



(51) International Patent Classification:

H04N 19/593 (2014.01) H04N 19/176 (2014.01)

H04N 19/105 (2014.01) H04N 19/11 (2014.01)

H04N 19/119 (2014.01) G06T 9/00 (2006.01)

H04N 19/136 (2014.01) H04N 19/159 (2014.01)

(21) International Application Number:

PCT/FI2020/050327

(22) International Filing Date:

15 May 2020 (15.05.2020)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

20195421 22 May 2019 (22.05.2019) FI

(71) Applicant: NOKIA TECHNOLOGIES OY [FI/FI];
Karakaari 7, 02610 Espoo (FI).

(72) Inventor: GHAZNAVI YOUVALARI, Ramin;
Myrskyläkatu 12 A 32, 33900 Tampere (FI).

(74) Agent: NOKIA TECHNOLOGIES OY et al.; Ari Aarnio,
IPR Department, Karakaari 7, 02610 Espoo (FI).

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

(54) Title: A METHOD, AN APPARATUS AND A COMPUTER PROGRAM PRODUCT FOR VIDEO ENCODING AND VIDEO DECODING

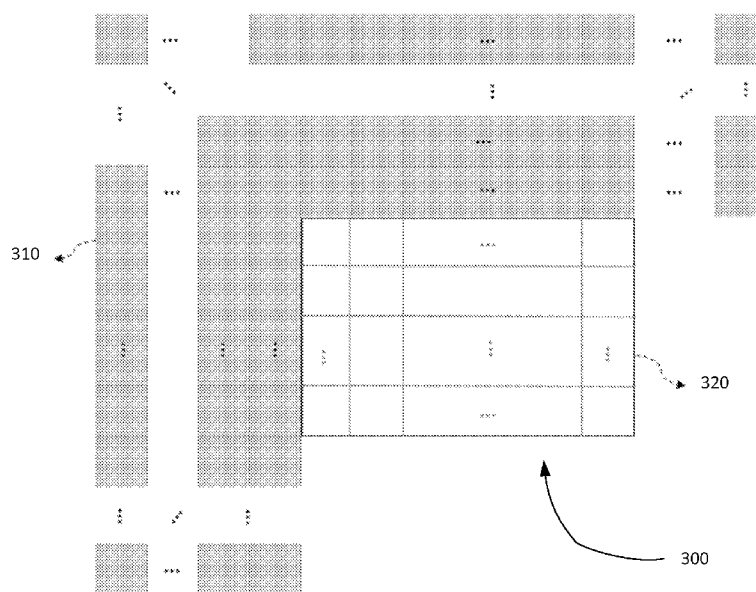


Fig. 3

(57) Abstract: The embodiments relate to a method and technical equipment, wherein the method comprises receiving a source picture; partitioning the source picture into a set of non-overlapping blocks; for a block, obtaining at least one sample/pixel value and its location from at least one neighboring block; deriving a prediction model for the block that relates the sample(s) of neighboring block(s) to the corresponding location(s) of the neighboring block(s); predicting a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area; deriving at least one other prediction model for at least one other prediction area using the predicted samples and their locations from the prediction area being previously predicted; and predicting a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least one other prediction area.



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, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

- *without international search report and to be republished upon receipt of that report (Rule 48.2(g))*

A METHOD, AN APPARATUS AND A COMPUTER PROGRAM PRODUCT FOR VIDEO ENCODING AND VIDEO DECODING

Technical Field

5

The present solution generally relates to video encoding and video decoding. In particular, the solution relates to adaptive intra prediction used in video coding.

Background

10

A video coding system may comprise an encoder that transforms an input video into a compressed representation suited for storage/transmission, and a decoder that can uncompress the compressed video representation back into a viewable form. The encoder may discard some information in the original video sequence in order to represent the video in a more compact form, for example, to enable the storage/transmission of the video information at a lower bitrate than otherwise might be needed.

15

20

25

Hybrid video codecs may encode the video information in two phases. Firstly, sample values (i.e. pixel values) in a certain picture area are predicted, e.g., by motion compensation means or by spatial means. Secondly, the prediction error, i.e. the difference between the prediction block of samples and the original block of samples, is coded. The video decoder on the other hand reconstructs the output video by applying prediction means similar to the encoder to form a prediction representation of the sample blocks and prediction error decoding. After applying prediction and prediction error decoding means, the decoder sums up the prediction and prediction error signals to form the output video frame.

Summary

30

The scope of protection sought for various embodiments of the invention is set out by the independent claims. The embodiments and features, if any, described in this specification that do not fall under the scope of the independent claims are to be interpreted as examples useful for understanding various embodiments of the invention.

35

Various aspects include a method, an apparatus and a computer readable medium comprising a computer program stored therein, which are characterized by what is

stated in the independent claims. Various embodiments are disclosed in the dependent claims.

5 According to a first aspect, there is provided a method comprising receiving a source picture; partitioning the source picture into a set of non-overlapping blocks; for a block, obtaining at least one sample/pixel value and its location from at least one neighboring block; deriving a prediction model for the block that relates the sample(s) of neighboring block(s) to the corresponding location(s) of the neighboring block(s); predicting a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area; deriving at least one other prediction model for at least one other prediction area using the predicted samples and their locations from the prediction area being previously predicted; and predicting a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least one other prediction area.

20 According to a second aspect, there is provided an apparatus comprising means for receiving a source picture; means for partitioning the source picture into a set of non-overlapping blocks; for a block, means for obtaining at least one sample/pixel value and its location from at least one neighboring block; means for deriving a prediction model for a block that relates the sample(s) of a neighboring block(s) to the corresponding location(s) of the neighboring block(s); means for predicting a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area; means for deriving at least one other prediction model for at least one other prediction area using the predicted samples and their locations from the prediction area being previously predicted; and means for predicting a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least one other prediction area.

30 According to a third aspect, there is provided a computer program product comprising computer program code configured to, when executed on at least one processor, cause an apparatus or a system to receive a source picture; partition the source picture into a set of non-overlapping blocks; for a block, obtain at least one sample/pixel value and its location from at least one neighboring block; derive a prediction model for the block that relates the sample(s) of neighboring block(s) to the corresponding location(s) of the neighboring block(s); predict a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area; derive at least one other prediction model for at least one other

prediction area using the predicted samples and their locations from the prediction area being previously predicted; and predict a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least one other prediction area.

5

According to an embodiment, the sample is a pixel or a sub-block.

According to an embodiment, the method is continued for each block.

10 According to an embodiment, the neighboring block is one of the following: a block on a top of a current block, a block on the left of a current block, a block on top-left of a current block, a block on bottom-left of a current block, a block on top-right of a current block.

15 According to an embodiment, a derived prediction models are stored for certain region in a picture.

According to an embodiment, a prediction model is derived by using samples and their locations of a certain sample component by samples of another sample component.

20

According to an embodiment, an existing prediction model is derived from a storage and the retrieved existing prediction model is used together with at the first or said at least one other derived prediction model.

25 According to an embodiment, the model derivation is executed at an encoder and/or in a decoder.

According to an embodiment, the apparatus comprises at least one processor and a memory including computer program code.

30

According to an embodiment, the computer program product is embodied on a non-transitory computer readable medium.

Description of the Drawings

35

In the following, various embodiments will be described in more detail with reference to the appended drawings, in which

Fig. 1 shows an encoding process according to an embodiment;

Fig. 2 shows a decoding process according to an embodiment;

Fig. 3 shows a method according to an embodiment for intra prediction using neighboring samples;

Figs. 4a-b show a method according to another embodiment for intra prediction;

Figs. 5a-b show a method according to another embodiment for intra prediction;

Figs. 6a-d show a method according to yet another embodiment for intra prediction;

Fig. 7 shows an example of available and non-available samples for prediction;

Fig. 8 is a flowchart illustrating a method according to an embodiment;

Fig. 9 is a flowchart illustrating a method according to another embodiment; and

Fig. 10 shows an apparatus according to an embodiment.

Description of Example Embodiments

In the following, several embodiments will be described in the context of one video coding arrangement. It is to be noted, however, that the invention is not limited to this particular arrangement. For example, the invention may be applicable to video coding systems like streaming system, DVD (Digital Versatile Disc) players, digital television receivers, personal video recorders, systems and computer programs on personal computers, handheld computers and communication devices, as well as network elements such as transcoders and cloud computing arrangements where video data is handled.

In the following, several embodiments are described using the convention of referring to (de)coding, which indicates that the embodiments may apply to decoding and/or encoding.

The Advanced Video Coding standard (which may be abbreviated AVC or H.264/AVC) was developed by the Joint Video Team (JVT) of the Video Coding Experts Group (VCEG) of the Telecommunications Standardization Sector of International

Telecommunication Union (ITU-T) and the Moving Picture Experts Group (MPEG) of International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC). The H.264/AVC standard is published by both parent standardization organizations, and it is referred to as ITU-T Recommendation H.264 and ISO/IEC International Standard 14496-10, also known as MPEG-4 Part 10 Advanced Video Coding (AVC). There have been multiple versions of the H.264/AVC standard, each integrating new extensions or features to the specification. These extensions include Scalable Video Coding (SVC) and Multiview Video Coding (MVC).

The High Efficiency Video Coding standard (which may be abbreviated HEVC or H.265/HEVC) was developed by the Joint Collaborative Team – Video Coding (JCT-VC) of VCEG and MPEG. The standard is published by both parent standardization organizations, and it is referred to as ITU-T Recommendation H.265 and ISO/IEC International Standard 23008-2, also known as MPEG-H Part 2 High Efficiency Video Coding (HEVC). Extensions to H.265/HEVC include scalable, multiview, three-dimensional, and fidelity range extensions, which may be referred to as SHVC, MV-HEVC, 3D-HEVC, and REXT, respectively. The references in this description to H.265/HEVC, SHVC, MV-HEVC, 3D-HEVC and REXT that have been made for the purpose of understanding definitions, structures or concepts of these standard specifications are to be understood to be references to the latest versions of these standards that were available before the date of this application, unless otherwise indicated.

Some key definitions, bitstream and coding structures, and concepts of H.264/AVC and HEVC and some of their extensions are described in this section as an example of a video encoder, decoder, encoding method, decoding method, and a bitstream structure, wherein the embodiments may be implemented. Some of the key definitions, bitstream and coding structures, and concepts of H.264/AVC are the same as in HEVC standard – hence, they are described below jointly. The aspects of various embodiments are not limited to H.264/AVC or HEVC or their extensions, but rather the description is given for one possible basis on top of which the present embodiments may be partly or fully realized.

