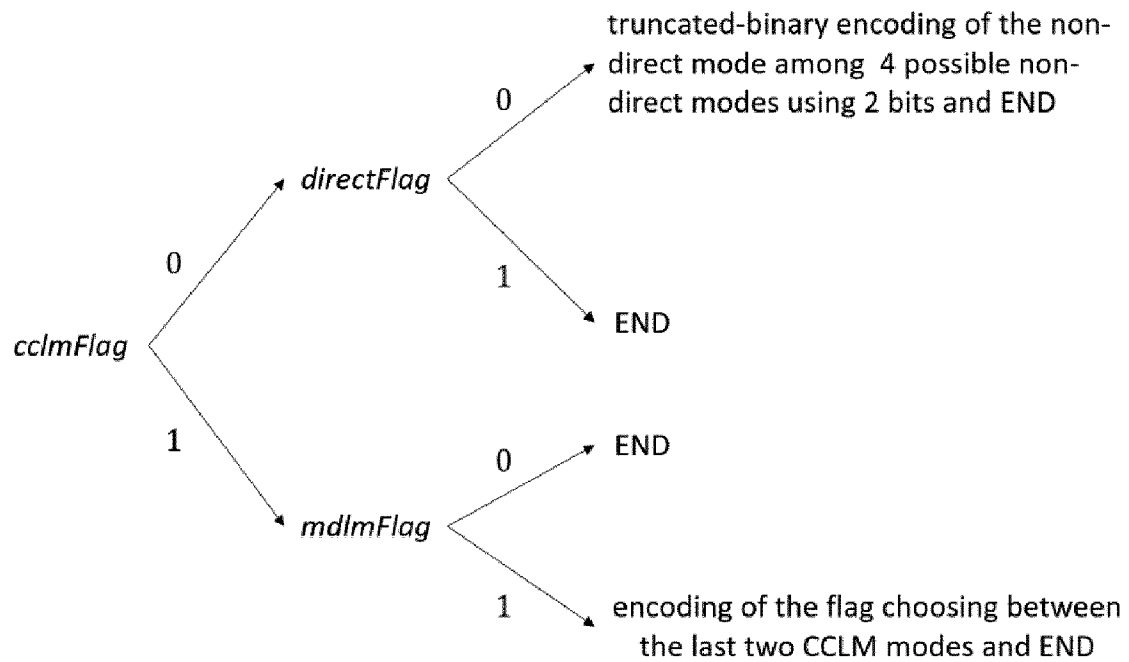


Figure 5

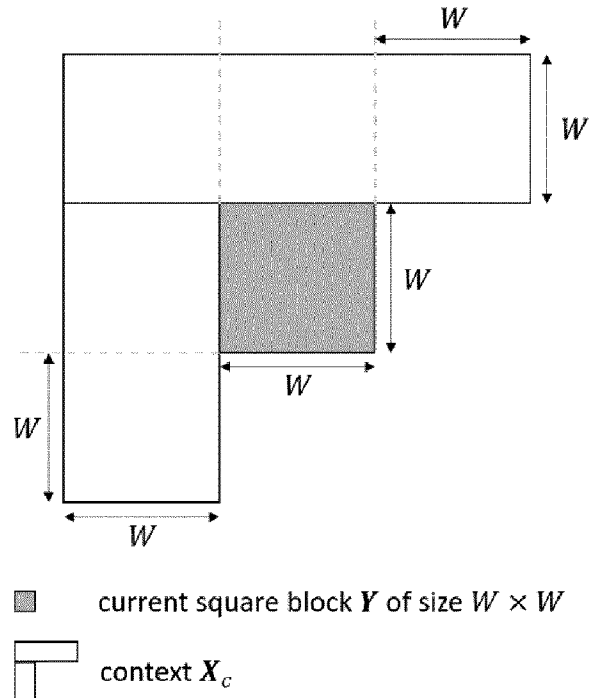


Figure 6

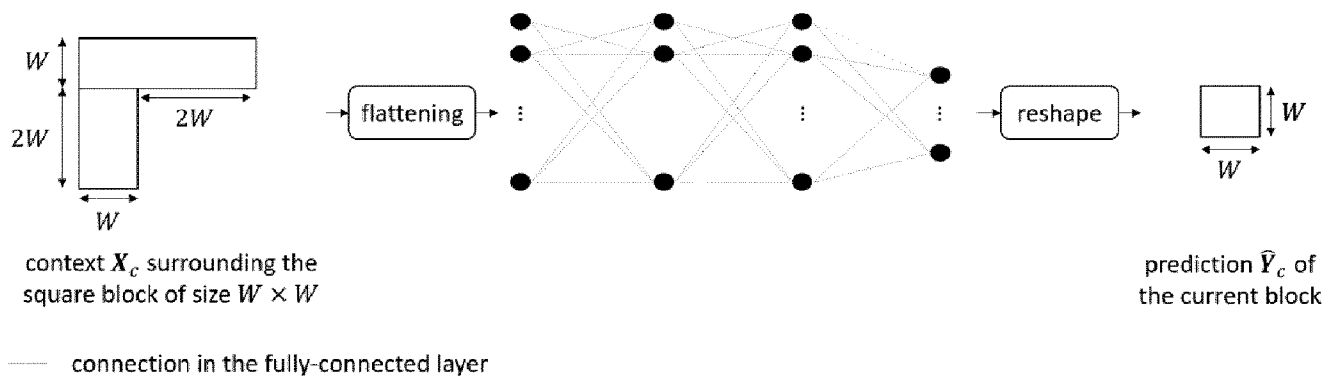


Figure 6b

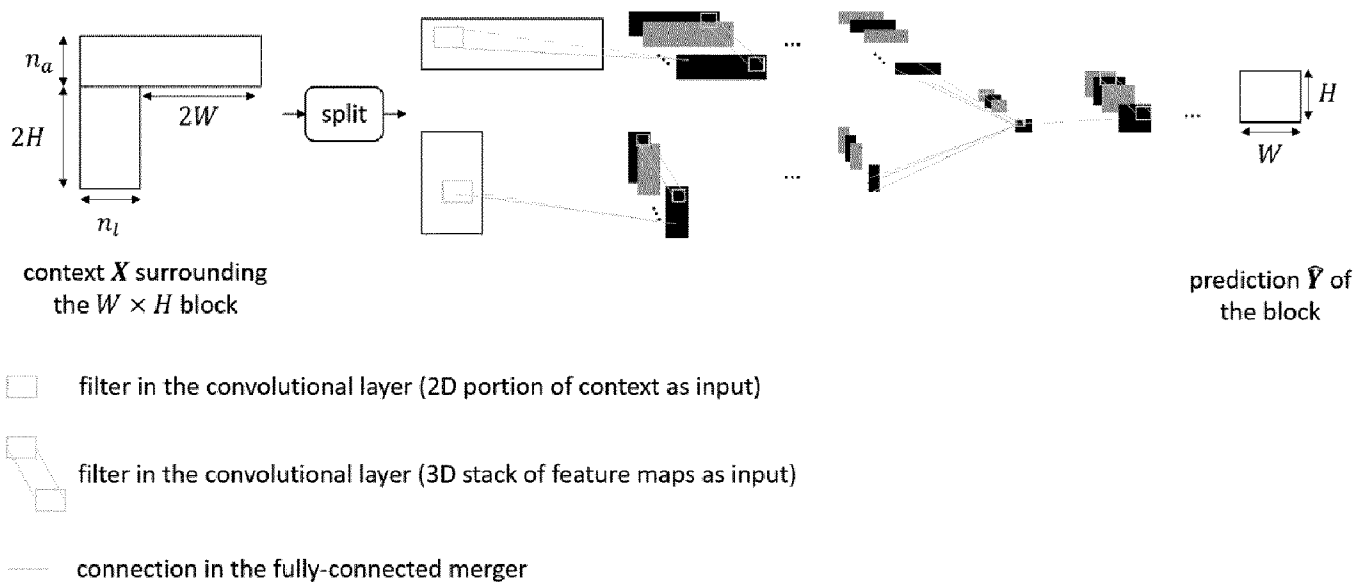


Figure 6c

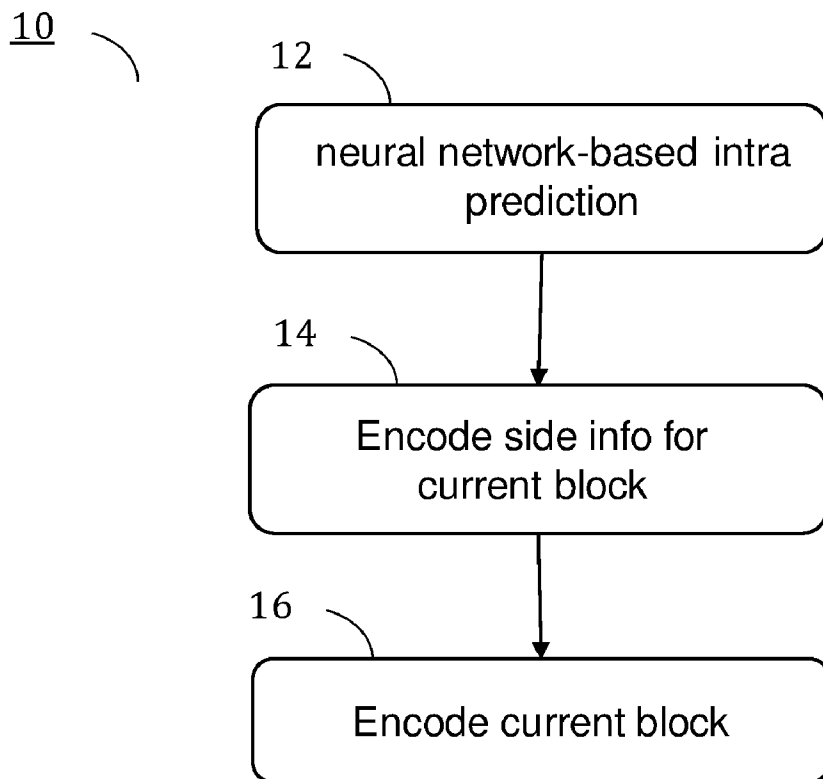


Figure 7

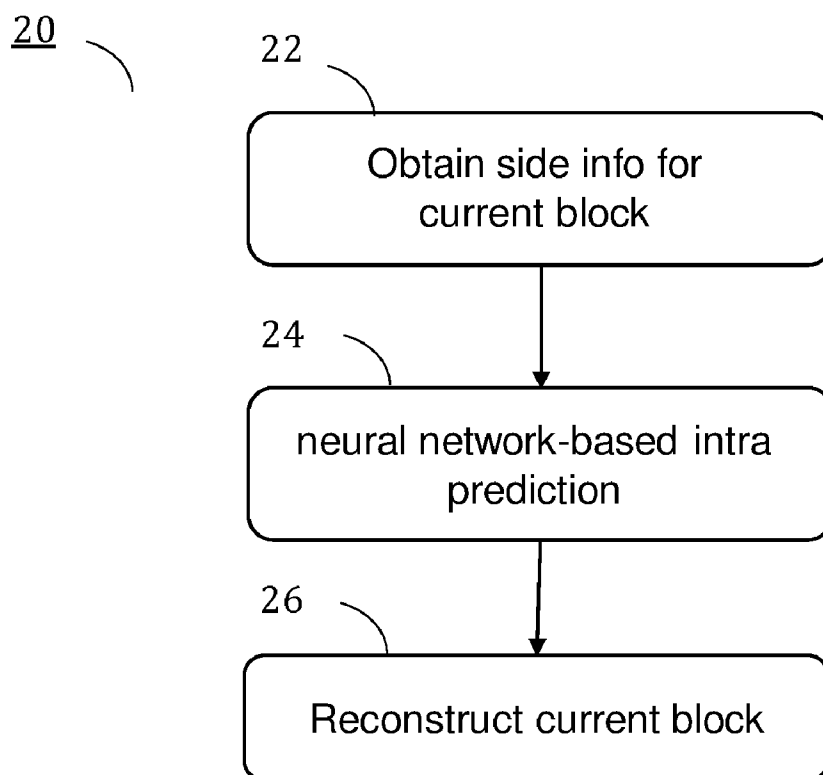


Figure 8

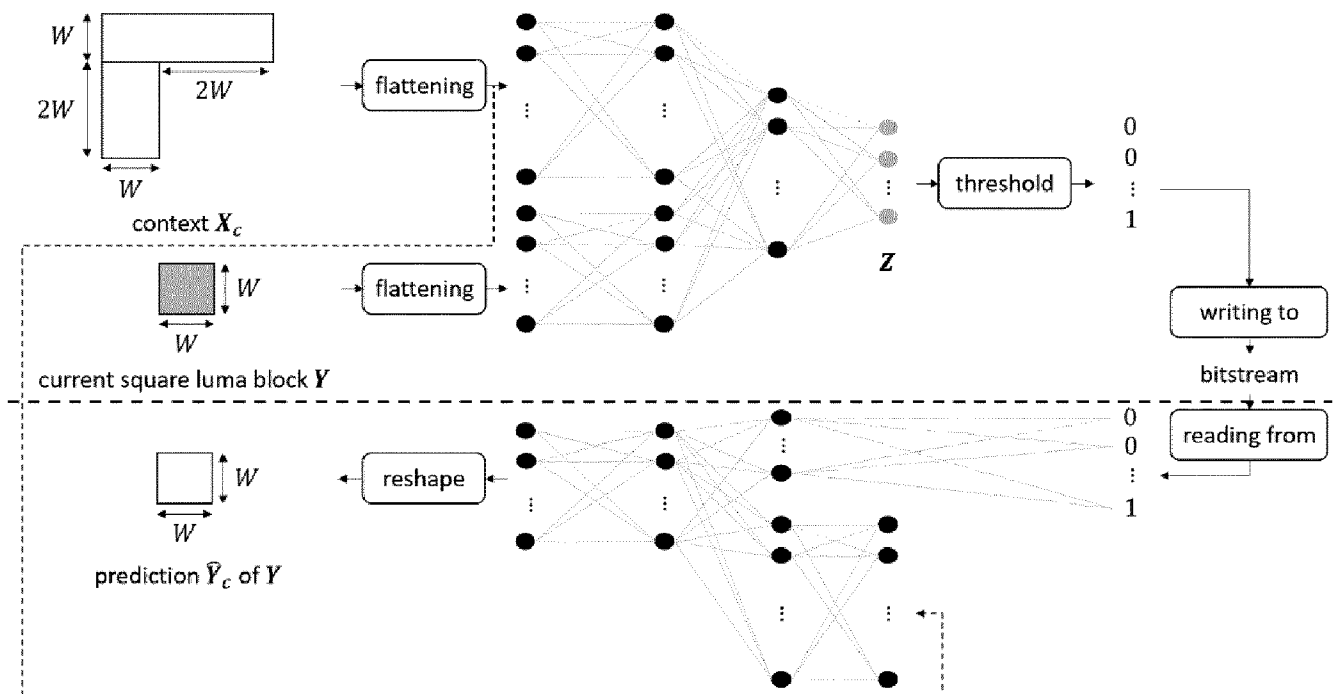


Figure 9

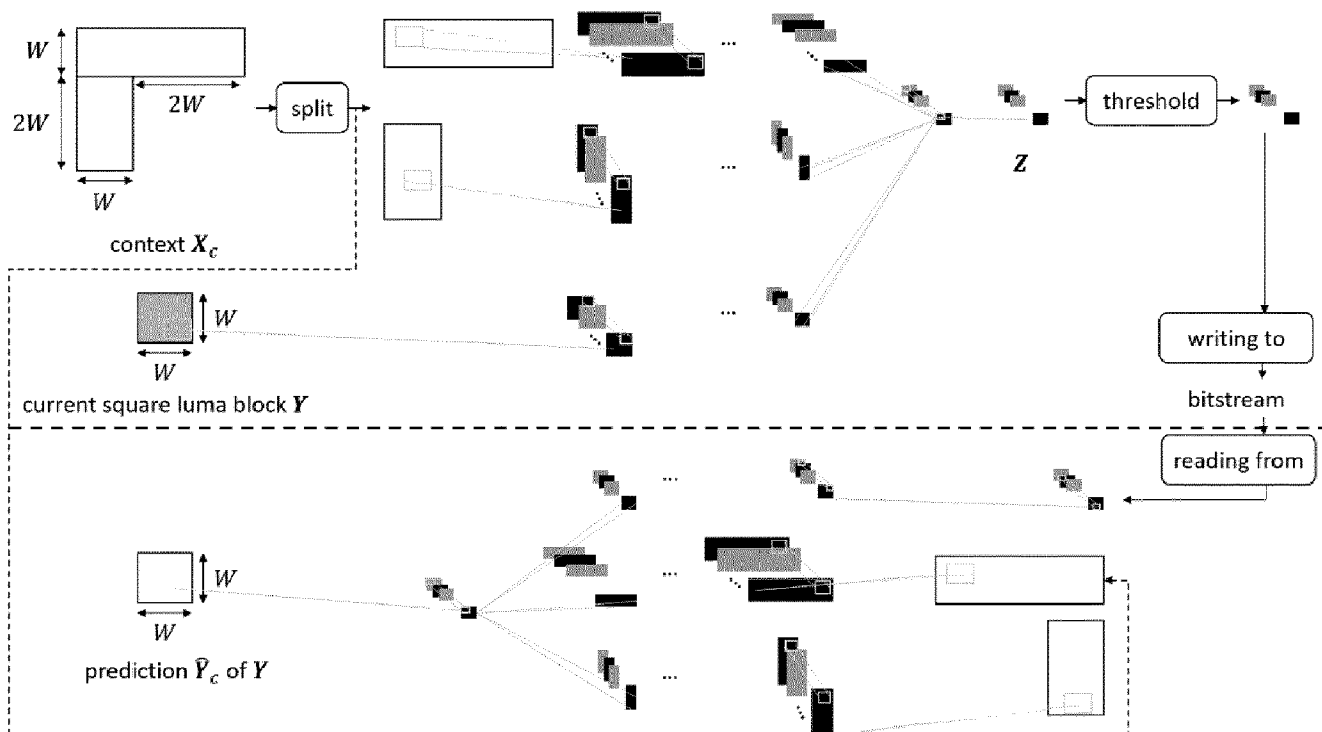


Figure 10

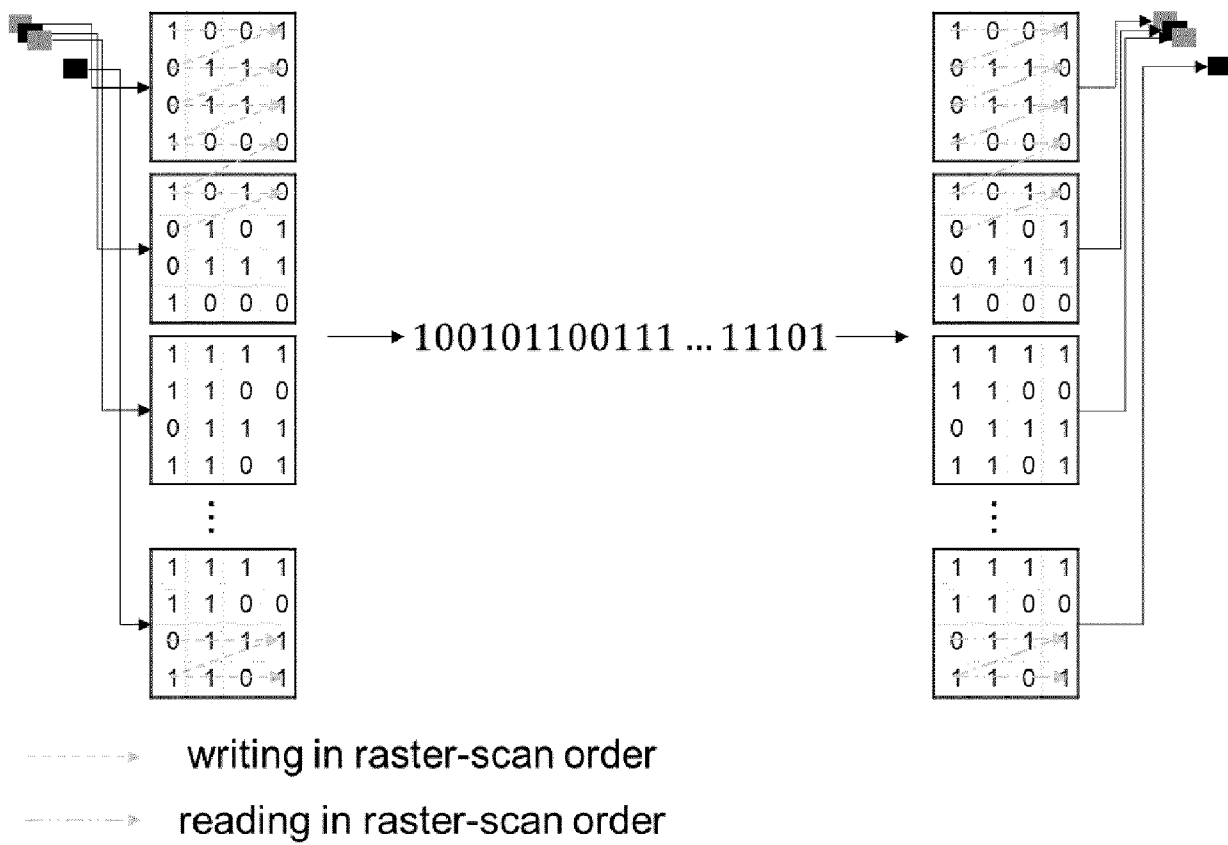


Figure 11

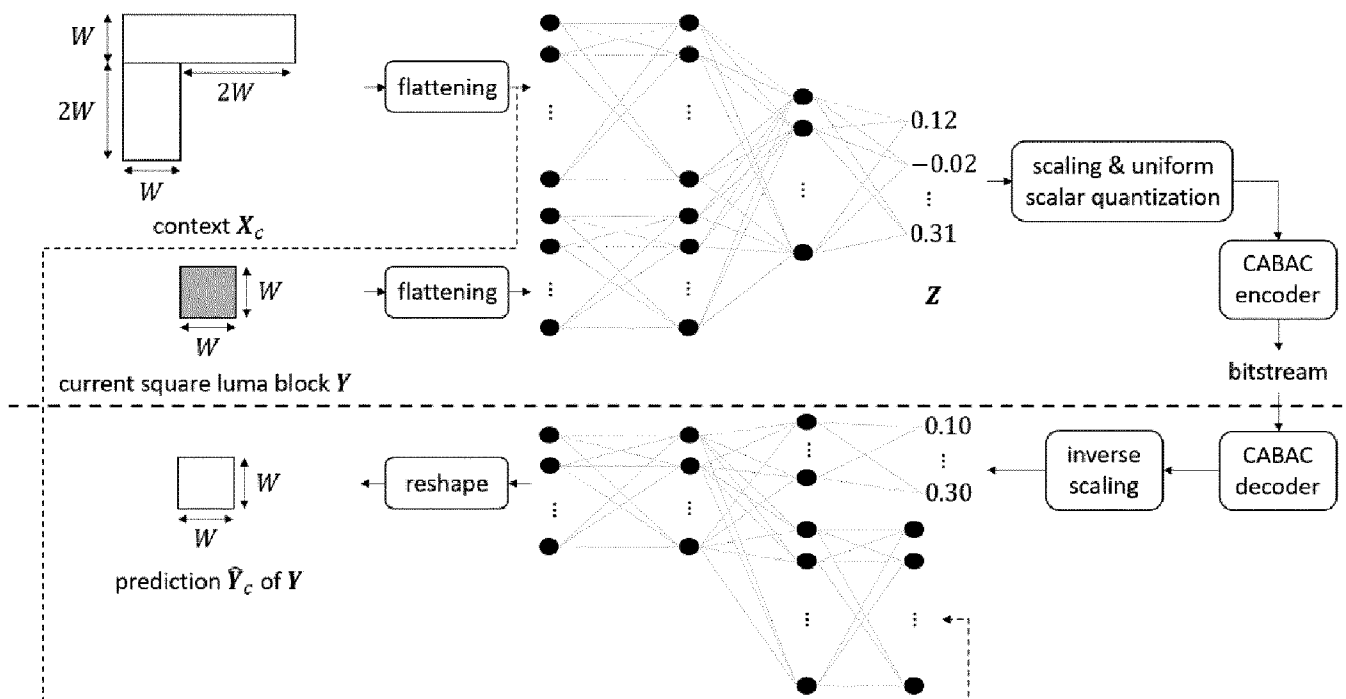


Figure 12

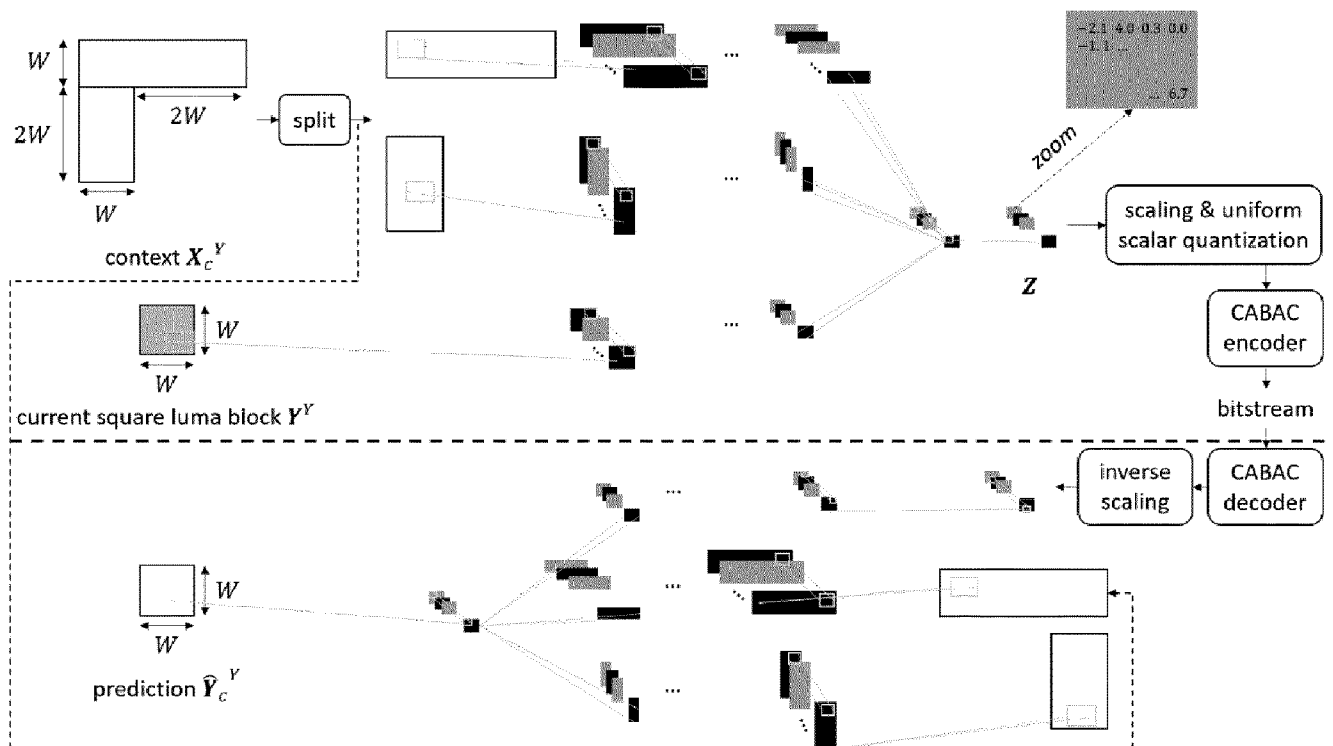


Figure 13

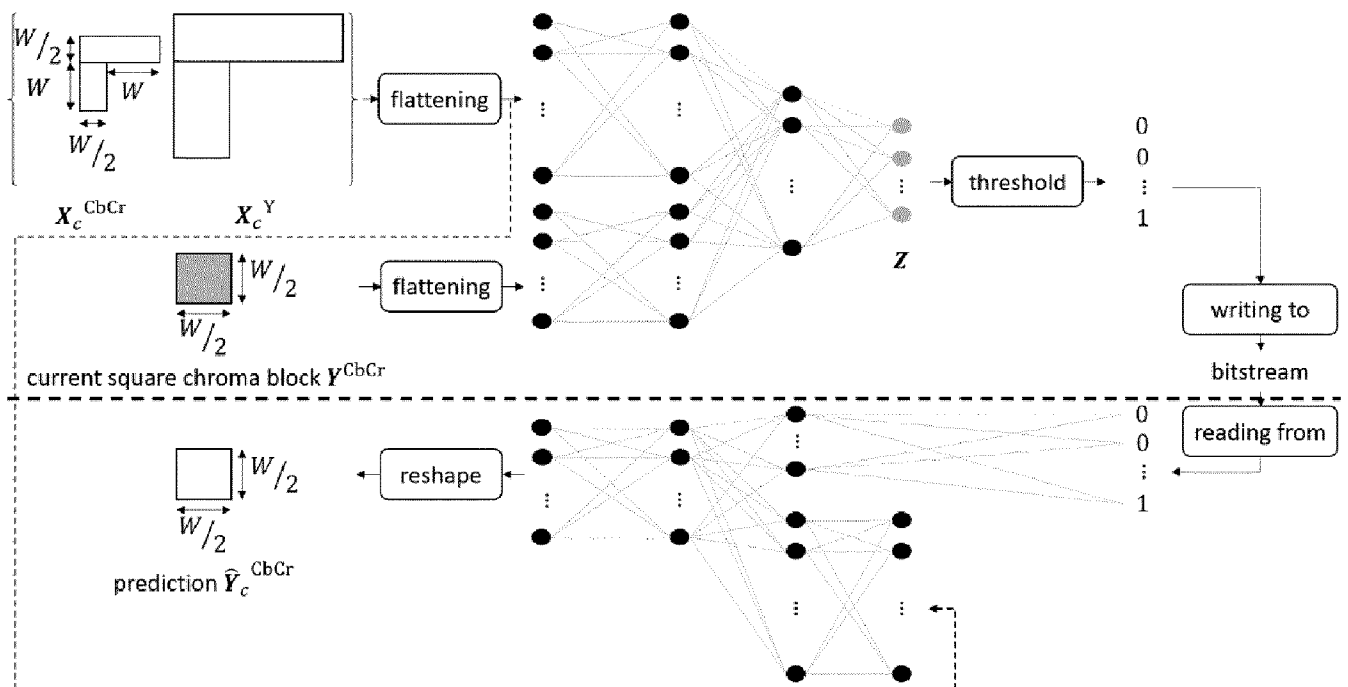


Figure 14

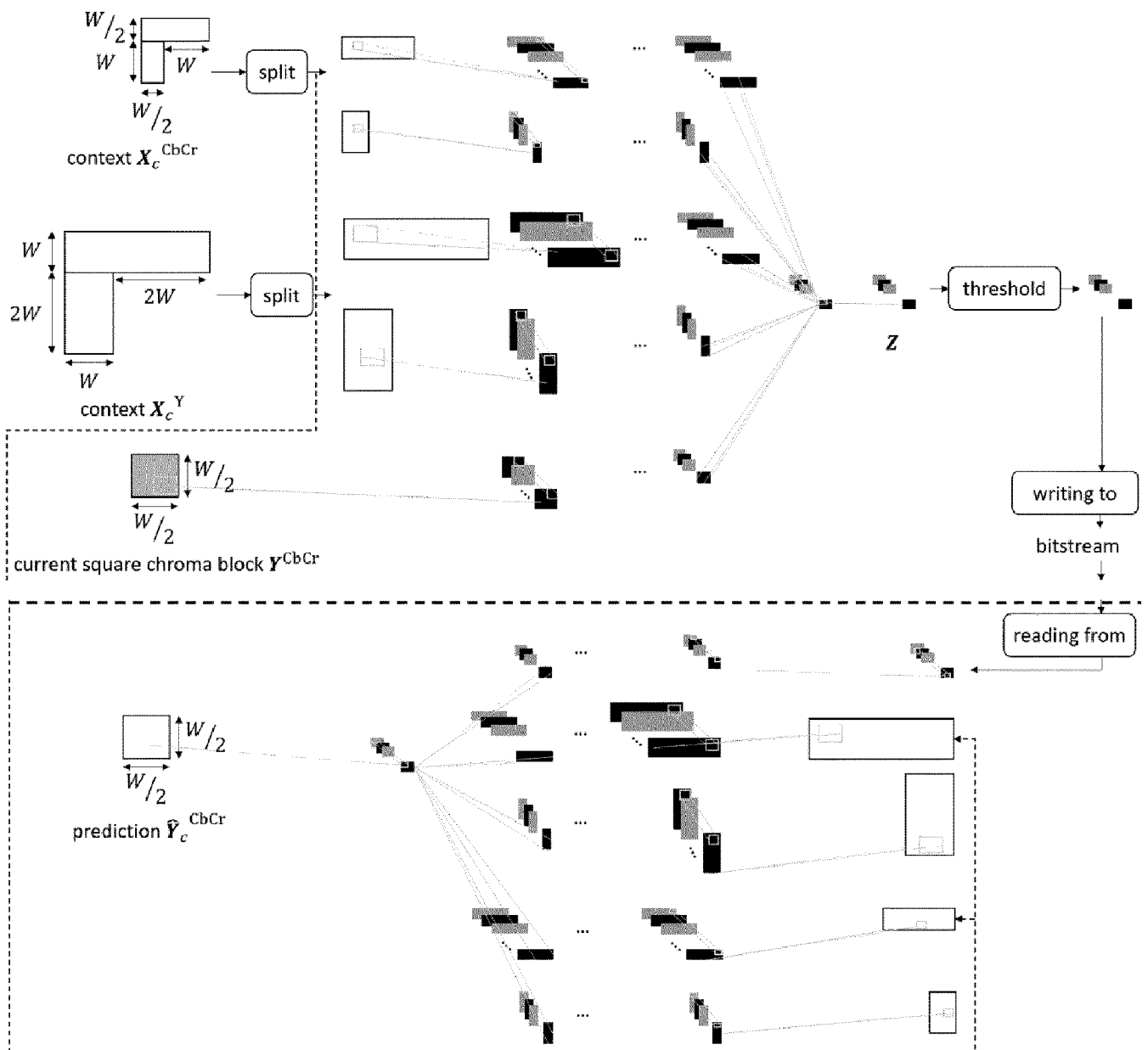


Figure 15

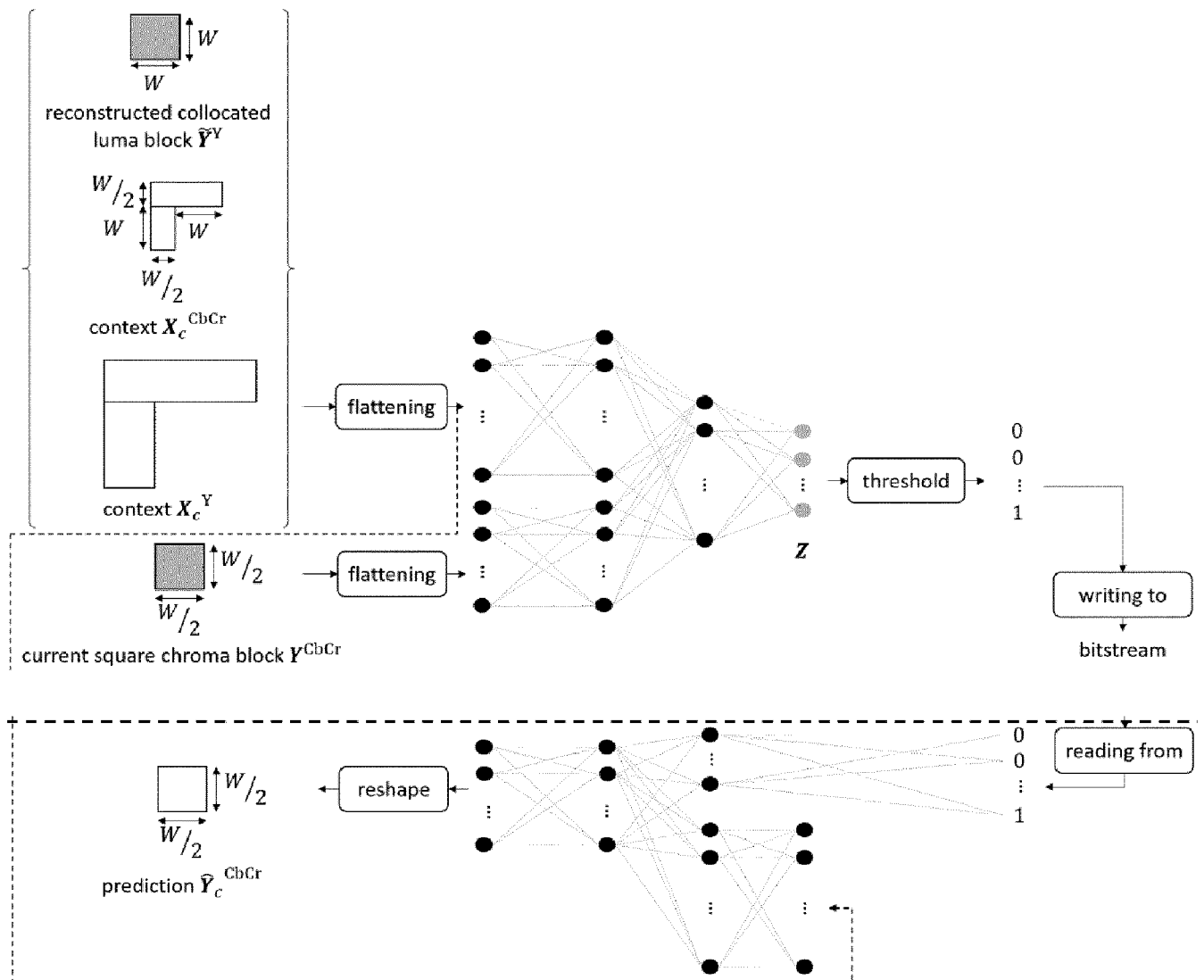


Figure 16

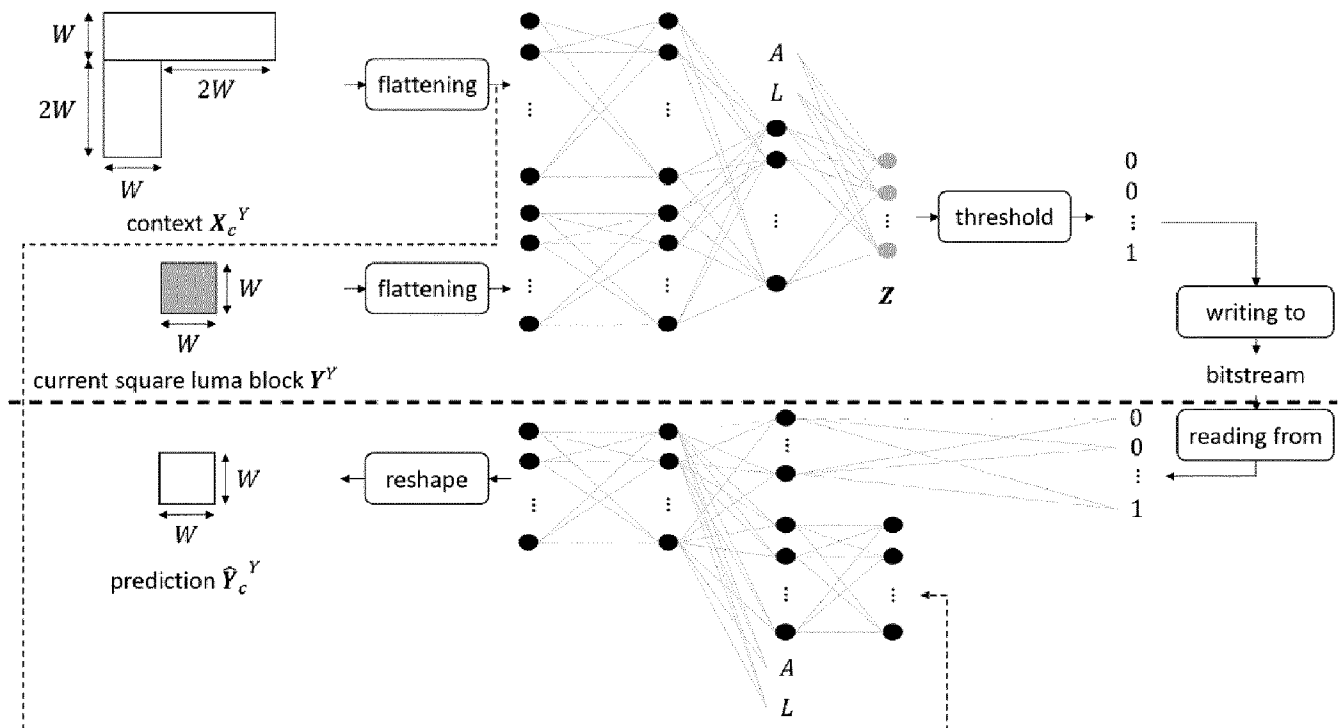


Figure 17

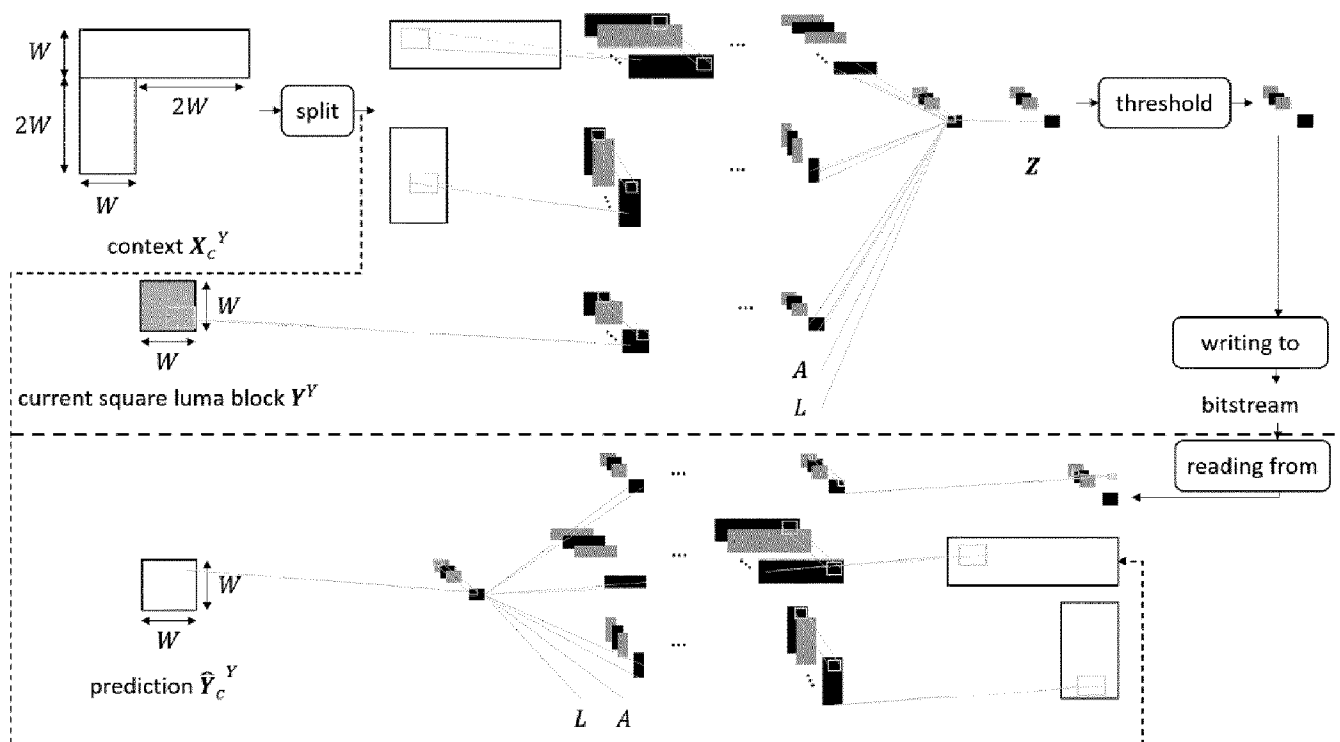


Figure 18

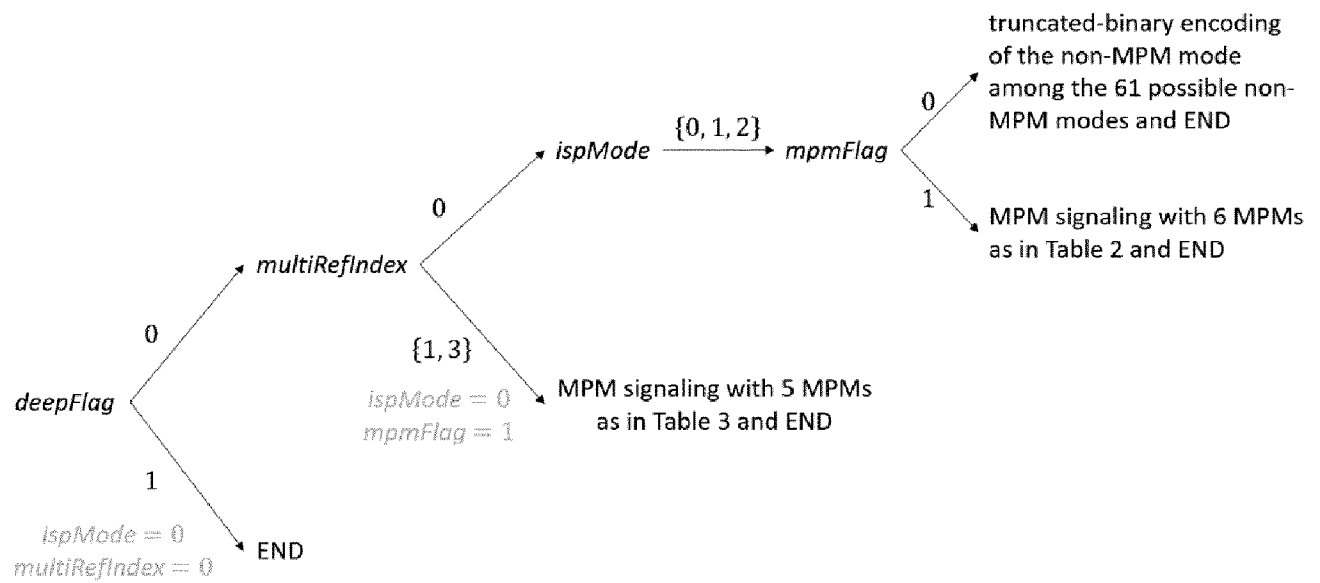


Figure 19

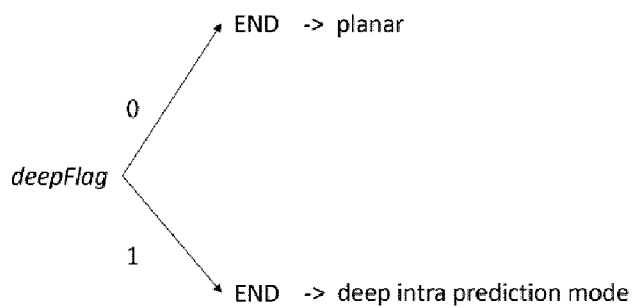


Figure 20

