



- (51) **International Patent Classification:**
H04N 7/32 (2006.01)
- (21) **International Application Number:**
PCT/US2012/036891
- (22) **International Filing Date:**
8 May 2012 (08.05.2012)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
13/115,638 25 May 2011 (25.05.2011) US
- (71) **Applicant** (for all designated States except US):
GOOGLE INC. [US/US]; 1600 Amphitheatre Parkway,
Mountain View, California 94043 (US).
- (72) **Inventors; and**
- (75) **Inventors/Applicants** (for US only): **WILKINS, Paul**
[GB/GB]; c/o Google Inc., 1600 Amphitheatre Parkway,
Mountain View, California 94043 (US). **BANKOSKI,**
James [US/US]; c/o Google Inc., 1600 Amphitheatre Park-
way, Mountain View, California 94043 (US). **XU, Yaowu**
[US/US]; c/o Google Inc., 1600 Amphitheatre Parkway,
Mountain View, California 94043 (US).
- (74) **Agents:** **BASILE, Andrew ,R, Jr.** et al.; Young Basile
Hanlon & MacFarlane, P.C., 3001 West Big Beaver Road,
Suite 624, Troy, Michigan 48083 (US).
- (81) **Designated States** (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ,
CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN,
HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR,
KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME,
MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ,
OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD,
SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR,
TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) **Designated States** (unless otherwise indicated, for every
kind of regional protection available): ARIPO (BW, GH,
GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, 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,
ML, MR, NE, SN, TD, TG).

[Continued on next page]

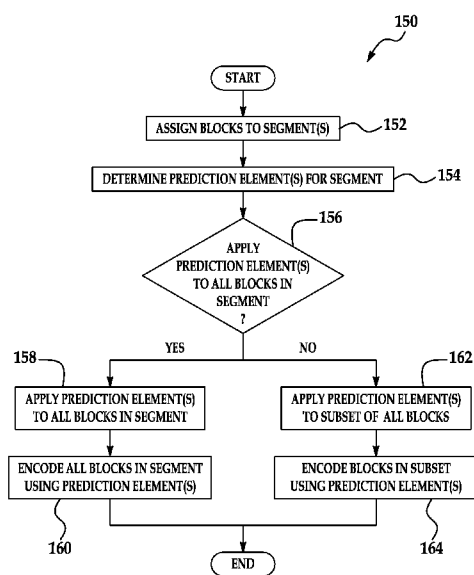
(54) **Title:** METHOD AND APPARATUS FOR USING SEGMENTATION-BASED CODING OF PREDICTION INFORMATION

FIG. 6

(57) **Abstract:** Disclosed herein is a method for encoding and decoding a video signal having at least one frame with a plurality of blocks. The encoding method includes assigning at least some of the plurality of blocks to a segment, determining at least one prediction element for the segment using a processor, applying the at least one prediction element to a first block and at least some of the other blocks in the segment and encoding the first block and the other blocks in the segment.



Published:

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

METHOD AND APPARATUS FOR USING SEGMENTATION-BASED CODING OF PREDICTION INFORMATION

TECHNICAL FIELD

[0001] The present invention relates in general to video encoding and decoding.

BACKGROUND

[0002] An increasing number of applications today make use of digital video for various purposes including, for example, remote business meetings via video conferencing, high definition video entertainment, video advertisements, and sharing of user-generated videos. As technology is evolving, users have higher expectations for video quality and expect high resolution video even when transmitted over communications channels having limited bandwidth.

[0003] To permit higher quality transmission of video while limiting bandwidth consumption, a number of video compression schemes are noted including formats such as VPx, promulgated by Google Inc. of Mountain View, California, and H.264, a standard promulgated by ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG), including present and future versions thereof. H.264 is also known as MPEG-4 Part 10 or MPEG-4 AVC (formally, ISO/IEC 14496-10).