In the description of existing standards as well as in the description of example embodiments, a syntax element may be defined as an element of data represented in the bitstream. A syntax structure may be defined as zero or more syntax elements present together in the bitstream in a specified order.

Similarly, to many earlier video coding standards, the bitstream syntax and semantics as well as the decoding process for error-free bitstreams are specified in H.264/AVC and HEVC. The encoding process is not specified, but encoders must generate conforming bitstreams. Bitstream and decoder conformance can be verified with the Hypothetical Reference Decoder (HRD). The standards contain coding tools that help in coping with transmission errors and losses, but the use of the tools in encoding is optional and no decoding process has been specified for erroneous bitstreams.

The elementary unit for the input to an H.264/AVC or HEVC encoder and the output of an H.264/AVC or HEVC decoder, respectively, is a picture. A picture given as an input to an encoder may also be referred to as a source picture, and a picture decoded by a decoder may be referred to as a decoded picture.

The source and decoded pictures may each be comprised of one or more sample arrays, such as one of the following sets of sample arrays, wherein each of the samples may represent one color component :

- Luma (Y) only (monochrome).
- Luma and two chroma (YCbCr or YCgCo).
- Green, Blue and Red (GBR, also known as RGB).

Arrays representing other unspecified monochrome or tri-stimulus color samplings (for example, YZX, also known as XYZ).

In the following, these arrays may be referred to as luma (or L or Y) and chroma, where the two chroma arrays may be referred to as Cb and Cr; regardless of the actual color representation method in use. The actual color representation method in use may be indicated e.g. in a coded bitstream e.g. using the Video Usability Information (VUI) syntax of H.264/AVC and/or HEVC. A component may be defined as an array or a single sample from one of the three sample arrays (luma and two chroma) or the array or a single sample of the array that compose a picture in monochrome format.

In H.264/AVC and HEVC, a picture may either be a frame or a field. A frame comprises a matrix of luma samples and possibly the corresponding chroma samples. A field is a set of alternate sample rows of a frame. Fields may be used as encoder input for example when the source signal is interlaced. Chroma sample arrays may be absent (and hence monochrome sampling may be in use) or may be subsampled when compared to luma sample arrays. Some chroma formats may be summarized as follows:

- In monochrome sampling there is only one sample array, which may be nominally considered the luma array.
- In 4:2:0 sampling, each of the two chroma arrays has half the height and half the width of the luma array.
- 5 – In 4:2:2 sampling, each of the two chroma arrays has the same height and half the width of the luma array.
- In 4:4:4 sampling when no separate color planes are in use, each of the two chroma arrays has the same height and width as the luma array.

10 In H.264/AVC and HEVC, it is possible to code sample arrays as separate color planes into the bitstream and respectively decode separately coded color planes from the bitstream. When separate color planes are in use, each one of them is separately processed (by the encoder and/or the decoder) as a picture with monochrome sampling.

15 When chroma subsampling is in use (e.g. 4:2:0 or 4:2:2 chroma sampling), the location of chroma samples with respect to luma samples may be determined in the encoder side (e.g. as pre-processing step or as part of encoding). The chroma sample positions with respect to luma sample positions may be pre-defined for example in a coding
20 standard, such as H.264/AVC or HEVC, or may be indicated in the bitstream for example as part of VUI of H.264/AVC or HEVC.

Generally, the source video sequence(s) provided as input for encoding may either represent interlaced source content or progressive source content. Fields of opposite parity have been captured at different times for interlaced source content. Progressive
25 source content contains captured frames. An encoder may encode fields of interlaced source content in two ways: a pair of interlaced fields may be coded into a coded frame or a field may be coded as a coded field. Likewise, an encoder may encode frames of progressive source content in two ways: a frame of progressive source content may
30 be coded into a coded frame or a pair of coded fields. A field pair or a complementary field pair may be defined as two fields next to each other in decoding and/or output order, having opposite parity (i.e. one being a top field and another being a bottom field) and neither belonging to any other complementary field pair. Some video coding standards or schemes allow mixing of coded frames and coded fields in the same
35 coded video sequence. Moreover, predicting a coded field from a field in a coded frame and/or predicting a coded frame for a complementary field pair (coded as fields) may be enabled in encoding and/or decoding.

A partitioning may be defined as a division of a set into subsets such that each element of the set is in exactly one of the subsets. A picture partitioning may be defined as a division of a picture into smaller non-overlapping units. A block partitioning may be defined as a division of a block into smaller non-overlapping units, such as sub-blocks.

5 In some cases, term block partitioning may be considered to cover multiple levels of partitioning, for example partitioning of a picture into slices, and partitioning of each slice into smaller units, such as macroblocks of H.264/AVC. It is noted that the same unit, such as a picture, may have more than one partitioning. For example, a coding unit of HEVC may be partitioned into prediction units and separately by another quadtree into transform units.

Hybrid video codecs, for example ITU-T H.263, H.264/AVC and HEVC, may encode the video information in two phases. At first, pixel value in a certain picture area (or “block”) are predicted for example by motion compensation means or by spatial means.

15 In the first phase, predictive coding may be applied, for example, as so-called sample prediction and/or so-called syntax prediction.

In the sample prediction, pixel or sample values in a certain picture area or “block” are predicted. These pixel or sample values can be predicted, for example, using one or more of the following ways:

- motion compensation mechanism;
- intra prediction mechanism.

25 Motion compensation mechanisms (which may also be referred to as inter prediction, temporal prediction or motion-compensation temporal prediction or motion-compensated prediction or MCP) involve finding and indicating an area in one of the previously encoded video frames that corresponds closely to the block being coded. Inter prediction may reduce temporal redundancy.

30 Intra prediction, where pixel or sample values can be predicted by spatial mechanisms, involve finding and indicating a spatial region relationship. Intra prediction can be performed in spatial or transform domain, i.e., either sample values or transform coefficients can be predicted. Intra prediction may be exploited in intra coding, where no inter-prediction is applied.

35 In the syntax prediction, which may also be referred to as parameter prediction, syntax elements and/or syntax element values and/or variables derived from syntax elements are predicted from syntax elements (de)coded earlier and/or variables derived earlier. Non-limiting examples of syntax prediction are provided below:

- In motion vector prediction, motion vectors, e.g. for inter and/or inter-view prediction may be coded differentially with respect to a block-specific predicted motion vector. In many video codecs, the predicted motion vectors are created in a predefined way, for example by calculating the media of the encoded or decoded motion vectors of the adjacent blocks. Another way to create motion vector predictions, sometimes referred to as advanced motion vector prediction (AMVP), is to generate a list of candidate predictions from adjacent blocks and/or co-located blocks in temporal reference pictures and signaling the chosen candidate as the motion vector predictor. In addition to predicting the motion vector values, the reference index of previously coded/decoded picture can be predicted. Differential coding of motion vectors may be disabled according to slice boundaries.
- The block partitioning, e.g. from CTU to CUs and down to PUs, may be predicted.
- In filter parameter prediction, the filtering parameters e.g. for sample adaptive offset may be predicted.

Prediction approaches using image information from a previously coded image can also be called as inter prediction methods which may also be referred to as temporal prediction and motion compensation. Prediction approaches using image information within the same image can also be called as intra prediction methods.

The second phase is coding the error between the prediction block of samples and the original block of samples. This may be accomplished by transforming the difference in sample values using a specified transform. This transform may be e.g. a Discrete Cosine Transform (DCT) or a variant thereof. After transforming the difference, the transformed difference is quantized, and entropy coded.

By varying the fidelity of the quantization process, encoder can control the balance between the accuracy of the pixel representation (picture quality) and size of the resulting coded video representation (file size of transmission bitrate).

The decoder reconstructs the output video by applying a prediction mechanism similar to that used by the encoder in order to form a prediction representation of the sample blocks (using the motion or spatial information created by the encoder and included in the compressed representation of the image) and prediction error decoding (the inverse operation of the prediction error coding to recover the quantized error signal in the spatial domain).

After applying sample prediction and error decoding processes the decoder combines the prediction and the prediction error signals (the sample values) to form the output video frame.

5 An example of an encoding process is illustrated in Fig. 1. Fig. 1 illustrates an image to be encoded (I_n); a prediction representation of an image block (P'_n); a prediction error signal (D_n); a reconstructed prediction error signal (D'_n); a preliminary reconstructed image (I'_n); a final reconstructed image (R'_n); a transform (T) and inverse transform (T^{-1}); a quantization (Q) and inverse quantization (Q^{-1}); entropy encoding (E);
10 a reference frame memory (RFM); inter prediction (P_{inter}); intra prediction (P_{intra}); mode selection (MS) and filtering (F). An example of a decoding process is illustrated in Fig. 2. Fig. 2 illustrates a prediction representation of an image block (P'_n); a reconstructed prediction error signal (D'_n); a preliminary reconstructed image (I'_n); a final reconstructed image (R'_n); an inverse transform (T^{-1}); an inverse quantization (Q^{-1}); an
15 entropy decoding (E^{-1}); a reference frame memory (RFM); a prediction (either inter or intra) (P); and filtering (F).

The decoder (and encoder) may also apply additional filtering processes in order to improve the quality of the output video before passing it for display and/or storing as a
20 prediction reference for the forthcoming pictures in the video sequence.

In many video codecs, including H.264/AVC and HEVC, motion information is indicated by motion vectors associated with each motion compensated image block. Each of these motion vectors represents the displacement of the image block in the picture to
25 be coded (in the encoder) or decoded (at the decoder), and the prediction source block in one of the previously coded or decoded image (or pictures). H.264/AVC and HEVC, as many other video compression standards, a picture is divided into a mesh of rectangles, for each of which a similar block in one of the reference pictures is indicated for inter prediction. The location of the prediction block is coded as a motion vector that
30 indicates the position of the prediction block relative to the block being coded.

Current intra prediction methods for image/video compression are not efficient, and still intra coded images or blocks consume significant amount of bitrate compared to the inter predicted frames or blocks in a video. One of the aspects that is not
35 considered in the intra prediction methods is the relation between the position/location of the samples and their pixel values. The texture of an image may include different behaviors in different parts of the image plane, for example, there could be certain deformations, samplings, etc. in certain parts, so that the conventional intra prediction methods are not capable of capturing such sample distribution in an efficient way. This

is due to the fact that most of the intra prediction methods use a specific prediction angle/direction for predicting a block of samples based on its neighbors.

5 The present embodiments relates to a method for intra prediction in video compression. The intra prediction method according to embodiments attempts to model the samples inside a block based on the available neighboring samples and their locations. For that, a portion of the neighboring samples from the left, above, above-left, above-right and bottom-left of the block along with their coordinates (x, y) is collected, as shown in Figure 3. Then a prediction model is derived according to the
10 collected information from neighbors. The derived prediction model is used for predicting the samples inside the block using the location of each sample inside the prediction block.