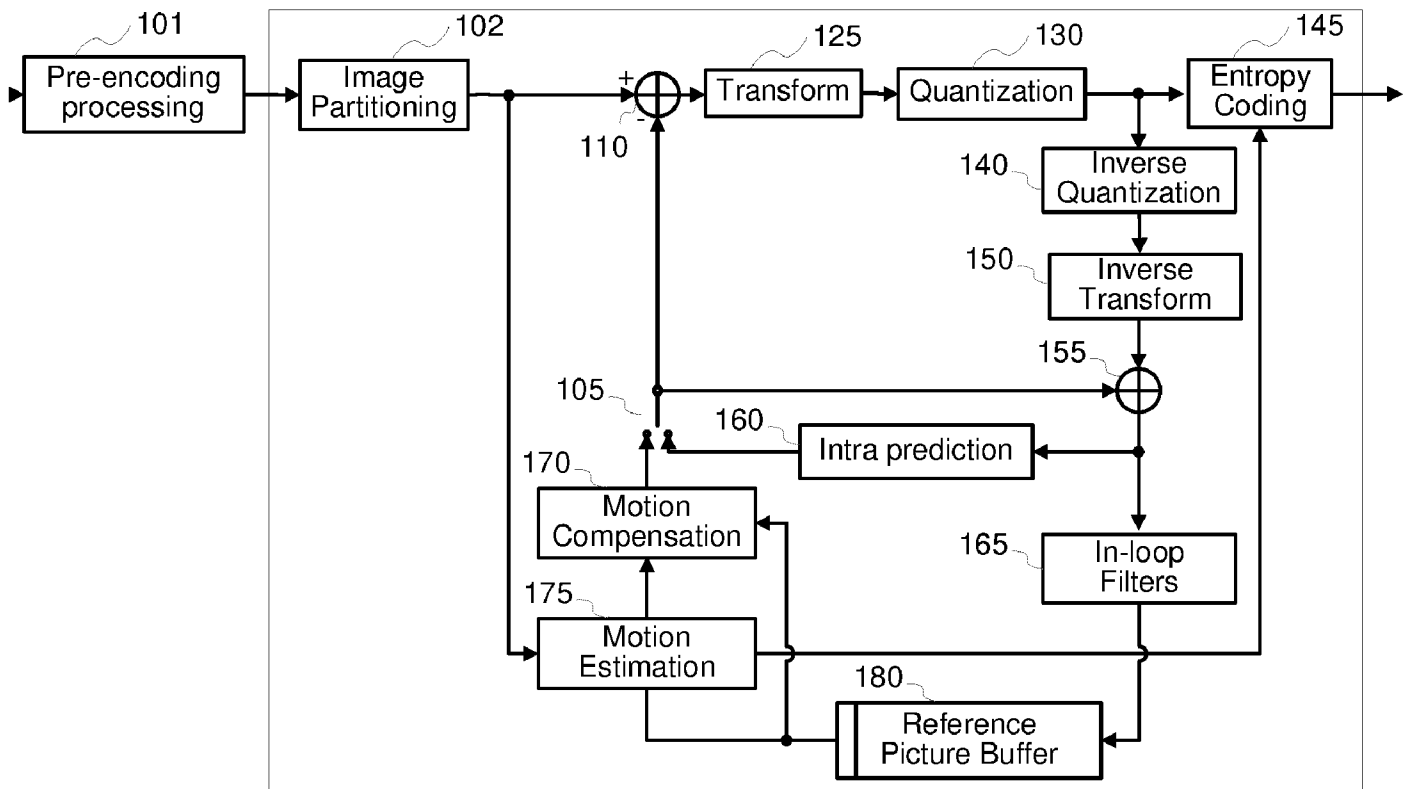
**100**

Figure 21

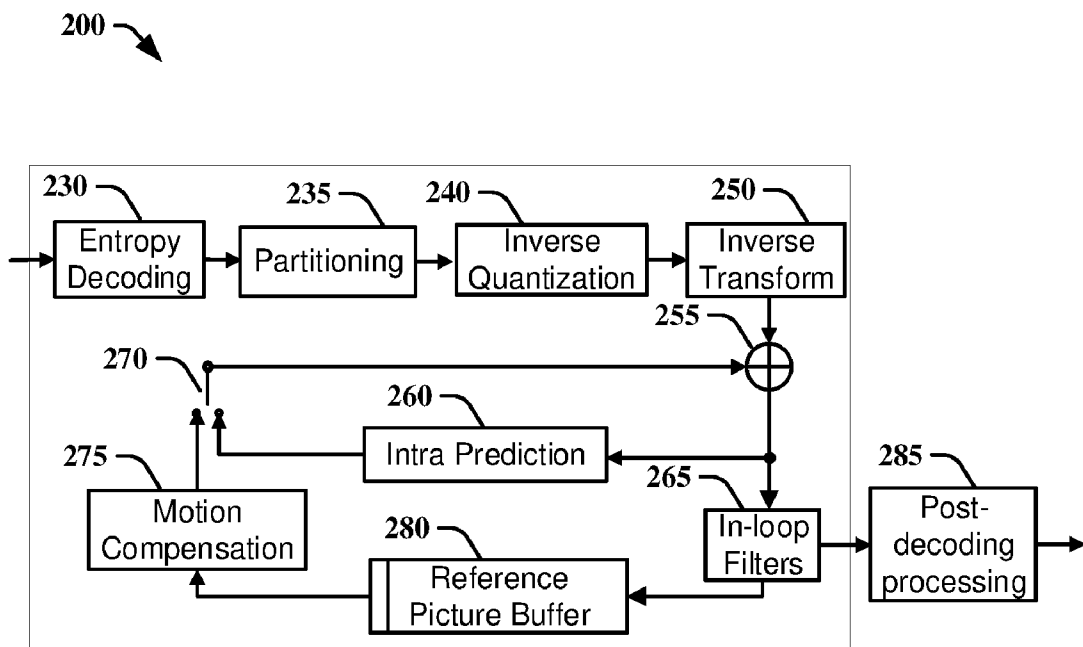


Figure 22

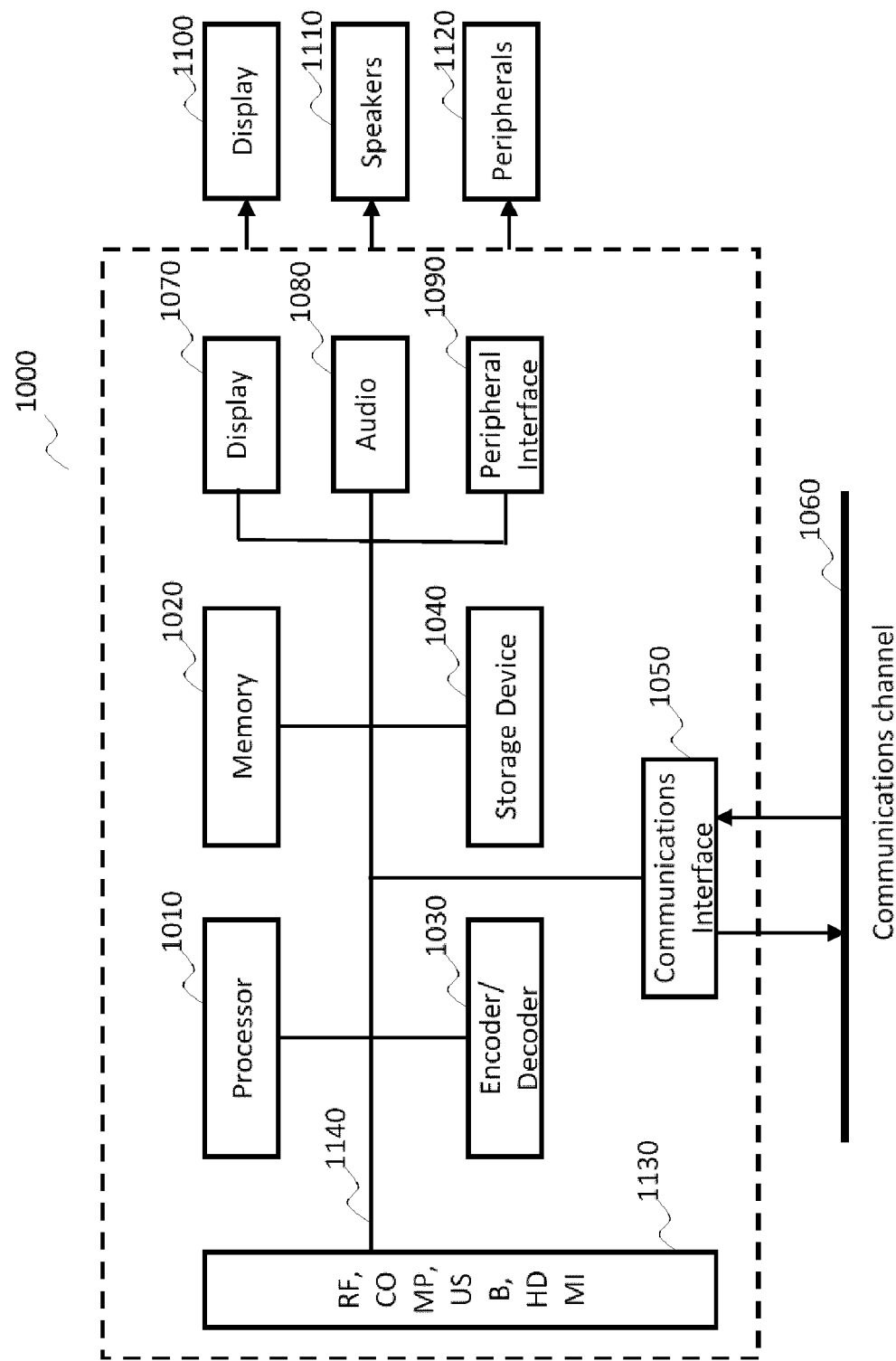


Figure 23

## INTERNATIONAL SEARCH REPORT

International application No  
PCT/EP2020/078460

A. CLASSIFICATION OF SUBJECT MATTER  
INV. H04N19/11 H04N19/593 H04N19/46  
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, COMPENDEX, INSPEC, 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         | PFAFF (FRAUNHOFER) J ET AL: "Intra Prediction Modes based on Neural Networks",<br>10. JMET MEETING; 20180410 - 20180420; SAN DIEGO; (THE JOINT VIDEO EXPLORATION TEAM OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16),<br>,<br>no. JMET-J0037<br>3 April 2018 (2018-04-03), XP030151210,<br>Retrieved from the Internet:<br>URL:http://phenix.int-evry.fr/jvet/doc_end_user/documents/10_San%20Diego/wg11/JVET-J0037-v1.zip JVET-J0037.docx<br>[retrieved on 2018-04-03] | 1-4,12,<br>20,22,<br>24-27 |