[0004] These compression schemes can use different techniques to achieve compression. For example, many of these schemes use prediction techniques that predict where information contained in a portion of a first frame or first region of a frame can be found in a second frame or second region of the frame. The difference between the prediction and the portion of data in the second frame or second frame region is calculated to form a residual. One type of prediction, known as intra prediction, can be based on previously coded image samples within the current frame. Another type of prediction known as inter prediction can be based on previously coded frames ("reference frames"). One inter prediction technique, for example, can utilize block-based motion estimation and compensation. Motion estimation can involve searching for a sample region in a reference frame that, for example, closely matches a current block in a current frame. The luma and chroma samples of the matching sample region are subtracted from the current block to produce an error residual that is encoded. A motion vector is also

encoded that describes the position of the matching sample region relative to the position of the current block.

[0005] In some motion estimation search algorithms, there is a trade-off between the cost of coding prediction information needed to perform the prediction and the quality of the prediction (i.e. finding the matching region). In other words, finding the “best” prediction can come with the cost of an increased bitrate. Conversely, decreasing bitrate can result in not finding the most suitable prediction.

SUMMARY

[0006] Methods and apparatuses for encoding and decoding a video signal are disclosed herein.

[0007] In accordance with one aspect of the disclosed embodiments, a method for encoding a video signal having at least one frame with a plurality of blocks each having pixels includes assigning at least some of the plurality of blocks to a segment and determining at least one prediction element for the segment using a processor. The method also includes applying the at least one prediction element to a first block and at least some of the other blocks in the segment encoding the first block and the other blocks in the segment.

[0008] In accordance with another aspect of the disclosed embodiments, an apparatus for encoding a video signal having at least one frame with a plurality of blocks includes a memory and a processor configured to execute instructions stored in the memory to assign at least some of the plurality of blocks to a segment. The processor is also configured to execute instructions stored in the memory to determine at least one prediction element for the segment using a processor and apply at least one prediction element to a first block and at least some of the other blocks in the segment. Further, the process is configured to execute instructions stored in the memory to encode the first block and the other blocks in the segment

[0009] In accordance with yet another aspect of the disclosed embodiments, a method for decoding a video signal having at least one frame with a plurality of blocks includes assigning at least some of the plurality of blocks to a segment and obtaining at least one prediction element for the segment using a processor. The method also includes applying the at least one prediction element to a first block and at least some of the other blocks in the segment and decoding the first block using the at least one prediction element.

[0010] These and other embodiments will be described in additional detail hereinafter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] The description herein makes reference to the accompanying drawings wherein like reference numerals refer to like parts throughout the several views, and wherein:

[0012] FIG. 1 is a diagram of an encoder and decoder system in accordance with one embodiment;

[0013] FIG. 2 is a diagram of a video bitstream that can be encoded and decoded by the system of FIG. 1;

[0014] FIG. 3 is a block diagram of an exemplary encoder implemented in the system of FIG. 1;

[0015] FIG. 4 is a block diagram of an exemplary decoder implemented in the system of FIG. 1;

[0016] FIG. 5 is a schematic diagram of intra-prediction and inter-prediction modes used in the encoder and decoder of FIGS. 3 and 4;

[0017] FIG. 6 is flow chart diagram of an exemplary process of prediction for use by the encoder and decoder of FIGS. 3 and 4;

[0018] FIG. 7A is an exemplary schematic diagram of motion vector designations for blocks in a frame for processing by the motion estimation process of FIG. 5; and

[0019] FIG. 7B is an exemplary schematic diagram of the blocks of FIG. 6 assigned to groups.

DETAILED DESCRIPTION

[0020] FIG. 1 is a diagram of an encoder and decoder system 10 for still or dynamic video images. An exemplary transmitting station 12 can be, for example, a computer having an internal configuration of hardware including a processor such as a central processing unit (CPU) 14 and a memory 16. CPU 14 can be a controller for controlling the operations of transmitting station 12. The CPU 14 is connected to memory 16 by, for example, a memory bus. Memory 16 can be random access memory (RAM) or any other suitable memory device. Memory 16 can store data and program instructions which are used by the CPU 14. Other suitable implementations of transmitting station 12 are possible.