15 The new intra prediction method is then used as an additional intra prediction mode along with the existing modes or can replace one or more of the existing ones in the codec (e.g., AVC, HEVC, VVC, etc.)

In the following, the intra prediction method according to present embodiment is described in more detailed manner.

20 The method according to the present embodiments derives a prediction model which can simulate the sample distribution behavior in different parts of the image/video.

25 A method according to an embodiment is shown in Figure 3. This embodiment relates to intra prediction using neighboring samples. In the embodiment of Figure 3, the prediction model (later referred to as “first model”) for prediction block’s 300 samples 320 is obtained by using neighboring samples 310, i.e. samples from the neighborhood of the block, which samples have already been predicted and reconstructed. Moreover, the location of these neighboring samples is also collected (E.g., x, y coordinates
30 relative to the top-left corner of the picture or current block). The location information may then be used for modeling the sample behavior inside the current block 300.

To that end, a prediction model can be derived in a way that relates the sample or pixel value P of a block 300 to its (x, y) location, neighboring sample values and neighboring
35 samples’ locations with at least one weight and/or at least one offset.

Different prediction models can be used for this purpose. For example, the prediction method can model the samples 320 based on both or either of the x and y locations of neighboring samples 310. In other words, the prediction model can be linear,

polynomial, etc. Some examples of the prediction model, in which sample P in location (x, y) is calculated are given below:

$$P(x, y) = ax^2 + by^2 + cx + dy + e \quad (1)$$

$$P(x, y) = ax + by + c \quad (2)$$

$$P(x, y) = a(x - x_0) + b(y - y_0) + c + P(x_0, y_0) \quad (3)$$

$$P(x, y) = a(y - y_0) + b + P(x_0, y_0) \quad (4)$$

$$P(x, y) = a(y - y_0) + b + P(x_0, y_0) \quad (5)$$

$$P(x, y) = ax + c \quad (6)$$

$$P(x, y) = by + c \quad (7)$$

In general, the prediction model can be defined as below, in which the sample P in location (x, y) is calculated based on the functions $f(x)$ and $f(y)$:

$$P(x, y) = af(x) + bf(y) + c \quad (8)$$

In an alternative approach, a sample P value can be calculated in each direction (i.e., x and y) relative to its direction or vice versa and the final predicted sample can be calculated based on certain weights between them:

$$\begin{cases} P_x = ax + c_0 \\ P_y = by + c_1 \end{cases} \quad (9)$$

$$\text{Then: } P(x, y) = W_0 P_x + W_1 P_y \quad (10)$$

In equation (10), the weights W_0 and W_1 can be calculated in different ways, e.g. based on the properties of the block (e.g., height, width, Luma/Chroma) or can be calculated based on the learning process from the neighboring blocks.

In the above examples, x and y refers to the location of the predicted sample in (x, y) coordinates. The remaining parameters (i.e., $a, b, c, d, e, P(x_0, y_0)$) are the parameters that are derived using the neighboring information (neighboring sample values and their locations) by using a training (or parameter derivation or learning process). The training (or parameter derivation or learning) process makes use of the neighboring

sample/pixel values along with their (x, y) location for calculating the relation of the sample/pixel values to the location of them. The training (parameter derivation or learning) process can be done in various ways e.g. by using one or more linear or polynomial regression methods. In an alternative approach, a set of neural networks (NNs) may be used for the learning process. The training (parameter derivation or learning) process is not limited to the described ones and any method can be used for such purpose considering both sample/pixel values and their (x, y) locations.

The model parameters of the above prediction model can be calculated for each block 300 based on the sample/pixel 310 values and their locations from the available neighboring blocks. A neighboring block can be one or more of the following: a block on a top of the current block, a block on the left of the current block, a block on top-left of the current block, a block on bottom-left of the current block, a block on top-right of the current block. For that, the collected neighboring information is used for a training or derivation process for example with linear regression, polynomial regression, logical regression, RANSAC (random sample consensus), etc. to calculate the mentioned parameters.

The parameter derivation process can be done also for example with gradient estimation approach.

$$a = \frac{1}{N-1} \sum_{i=1}^N g_x(P_n(x_i)) \quad (11)$$

$$b = \frac{1}{M-1} \sum_{j=1}^M g_y(P_n(y_j)) \quad (12)$$

where $P_n(x_i)$ and $P_n(y_i)$ are the sample values from the neighboring above and left of the block. Moreover, g_x and g_y are the gradient estimations in horizontal and vertical directions, respectively.

It is appreciated that a neighboring block can be an adjacent and/or non-adjacent block in a neighboring block can be an adjacent and/or non-adjacent block in a neighboring region. Neighboring region relates to blocks locating at certain distance from the current block, wherein the distance may be defined by one or more blocks in a certain direction, e.g. up or left. The size of the neighboring region may be selected based on the size of the current block. For example, the blocks in the top, left, top-left, bottom-left and top-right regions of the current block may be used to train the model. The size of the neighboring block may be considered in training process.

According to another embodiment, the encoder may derive different prediction models for one block based on the neighboring samples and locations. For example, one

prediction model may be derived based on the sample and locations of the left-side neighbor, one prediction model may be derived based on samples and their locations from the top-side neighbor, and another prediction model based on both left- and top-side neighbor samples and their locations. In such case, the best performing prediction model may be selected out of these prediction models. A flag or a predefined index may be encoded to the bitstream in order to inform the decoder that which prediction model is used. Alternatively, the prediction model parameters may be transmitted/encoded into the bitstream. In an alternative way, the decoder may identify which prediction model out of the tested ones are selected in the encoder side. The latter one requires that the encoder and decoder use the same criteria for selecting among the tested prediction model derivation schemes.

The parameter derivation (training) process may be applied more than once in order to have more accurate parameters for prediction. This is particularly important for removing the outliers in the collected samples from the neighborhood.

According to another embodiment, the block can be divided into multiple subblocks, and a separate training or model derivation is applied for each subblock. Figures 4 – 6 illustrate examples on this embodiment.

As is realized e.g. from Figure 4a, a block 400 is divided into at least two sub-blocks 421, 422. The first model derivation that has been described in above, is applied to the first subblock (e.g., the top-left subblock) 421, which represents the first prediction area, and the first subblock 421 is predicted by using the first model and using the sample values and locations of available neighboring samples 420. For the second subblock 422, which represents the second prediction area, a separate model is derived that may include samples and their locations from the first subblock 421 along with samples and locations of default neighboring block(s) 420. This is illustrated in Figure 4b, where the available samples from first predicted area 425 predicting the second prediction area 422 are referred to with a reference number 430.

Similarly, in Figure 5a, the block 500 is divided into two subblocks 521, 522. The first subblock (the first prediction area) 521 is predicted based on the first model which is derived from the neighboring available samples 520 (sample values and locations). However, as in Figure 4b, for the second subblock (the second prediction area) 522, shown in Figure 5b, samples from the first subblock 525 are available. The values and locations of these available samples 530 from the first subblock 525 are used for predicting the second subblock 522. Hence, a separate model is derived for this

subblock that may include also the samples and their locations 530 from the first subblock 525 along with the default neighboring samples 520.

Figure 6 illustrates an example where a block 600 is divided into 4 subblocks 621, 622, 623, 624.

The division into subblocks 621, 622, 623, 624 may be based on a predefined structure e.g., based on the size of the block, or it can be based on a separate learning from the neighboring samples. In other words, a learning operation or a simple neighboring check may be applied in order to decide the subblock partitioning based on the sample's values in neighbors. According to an alternative embodiment, the subblock partitioning may be inherited from the neighboring block(s).

In Figure 6a, the current prediction area 621 is predicted based on the first model which is derived from the neighboring available samples 620, comprising both sample values and the (x, y) locations of them. As shown in Figure 6b, the current prediction area 622 utilizes also the sample/pixel values and the (x, y) location of the area that has been predicted in Figure 6a, i.e. predicted area 621, which is available for modeling for the current prediction area 622. As shown in Figure 6c, both the first and the second areas 621, 622 have been predicted, and therefore the sample/pixel values and the (x, y) location of predicted areas are thus also available for modeling for the current prediction area 623. Finally, as shown in Figure 6d, the sample/pixel values and the (x, y) location of each predicted area 621, 622, 623 are available for modelling for the current prediction area 624.

The partitioning may result in partitions having different sizes. Thus, the partitioning may be done in an unequal way based on the block properties, e.g., texture and size of the current and neighboring blocks, etc.

In block-based image/video coding, due to the raster scan coding order, the samples from right and bottom sides of the blocks are not available for prediction purposes. This is also illustrated in Figure 7 showing the available samples 720 for prediction and non-available samples 730.

The method according to the present embodiments may be used for predicting the non-available areas from the right and/or bottom sides of the prediction block by using the sample values or locations of available samples. The predicted samples for the non-available areas can be used, for example,

- in the existing intra prediction methods;

- in re-modelling the prediction model, which is derived for the block, based on the earlier available samples;
- for further tuning the prediction model;
- jointly with the earlier prediction model. For example, two sets of prediction can be calculated. The first prediction is based on the earlier prediction model that is derived based on the available samples (from left and/or above sides). The second prediction model that is derived based on the samples from the predicted non-available areas (the second prediction may also include the samples which are available by default). The final prediction may use either or both of these predictions with a certain weighting between them. Higher weights may be applied to the first prediction since it uses the default available samples not the predicted ones (the predicted ones may not be very accurate).

The non-available area prediction is not limited to the right and bottom side areas. When a block is located in a picture boundary, samples from the side of the block which is in picture boundary (can be left, above-left, above or all) are not available. In conventional prediction scenario in existing codecs, these non-available ones are padded from the available ones. For example, if the above samples are not available but the left side samples are available, then the left samples are projected or padded to the above side for having better prediction. If none of the sides are available (e.g., block is located in top-left boundary area of picture) as fixed value may be used for the non-available areas. However, by using the present embodiments, these areas can be predicted from the samples that are available from at least one of the sides for the final prediction.

The derived prediction model (consisting of its weights and/or coefficients) of each block (described in above), may be used for intra prediction of the neighboring blocks. For example, a shared memory may be defined where these models are stored for a certain region in an image. In this case, for predicting the current block's samples, a model can be derived based on the neighboring samples (i.e. sample values and locations), moreover, the available models from the shared memory can be used for having a more efficient prediction. The shared models from the previous blocks may be scaled or tuned before using them for the current block. The scaling or tuning may be done for example based on the distance and/or size of the current block to the shared model's block.