| A         | abstract<br>figure 3<br>section 1.1.4 "Integration into the present codec: Decoder"<br>-/-                                                                                                                                                                                                                                                                                                                                                                          | 5-11,<br>13-18,23          |



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

17 December 2020

Date of mailing of the international search report

12/01/2021

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

Stoufs, Maryse

## INTERNATIONAL SEARCH REPORT

International application No

PCT/EP2020/078460

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages                                                                                                                                                                                                                                                                   | Relevant to claim No. |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
|           | table 1                                                                                                                                                                                                                                                                                                                                              |                       |
| X         | -----<br>MINNEN D ET AL: "Spatially adaptive image compression using a tiled deep network", 2017 IEEE INTERNATIONAL CONFERENCE ON IMAGE PROCESSING (ICIP), IEEE, 17 September 2017 (2017-09-17), pages 2796-2800, XP033323085, DOI: 10.1109/ICIP.2017.8296792 [retrieved on 2018-02-20]                                                              | 1-10,21, 23,25-27     |
| A         | figures 2, 3<br>section 2.1. "Spatial Context Prediction"<br>section 2.2. "Residual Encoding with Recurrent Networks"                                                                                                                                                                                                                                | 11-18, 20,22,24       |
| X,P       | -----<br>BRAND FABIAN ET AL: "Intra Frame Prediction for Video Coding Using a Conditional Autoencoder Approach", 2019 PICTURE CODING SYMPOSIUM (PCS), IEEE, 12 November 2019 (2019-11-12), pages 1-5, XP033688172, DOI: 10.1109/PCS48520.2019.8954546 [retrieved on 2020-01-08]<br>figure 1<br>section III. "Out Method"                             | 1-18, 20-27           |
| A         | -----<br>DEEPAK PATHAK ET AL: "Context Encoders: Feature Learning by Inpainting", 2016 IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), 21 November 