[0021] A network 28 connects transmitting station 12 and a receiving station 30 for encoding and decoding of the video stream. Specifically, the video stream can be encoded by an encoder in transmitting station 12 and the encoded video stream can be decoded by a decoder in receiving station 30. Network 28 can, for example, be the Internet. Network 28 can also be a local area network (LAN), wide area network (WAN), virtual private network (VPN), or any other means of transferring the video stream from transmitting station 12.

[0022] Receiving station 30, in one example, can be a computer having an internal configuration of hardware include a processor such as a central processing unit (CPU) 32 and a memory 34. CPU 32 is a controller for controlling the operations of receiving station 30. CPU 32 can be connected to memory 34 by, for example, a memory bus. Memory 34 can be RAM or any other suitable memory device. Memory 34 stores data and program instructions which are used by CPU 32. Other suitable implementations of receiving station 30 are possible.

[0023] A display 36 configured to display a video stream can be connected to receiving station 30. Display 36 can be implemented in various ways, including by a liquid crystal display (LCD) or a cathode-ray tube (CRT). The display 36 can be configured to display a video stream 38 decoded by the decoder in receiving station 30.

[0024] Other implementations of the encoder and decoder system 10 are possible. For example, one implementation can omit the network 28 and/or the display 36. In another implementation, a video stream can be encoded and then stored for transmission at a later time by receiving station 12 or any other device having memory. In another implementation, additional components can be added to the encoder and decoder system 10. For example, a display or a video camera can be attached to transmitting station 12 to capture the video stream to be encoded.

[0025] FIG. 2 is a diagram a typical video stream 50 to be encoded and decoded. Video coding formats, such as VP8 or H.264, provide a defined hierarchy of layers for video stream 50. Video stream 50 includes a video sequence 52. At the next level, video sequence 52 consists of a number of adjacent frames 54, which can then be further subdivided into a single frame 56. At the next level, frame 56 can be divided into a series of blocks 58 (e.g. blocks), which can contain data corresponding to, for example, a 16x16 block of displayed pixels in frame 56. Each block can contain luminance and chrominance data for the corresponding pixels. Blocks 58 can also be of any other suitable size such as 16x8 pixel groups or 8x16 pixel groups.

[0026] FIG. 3 is a block diagram of an encoder 70 in accordance with one embodiment. In one embodiment, encoder 70 can be implemented, as described previously, in transmitting station 12. Encoder 70 encodes an input video stream 50. Encoder 70 has the following stages to perform the various functions in a forward path (shown by the solid connection lines) to produce an encoded or a compressed bitstream 88: an intra/inter prediction stage 72, a transform stage 74, a quantization stage 76 and an entropy encoding stage 78. Encoder 70 also includes a reconstruction path (shown by the dotted connection lines) to reconstruct a frame for encoding of further blocks. Encoder 70 has the following stages to perform the various functions in the reconstruction path: a dequantization stage 80, an inverse transform stage 82, a reconstruction stage 84 and a loop filtering stage 86. Other structural variations of encoder 70 can be used to encode input video stream 50.

[0027] When input video stream 50 is presented for encoding, each frame 56 within input video stream 50 is processed in units of blocks. At intra/inter prediction stage 72, each block can be encoded using either intra-frame prediction (i.e., within a single frame) or inter-frame prediction (i.e. from frame to frame). In either case, a prediction block can be formed. In the case of intra-prediction, a prediction block can be formed from samples in the current frame that have been previously encoded and reconstructed. In the case of inter-prediction, a prediction block can be formed from samples in one or more previously constructed reference frames.