The model derivation may make use of the neighboring blocks prediction mode for deciding which side of the neighboring samples can be used for model derivation. This

means that the neighboring block's prediction mode can be inferred to get the direction of samples and also the weighting for the final model parameters.

The present embodiments can also be used for cross-component scheme. The samples of a block in a certain sample component (e.g., YUV or YCbCr) may use the corresponding samples and their locations in different sample component for deriving the prediction model. For example, chroma components (Cb, Cr) may use the corresponding samples in Luma channel for model derivation. Cross chroma prediction can be also used, i.e., Cb from Cr or vice versa. Another approach may be to derive the Luma prediction model from one or more of the chroma (Cb, Cr) corresponding samples. In these cases, sample modification or scaling may be applied to the samples of the channel that are used for prediction model derivation.

According to another example, the present embodiment may be used jointly with other existing prediction models. In other words, there may be the prediction of a block done in multiple steps. For example, a first prediction of block can be done based on one or more of the existing prediction models; a second prediction can be done based on the method according to the present embodiments. The final prediction of a sample can be done by joint prediction as shown below:

$$P_{final} = W_0 P_0 + W_1 P_{prop} \quad (13)$$

In this formula, P_{final} is the final prediction of the sample. P_0 is predicted sample by using one or more of the existing methods in the codec and P_{prop} is the predicted sample based on the method according to the present embodiments. Furthermore, W_0 and W_1 are the weights that are used for joint prediction. In a simple case, equal weights could be used that is $W_0 = W_1 = \frac{1}{2}$.

It is appreciated that the joint prediction method may consist of more than two prediction methods with more than two weights.

Fig. 8 is a flowchart illustrating a method according to an embodiment. A method comprises receiving 810 a source picture; partitioning 820 the source picture into a set of non-overlapping blocks; for a block, obtaining 830 at least one sample/pixel value and its location from at least one neighboring block; deriving 840 a prediction model for the block that relates the sample(s) of neighboring block(s) to the corresponding location(s) of the neighboring block(s); predicting 850 a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area; deriving 860 a second prediction model for a second

prediction area using the predicted samples and their locations from the first prediction area; and predicting 870 a sample value of the sample of a second prediction area based on the second prediction model and the sample location inside said at least one other prediction area.

5

Figure 9 is a flowchart illustrating a method according to another embodiment. The method comprises receiving 910 a source picture; partitioning 920 the source picture into a set of non-overlapping blocks; for a block, obtaining 930 at least one sample/pixel value and its location from at least one neighboring block; deriving 940 a prediction model for the block that relates the sample(s) of neighboring block(s) to the corresponding location(s) of the neighboring block(s); predicting 950 a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area; deriving 960 at least one other prediction model for at least one other prediction area using the predicted samples and their locations from the prediction area being previously predicted; and predicting 970 a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least one other prediction area.

10

15

20

It is appreciated that term “receiving” is understood here to mean that the information is read from a memory or received over a communications connection.

25

30

35

An apparatus according to an embodiment comprises means for receiving a source picture; means for partitioning the source picture into a set of non-overlapping blocks; for a block, means for obtaining at least one sample/pixel value and its location from at least one neighboring block; means for deriving a prediction model for the block that relates the neighboring sample(s)/pixel(s) value(s) to their corresponding location; means for predicting a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area; deriving at least one other prediction model for at least one other prediction area using the predicted samples and their locations from the prediction area being previously predicted; and means predicting a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least one other prediction area. The means comprises at least one processor, and a memory including a computer program code, wherein the processor may further comprise processor circuitry. The memory and the computer program code are configured to, with the at least one processor, cause the apparatus to perform the method of Figure 8 or Figure 9 according to various embodiments.

An apparatus according to an embodiment is illustrated in Figure 10. An apparatus of this embodiment is a camera having multiple lenses and imaging sensors, but also other types of cameras may be used to capture wide view images and/or wide view video.

5

The terms wide view image and wide view video mean an image and a video, respectively, which comprise visual information having a relatively large viewing angle, larger than 100 degrees. Hence, a so called 360 panorama image/video as well as images/videos captured by using a fisheye lens may also be called as a wide view image/video in this specification. More generally, the wide view image/video may mean an image/video in which some kind of projection distortion may occur when a direction of view changes between successive images or frames of the video so that a transform may be needed to find out co-located samples from a reference image or a reference frame. This will be described in more detail later in this specification.

10

15

The camera 2700 of Figure 10 comprises two or more camera units 2701 and is capable of capturing wide view images and/or wide view video. Each camera unit 2701 is located at a different location in the multi-camera system and may have a different orientation with respect to other camera units 2501. As an example, the camera units 2701 may have an omnidirectional constellation so that it has a 360-viewing angle in a 3D-space. In other words, such camera 2700 may be able to see each direction of a scene so that each spot of the scene around the camera 2700 can be viewed by at least one camera unit 2701.

20

25

The camera 2700 of Figure 10 may also comprise a processor 2704 for controlling the operations of the camera 2700. There may also be a memory 2706 for storing data and computer code to be executed by the processor 2704, and a transceiver 2708 for communicating with, for example, a communication network and/or other devices in a wireless and/or wired manner. The camera 2700 may further comprise a user interface (UI) 2710 for displaying information to the user, for generating audible signals and/or for receiving user input. However, the camera 2700 need not comprise each feature mentioned above or may comprise other features as well. For example, there may be electric and/or mechanical elements for adjusting and/or controlling optics of the camera units 2701 (not shown).

30

35

Figure 10 also illustrates some operational elements which may be implemented, for example, as a computer code in the software of the processor, in a hardware, or both. A focus control element 2714 may perform operations related to adjustment of the optical system of camera unit or units to obtain focus meeting target specifications or

some other predetermined criteria. An optics adjustment element 2716 may perform movements of the optical system or one or more parts of it according to instructions provided by the focus control element 2714. It should be noted here that the actual adjustment of the optical system need not be performed by the apparatus, but it may be performed manually, wherein the focus control element 2714 may provide information for the user interface 2710 to indicate a user of the device how to adjust the optical system.

The various embodiments can be implemented with the help of computer program code that resides in a memory and causes the relevant apparatuses to carry out the method. For example, a device may comprise circuitry and electronics for handling, receiving and transmitting data, computer program code in a memory, and a processor that, when running the computer program code, causes the device to carry out the features of an embodiment. Yet further, a network device like a server may comprise circuitry and electronics for handling, receiving and transmitting data, computer program code in a memory, and a processor that, when running the computer program code, causes the network device to carry out the features of an embodiment. The computer program code comprises one or more operational characteristics. Said operational characteristics are being defined through configuration by said computer based on the type of said processor, wherein a system is connectable to said processor by a bus, wherein a programmable operational characteristic of the system comprises receiving a source picture; partitioning the source picture into a set of non-overlapping blocks; for a block, obtaining at least one sample/pixel value and its location from at least one neighboring block; deriving a prediction model for the block that relates the sample(s) of neighboring block(s) to the corresponding location(s) of the neighboring block(s); predicting a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area; deriving at least one other prediction model for at least one other prediction area using the predicted samples and their locations from the prediction area being previously predicted; and predicting a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least one other prediction area.

If desired, the different functions discussed herein may be performed in a different order and/or concurrently with other. Furthermore, if desired, one or more of the above-described functions and embodiments may be optional or may be combined.

Although various aspects of the embodiments are set out in the independent claims, other aspects comprise other combinations of features from the described embodiments and/or the dependent claims with the features of the independent claims, and not solely the combinations explicitly set out in the claims.

5

It is also noted herein that while the above describes example embodiments, these descriptions should not be viewed in a limiting sense. Rather, there are several variations and modifications, which may be made without departing from the scope of the present disclosure as, defined in the appended claims.

10

Claims:

1. A method, comprising:
 - receiving a source picture;
 - 5 - partitioning the source picture into a set of non-overlapping blocks;
 - for a block, obtaining at least one sample/pixel value and its location from at least one neighboring block;
 - deriving a prediction model for the block that relates the sample(s) of neighboring block(s) to the corresponding location(s) of the neighboring block(s) ;
 - 10 - predicting a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area;
 - deriving at least one other prediction model for at least one other prediction area using the predicted samples and their locations from the prediction area being previously predicted; and
 - 15 - predicting a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least one other prediction area.
2. The method according to claim 1, wherein the sample is a pixel or a sub-block.
- 20 3. The method according to claim 1 or 2, further comprising continuing the method for each block.
4. The method according to any of the claims 1 to 3, wherein the neighboring block is one of the following: a block on a top of a current block, a block on the left of a current block, a block on top-left of a current block, a block on bottom-left of a current block, a block on top-right of a current block.
- 25 5. The apparatus according to any of the claims 1 to 4, further comprising storing a derived prediction models for certain region in a picture.
- 30 6. The method according to any of the claims 1 to 5, further comprising deriving a prediction model by using samples and their locations of a certain sample component by samples of another sample component.
- 35 7. The method according to any of the claims 1 to 6, further comprising retrieving an existing prediction model from a storage and using the retrieved existing prediction model together with at the first or said at least one other derived prediction model.

8. The method according to any of the claims 1 to 7, wherein the model derivation is executed at an encoder and/or in a decoder.

9. An apparatus comprising:

- 5 - means for receiving a source picture;
- means for partitioning the source picture into a set of non-overlapping blocks;
- for a block, means for obtaining at least one sample/pixel value and its location from at least one neighboring block;
- means for deriving a prediction model for a block that relates the sample(s) of a neighboring block(s) to the corresponding location(s) of the neighboring block(s)
- 10 - means for predicting a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area;
- means for deriving at least one other prediction model for at least one other prediction area using the predicted samples and their locations from the prediction area being
- 15 previously predicted; and
- means for predicting a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least one other prediction area.

20 10. The apparatus according to claim 9, wherein the sample is a pixel or a sub-block.

11. The apparatus according to claim 9 or 10, further comprising means for continuing the method for each block.

25 12. The apparatus according to any of the claims 9 to 11, wherein the neighboring block is one of the following: a block on a top of a current block, a block on the left of a current block, a block on top-left of a current block, a block on bottom-left of a current block, a block on top-right of a current block.

30 13. The apparatus according to any of the claims 9 to 12, further comprising means for storing a derived prediction models for certain region in a picture.

14. The apparatus according to any of the claims 9 to 13, further comprising means for deriving a prediction model by using samples and their locations of a certain sample component by samples of another sample component.

35

15. The apparatus according to any of the claims 9 to 14, further comprising means for retrieving an existing prediction model from a storage and using the retrieved existing

prediction model together with at the first or said at least one other derived prediction model.