2016 (2016-11-21), pages 2536-2544, XP055535158, DOI: 10.1109/CVPR.2016.278<br>ISBN: 978-1-4673-8851-1<br>the whole document                                                   | 1-18, 20-27           |
| A         | -----<br>WANG LI ET AL: "Enhancing HEVC Spatial Prediction by Context-based Learning", ICASSP 2019 - 2019 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING (ICASSP), IEEE, 12 May 2019 (2019-05-12), pages 4035-4039, XP033566175, DOI: 10.1109/ICASSP.2019.8683624 [retrieved on 2019-04-04]<br>the whole document<br>----- | 1-18, 20-27           |

## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/EP2020/078460

### 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.: 19  
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:  
see FURTHER INFORMATION sheet PCT/ISA/210
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:

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.:

#### 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**

Continuation of Box II.2

Claims Nos.: 19

Claim 19 specifies "the input data further comprises the intra prediction mode (L) of the block located on the left side of the block being encoded or decoded and the intra prediction mode (A) of the block located above the block being encoded or decoded". It is not clear how this input data is used within the neural network in combination with the input data being context information.

Turning to the description for clarification (see Fig. 17-18, p. 16 lines 14-33), the skilled person does not find further instructions as to how these features are effectively to be put into practice.

Fig. 17-18 merely show a drawing of a neural network where modes L and A seem to become nodes of the network? It is unclear what weights are then given to these nodes or how the layer to include this information should be selected.

Therefore it appears the additional features of said claim fail to meet the requirements of Article 5 and 6 PCT in such a way that no opinion may be provided for said claim 19.

The applicant's attention is drawn to the fact that claims relating to inventions in respect of which no international search report has been established need not be the subject of an international preliminary examination (Rule 66.1(e) PCT). The applicant is advised that the EPO policy when acting as an International Preliminary Examining Authority is normally not to carry out a preliminary examination on matter which has not been searched. This is the case irrespective of whether or not the claims are amended following receipt of the search report or during any Chapter II procedure. If the application proceeds into the regional phase before the EPO, the applicant is reminded that a search may be carried out during examination before the EPO (see EPO Guidelines C-IV, 7.2), should the problems which led to the Article 17(2) PCT declaration be overcome.