[0028] Next, still referring to FIG. 3, the prediction block can be subtracted from the current block at stage 72 to produce a residual block (residual). Transform stage 74 transforms the residual into transform coefficients in, for example, the frequency domain. Examples of block-based transforms include the Karhunen-Loève Transform (KLT), the Discrete Cosine Transform (“DCT”) and the Singular Value Decomposition Transform (“SVD”). In one example, the DCT transforms the block into the frequency domain. In the case of DCT, the transform coefficient values are based on spatial frequency, with the lowest frequency (i.e. DC) coefficient at the top-left of the matrix and the highest frequency coefficient at the bottom-right of the matrix.

[0029] Quantization stage 76 converts the transform coefficients into discrete quantum values, which are referred to as quantized transform coefficients. The quantized transform coefficients are then entropy encoded by entropy encoding stage 78. The entropy-encoded coefficients, together with the information required to decode the block, such as the type of prediction used, motion vectors and quantizer value, are then output to

compressed bitstream 88. The compressed bitstream 88 can be formatted using various techniques, such as run-length encoding (RLE) and zero-run coding.

[0030] The reconstruction path in FIG. 3 is present to ensure that both encoder 70 and a decoder 100 (described below) use the same reference frames to decode compressed bitstream 88. The reconstruction path performs functions that are similar to functions that take place during the decoding process that are discussed in more detail below, including dequantizing the quantized transform coefficients at dequantization stage 80 and inverse transforming the dequantized transform coefficients at an inverse transform stage 82 in order to produce a derivative residual block (derivative residual). At reconstruction stage 84, the prediction block that was predicted at intra/inter prediction stage 72 can be added to the derivative residual to create a reconstructed block. A loop filter 86 can then be applied to the reconstructed block to reduce distortion such as blocking artifacts.

[0031] Other variations of encoder 70 can be used to encode compressed bitstream 88. For example, a non-transform based encoder can quantize the residual signal directly without transform stage 74. In another embodiment, an encoder can have quantization stage 76 and dequantization stage 80 combined into a single stage. The operation of encoding can be performed in many different ways and can produce a variety of encoded data formats. The above-described embodiments of encoding can illustrate some exemplary encoding techniques. However, in general, encoding is understood to mean any transformation of data from one form to another that may or may not include compression, reversibility or loss of data.

[0032] FIG. 4 is a block diagram of a decoder 100 to decode compressed bitstream 88. Decoder 100 can be implemented, as described previously, in receiving station 30. Encoder 70 encodes an input video stream 50. Decoder 100, similar to the reconstruction path of the encoder 70 discussed previously, includes the following stages to perform various functions to produce an output video stream 116 from compressed bitstream 88: an entropy decoding stage 102, a dequantization stage 104, an inverse transform stage 106, an intra/inter prediction stage 108, a reconstruction stage 110, a loop filter stage 112 and a deblocking filtering stage 114. Other structural variations of decoder 100 can be used to decode compressed bitstream 88.

[0033] When compressed bitstream 88 is presented for decoding, the data elements within compressed bitstream 88 can be decoded by entropy decoding stage 102 (using, for example, Context Adaptive Binary Arithmetic Decoding) to produce a set of

quantized transform coefficients. Dequantization stage 104 dequantizes the quantized transform coefficients, and inverse transform stage 106 inverse transforms the dequantized transform coefficients to produce a derivative residual that can be identical to that created by the reconstruction stage in the encoder 70. Using header information decoded from the compressed bitstream 88, decoder 100 can use intra/inter prediction stage 108 to create the same prediction block as was created in encoder 70. At the reconstruction stage 110, the prediction block can be added to the derivative residual to create a reconstructed block. The loop filter 112 can be applied to the reconstructed block to reduce blocking artifacts. Deblocking filter 114 can be applied to the reconstructed block to reduce blocking distortion, and the result is output as output video stream 116.

[0034] Other variations of decoder 100 can be used to decode compressed bitstream 88. For example, a decoder can produce output video stream 116 without deblocking filtering stage 114.