5 16. The apparatus according to any of the claims 9 to 15, wherein the model derivation is executed at an encoder and/or in a decoder.

17. An apparatus according to any of the claims 9 to 16, further comprising at least one processor and a memory including computer program code.

10 18. A computer program product comprising computer program code configured to, when executed on at least one processor, cause an apparatus or a system to

- receive a source picture;
- partition the source picture into a set of non-overlapping blocks;
- for a block, obtain at least one sample/pixel value and its location from at least one

15 neighboring block;

- derive a prediction model for the block that relates the sample(s) of neighboring block(s) to the corresponding location(s) of the neighboring block(s);
- predict a sample value of the sample of a first prediction area based on the derived prediction model and the sample location inside the first prediction area;

20 - derive at least one other prediction model for at least one other prediction area using the predicted samples and their locations from the prediction area being previously predicted; and

- predict a sample value of the sample of the at least one other prediction area based on said at least one other prediction model and the sample location inside said at least

25 one other prediction area.

19. A computer program product according to claim 18, wherein the computer program product is embodied on a non-transitory computer readable medium.

Fig. 1

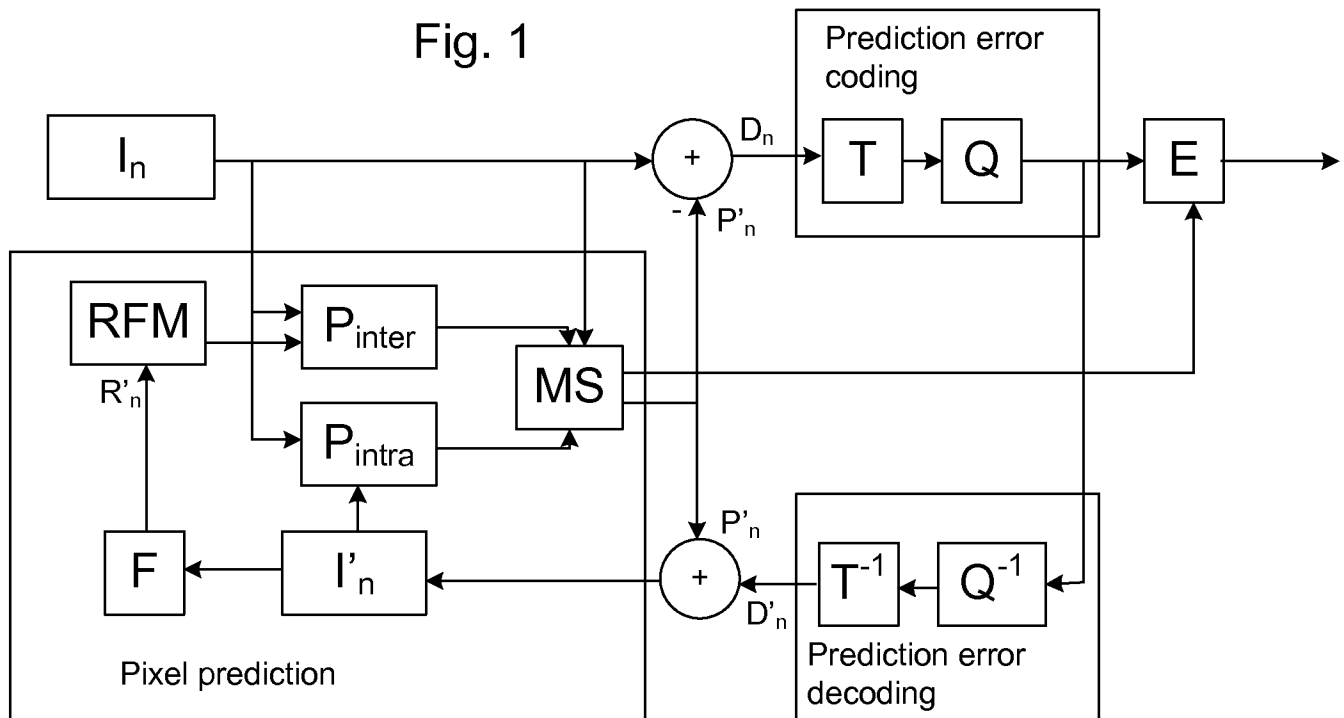
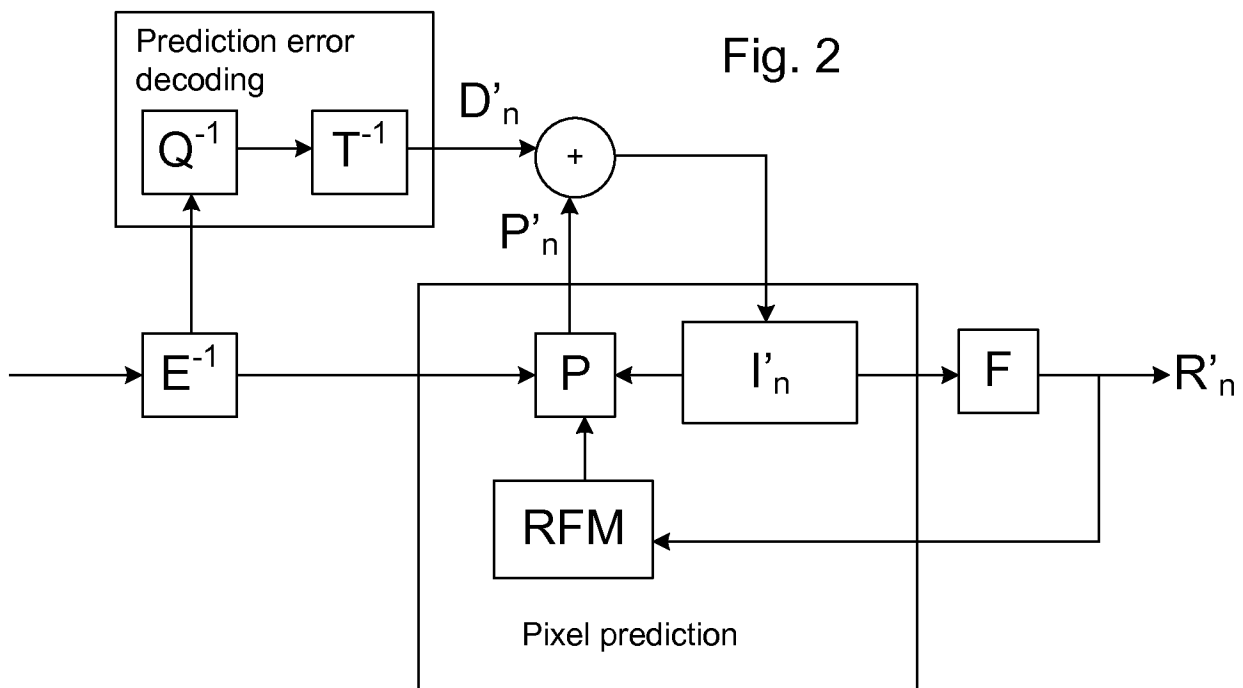


Fig. 2



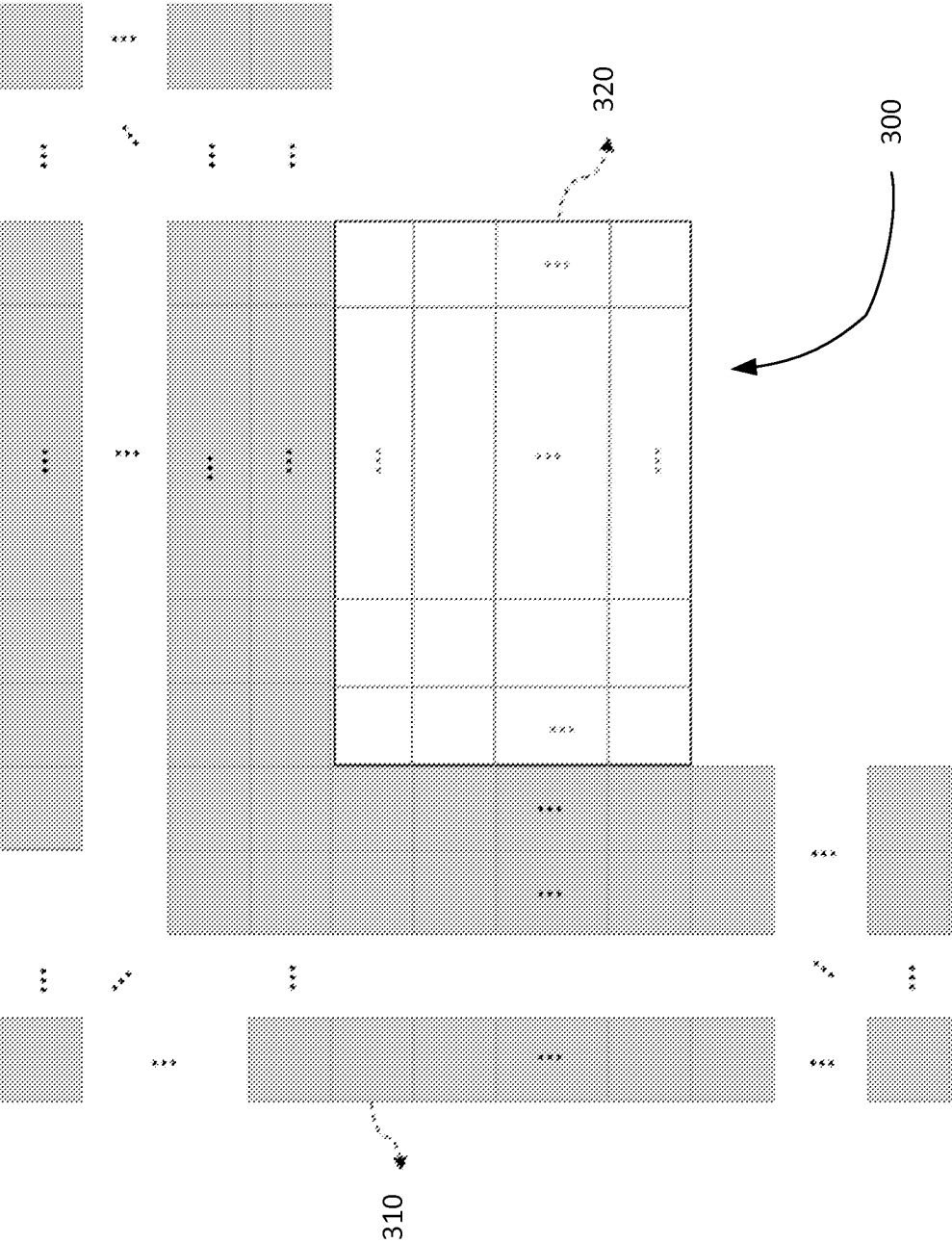


Fig. 3

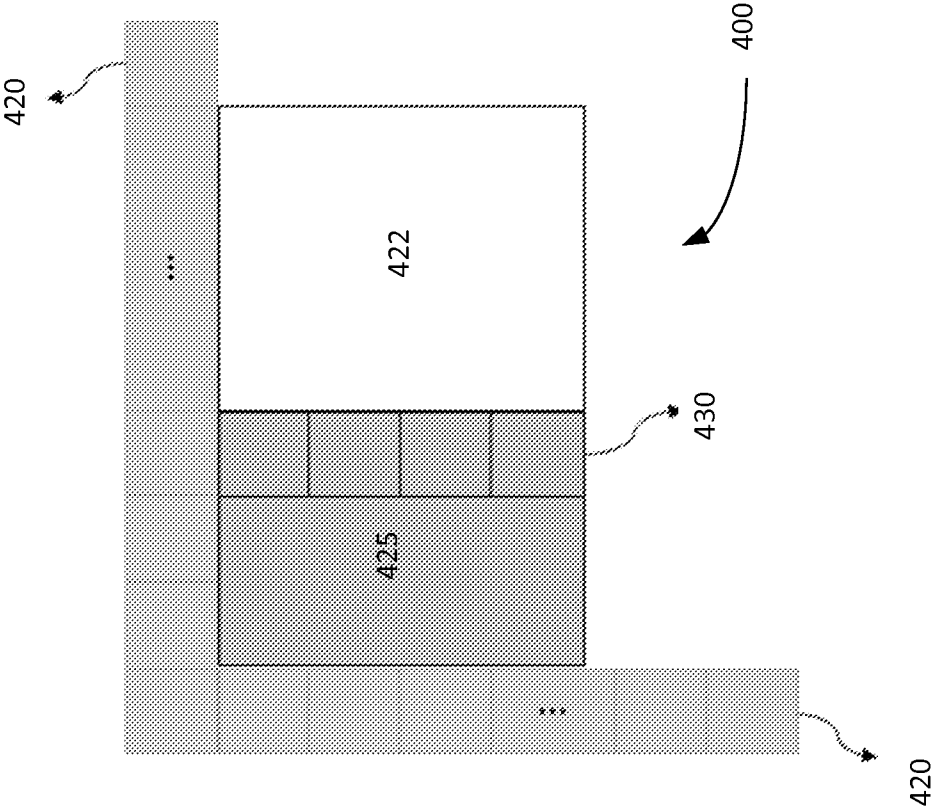


Fig. 4b

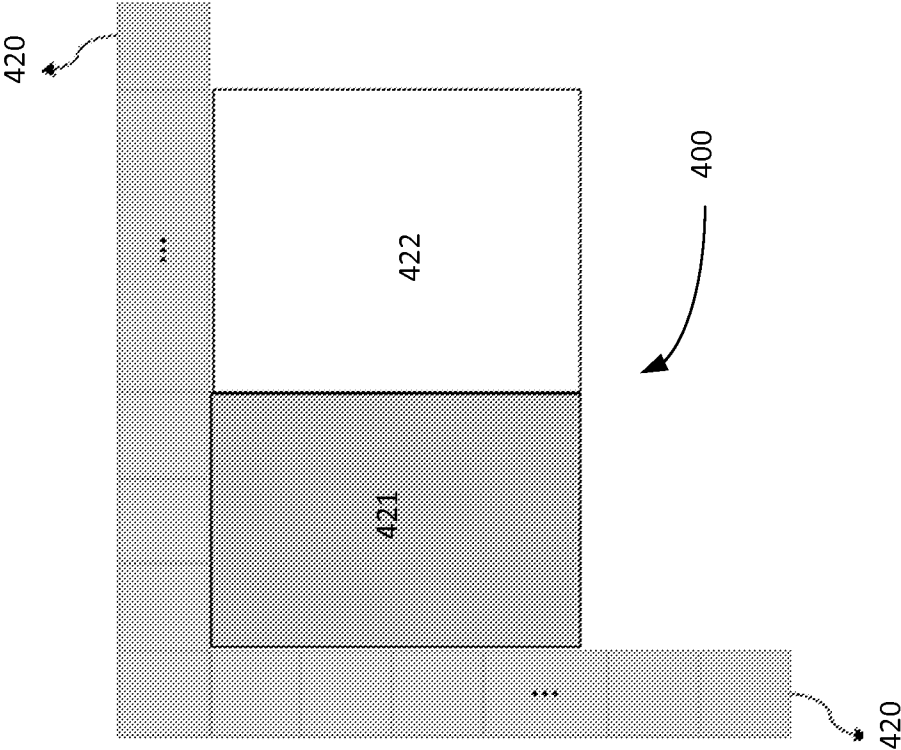


Fig. 4a

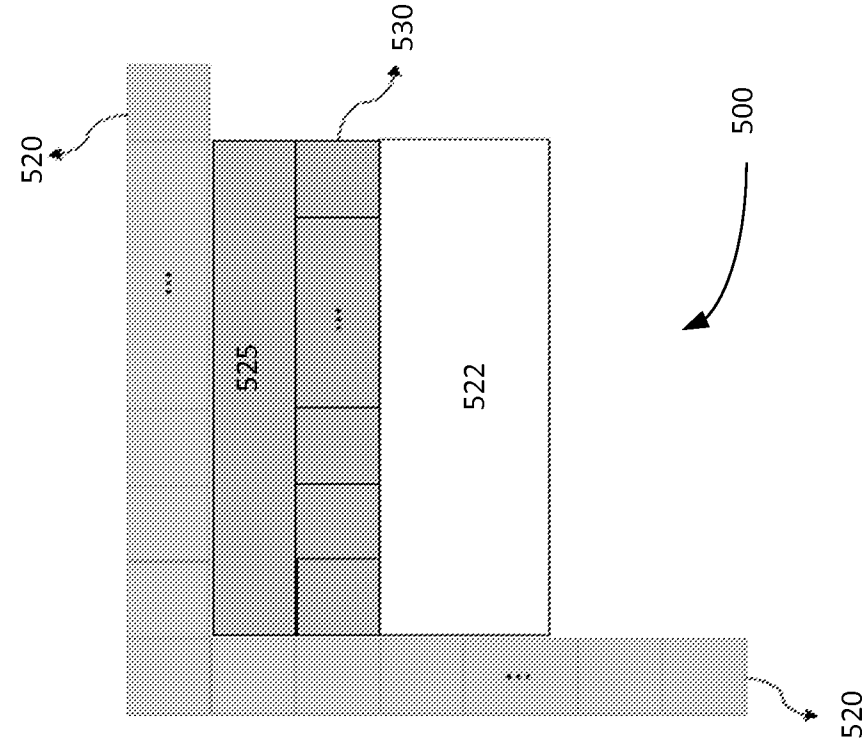


Fig. 5b