[0035] FIG. 5 is a schematic diagram of intra prediction and inter prediction. As illustrated, FIG. 5 shows reference frames 130, 132 and a current frame 134 that is currently being encoded or decoded. As discussed previously, each frame can be processed in units of blocks at intra/inter prediction stage 72 or intra/inter prediction stage 108, and each block can be coded using either intra prediction, inter prediction or some combination of inter prediction and intra prediction. For example, a current block 136 is shown being encoded or decoded using inter prediction from a block 138 from previously coded reference frame 130. Similarly, a current block 140 is shown being encoded or decoded using inter prediction from a block 142 from previously coded reference frame 132. Also, for example, a current block 144 is shown being encoded or decoded using intra prediction from a block 146 within current frame 134.

[0036] Inter prediction can utilize block-based motion estimation to compensate for movement of blocks each having, for example, $M \times N$ samples (e.g., 16×16) in the current frame. To predict the position of a current block (e.g. current block 140) in a current frame (e.g., frame 134), an area can be searched in a reference frame (e.g., previously coded frame 132) to find a best-matching block. The searched area or search window can be a region in the reference frame that is centered about the same coordinates as the current block in the current frame that is extended by a maximum displacement R . As such, the search window can have an area of $(2R + M) \times (2R + N)$. Other search areas/windows are possible. For example, the search window can be centered around the best-matching block found in a previous search.

[0037] Block-based motion estimation is the process of using a search scheme to find the best-matching block in the search window on the basis of a matching criterion. In some instances, the matching criterion is a measure of error between a block in the search window and the current block, and the best matching block is a block in the search window that has the lowest measure of error. For example, the measure of error can be the mean squared error, the mean absolute difference or a normalized cross-correlation function between the current block and the search window block. Other matching criteria are also possible.

[0038] The displacement between the current block and the best-matching block is saved as a motion vector for the current block. Subsequently, the best-matching block (i.e. the predictor) is subtracted from the current block to form the residual block. As discussed previously, both the residual block and the motion vector can be further processed and compressed (e.g., through transformation, quantization, and entropy encoding).

[0039] Both accuracy (i.e., prediction quality) and bitrate are important characteristics of motion estimation and compensation. Spending more bits on the prediction signal by, for example, increasing the number or precision of motion vectors, can result in a reduction in the size (i.e., number of bits) of the residual block. Hence, in some motion estimation and compensation algorithms, there is a trade-off between the quality of the prediction and the bitrate. In other words, finding the “best” matching block can come with a cost of additional bits. Conversely, decreasing the number of bits used to represent the prediction signal can result in not finding the “best” matching block.

[0040] FIG. 6 is an exemplary prediction process 150 for use by, for example, encoder 70 or decoder 100. Beginning at step 152, one or more blocks in a current frame (e.g., frame 134) are assigned to a segment. Blocks in the current frame can be assigned to the same segment based on one or more factors such as motion information, color information or spatial information. Other parameters are also available. Further, frames can have more than one segment. The layout of the segments and the allocation of blocks to various segments, or the segmentation map, can vary, for example, from frame to frame throughout the encoding or decoding process. In other embodiments, the segmentation map can persist for each frame, be updated periodically, or be updated based on some other factor such as a rate-distortion metric.

[0041] The segmentation can be explicitly coded in the compressed bitstream 88 or the allocation of blocks can be detected by decoder 100 implicitly, based upon other

information carried in compressed bitstream 88 or determined algorithmically through analysis of the content or encoding choices in one or more previously encoded frames, for example.

[0042] As one example, when motion information is the parameter used for determining the segmentation map, neighboring blocks having similar motion vectors are identified and are assigned to one or more segments. Through this process, all neighboring blocks having similar motion vectors are assigned to the same segment. Neighboring blocks can include adjacent blocks and/or non-adjacent blocks. For example, a neighboring block can be a block located within a predetermined number of blocks from a current block. The predetermined number of blocks can be any suitable number, such as within three blocks from another block.