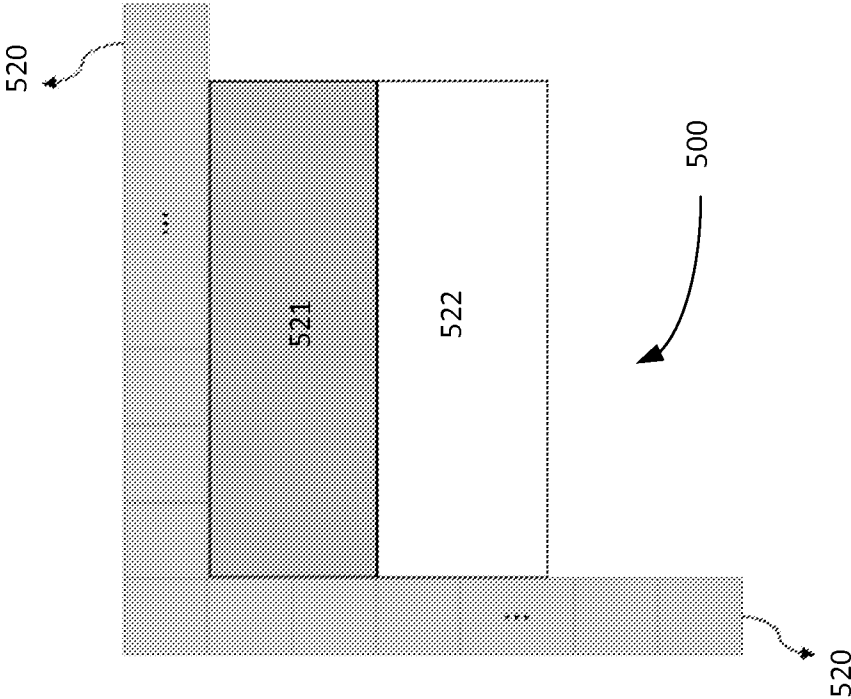
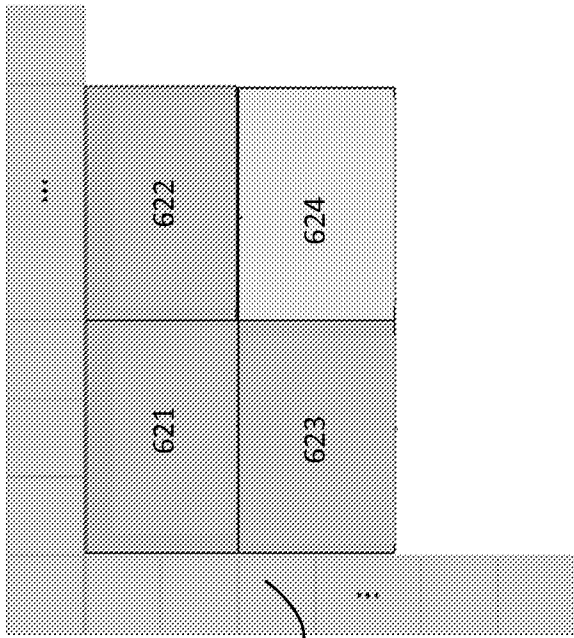
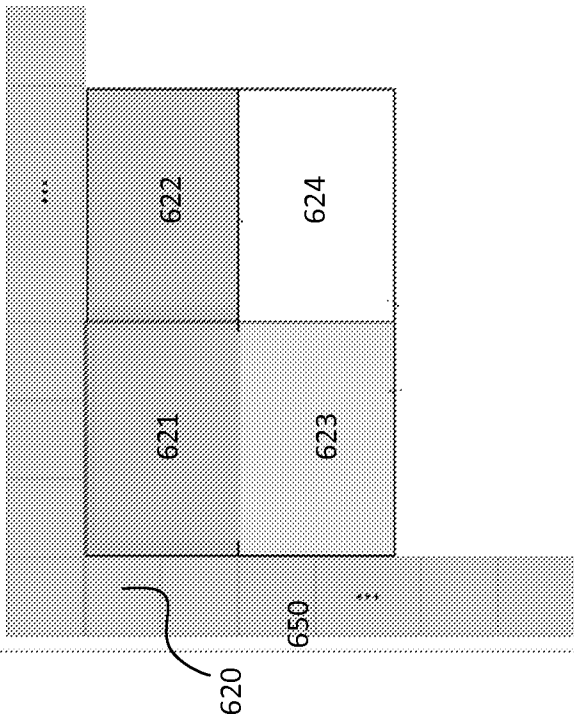
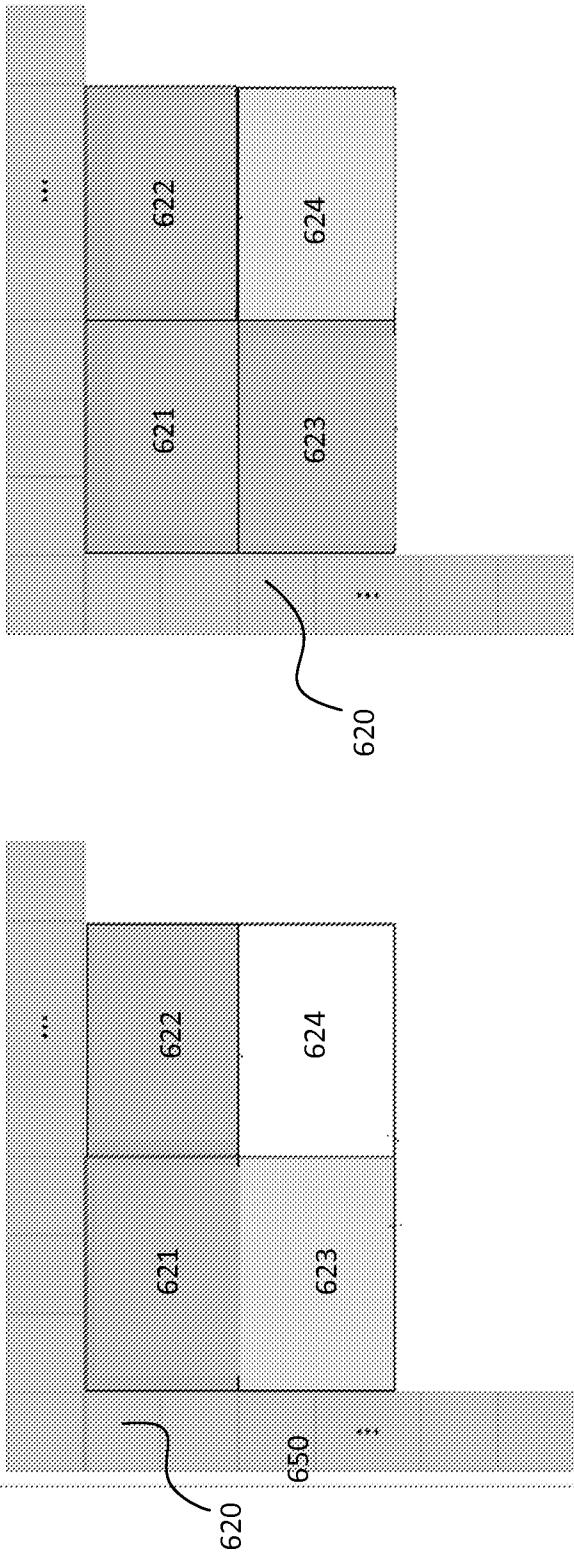
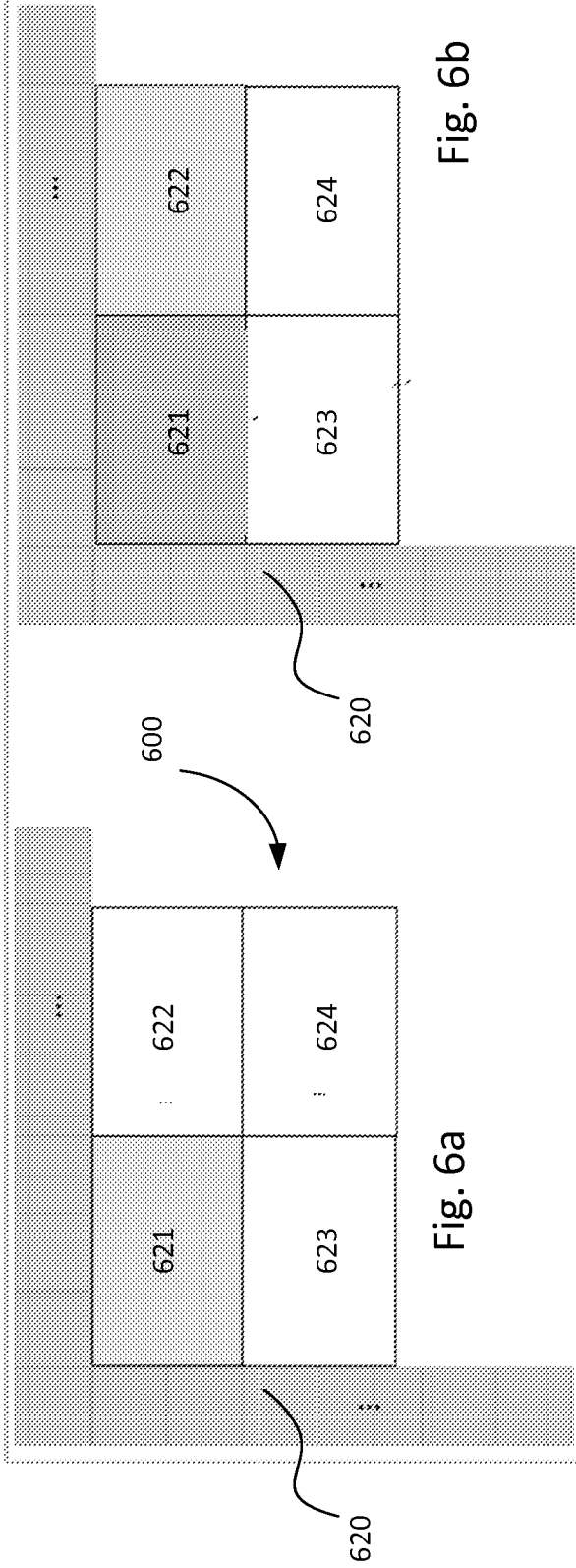
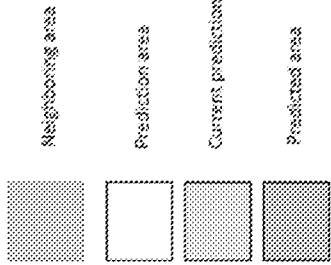


Fig. 5a



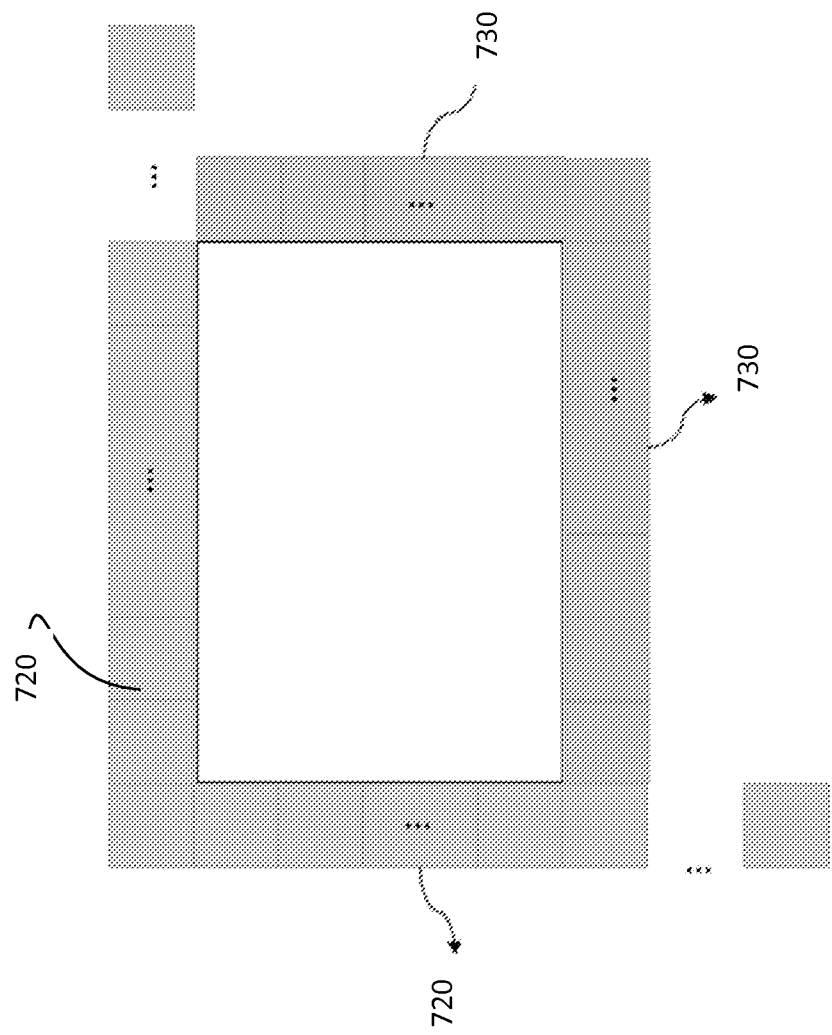


Fig. 7

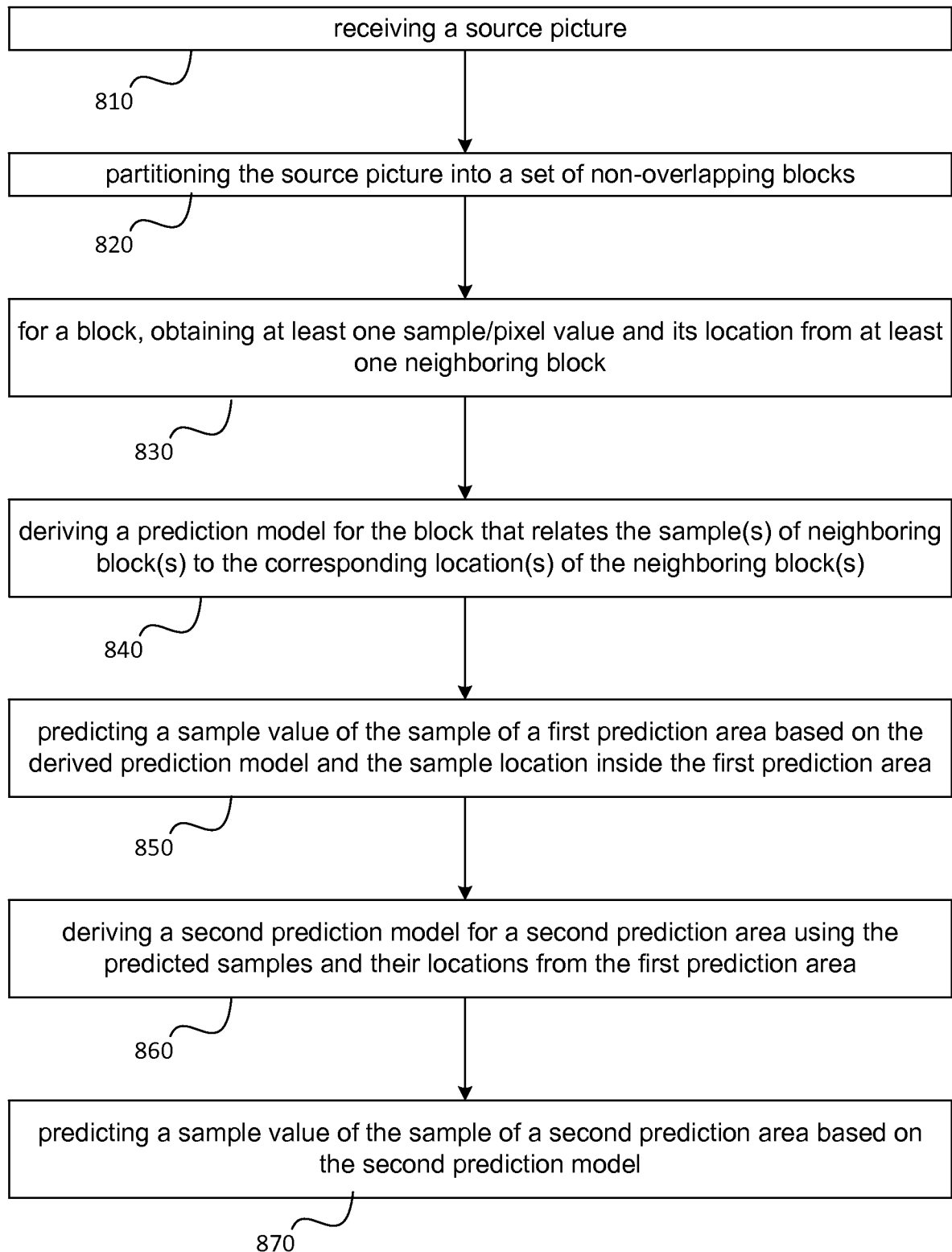


Fig. 8

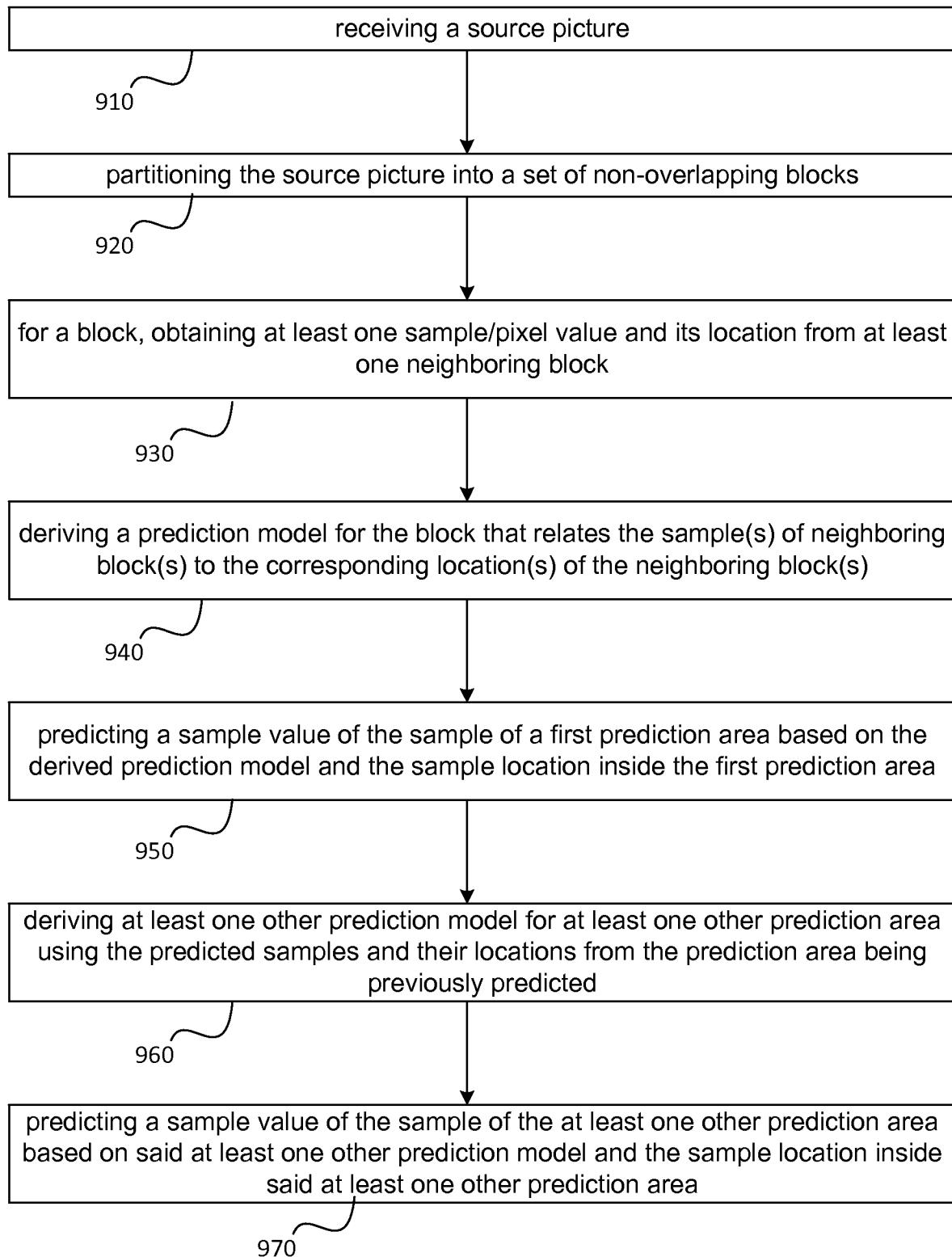


Fig. 9

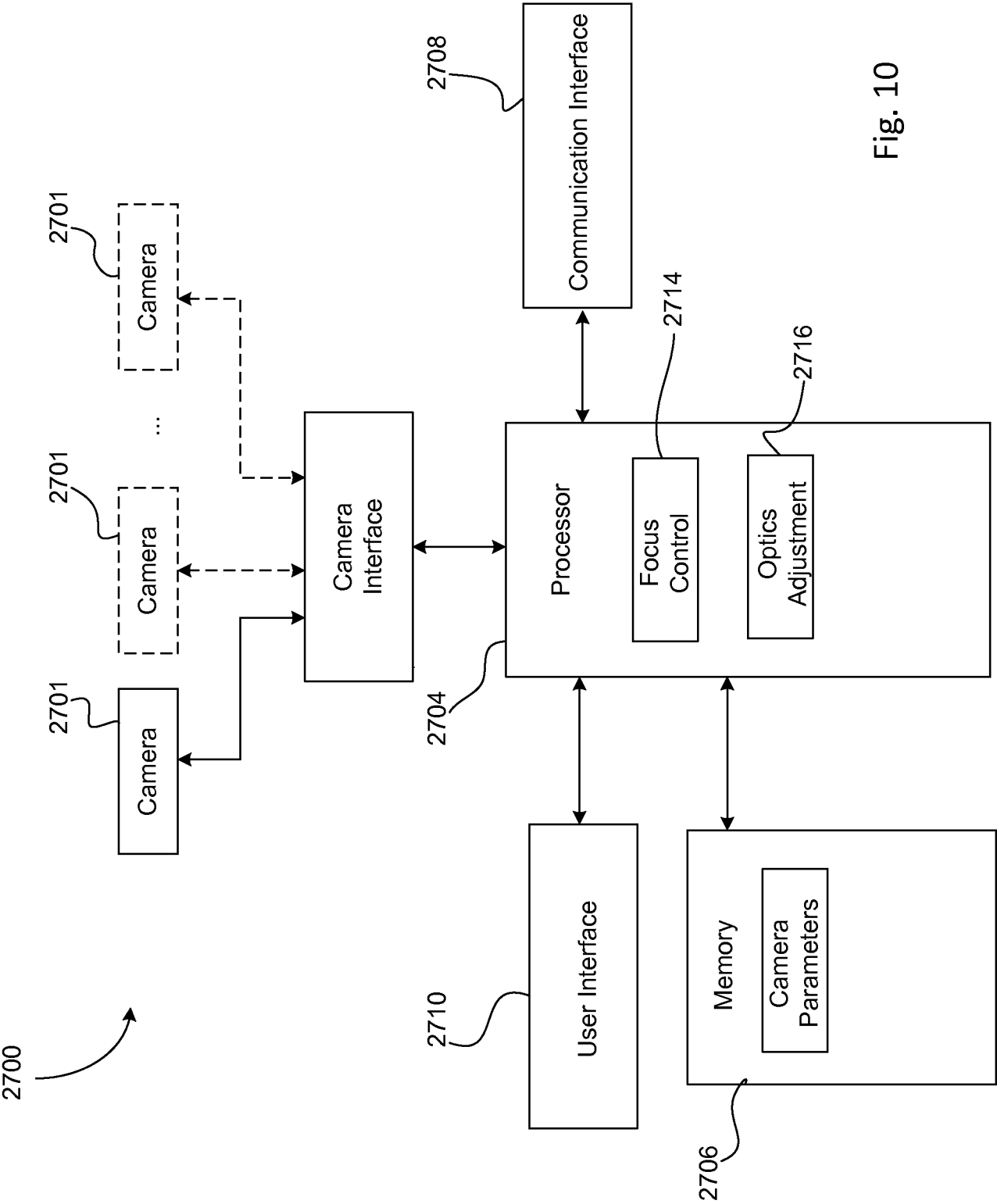


Fig. 10