[0043] Similar motion vectors also include motion vectors that are the same. In one implementation, the motion vectors for two neighboring blocks are considered to be similar when the distance between the motion vectors is less than some threshold distance. Other similarity measures are also possible. In some instances, blocks having unique motion vectors will be assigned to their own segment or may not be assigned to any segment.

[0044] FIG. 7A is an exemplary schematic diagram showing the assignment of motion vectors for blocks B0-B15 in a frame 200. To ease the reader's understanding, frame 200 has been simplified to include only 16 blocks, though any number of blocks can be included. As illustrated, block B0 contains a new motion vector (MV). Blocks B1-B3, B4, B7, B8, B11 and B15 all have motion vectors that are the same or similar to the motion vector of block B0, and are therefore assigned the motion vector from block B0. Similarly, block B5 contains a new motion vector, and blocks B6, B9 and B10 have motion vectors that are the same or similar to the motion vector of block B5, and are therefore assigned the motion vector from block B5. Each of blocks B12, B13 and B14 all have new motion vectors that are not used by any other blocks.

[0045] FIG. 7B is an exemplary schematic diagram showing the assignment of blocks B0-B15 of FIG. 7A to segments or a segmentation map 202. As illustrated, frame 200 includes 3 distinct segments. Blocks B0-B3, B4, B7, B8, B11 and B15, which share the motion vector for block B0, are assigned to segment 1. Blocks B5, B6, B9 and B10, which share the motion vector for block B5, are assigned to segment 2. Blocks B12, block B13 and block B14 are not assigned to any segment. Each of these blocks has a unique motion vector that is not shared by any other blocks in frame 200.

[0046] As another example, when color information is the parameter used for determining the segmentation map, neighboring blocks having similar color values can be identified and assigned to one or more segments. Similar to the example discussed previously with respect to motion information, all neighboring blocks having similar color values are assigned to the same segment. In one implementation, the color values for two neighboring blocks are considered to be similar when the difference between, for example, the Y, U and V components of the pixels is less than some threshold value. Threshold value(s) for each of the Y, U and V components can be the same or different. Further, the difference can be determined based on the average value of the color components in each block. In other embodiments, less than all of the components can be considered (e.g., only the Y component). Alternatively other color representation schemes (e.g., RGB) can be used.

[0047] As another example, when spatial information is the parameter used for determining the segmentation map, neighboring blocks having a similar spatial location can be identified and assigned to one or more segments. As one example, segments based on blocks having a similar spatial location can be a foreground segment and a background segment.

[0048] Apart from or in addition to the location, spatial information can also include the texture characteristics of the blocks in the video stream. Blocks having similar texture characteristics (e.g., uniform or non-uniform texture) can be assigned to the same segment. Spatial information can also include signal characteristics other than the location or texture information of a block.

[0049] Although, as discussed previously, there can be more than one segment in the current frame, the following description will explain process 150 with reference to only one segment. It is to be understood, however, that the steps of process 150 can equally apply to other segments (if any) in the current frame.

[0050] Once blocks have been assigned to a segment, control moves to step 154 to determine one or more prediction elements for the segment. A prediction element can be any information associated with motion estimation, motion compensation or other techniques of creating a prediction for the current block (e.g., intra prediction). For example, the prediction element can be a prediction mode, a reference frame or a motion vector. Other prediction elements are possible. For example, in some implementations, prediction elements can include an indication of some type of transformation (e.g., global

zoom, warp, rotation, brightness change) that should be associated and applied at the segment level.

[0051] In some instances, one segment can have a set of prediction elements and another segment can have another set of prediction elements. For example, returning to FIG. 7B, segment 1 can have reference frame and a motion vector for prediction elements and segment 2 can only have a prediction mode for a prediction element. The prediction elements associated with a certain segment (e.g., segment 1) do not necessarily have to persist from frame to frame. For example, in one frame, segment 1 can have reference frame and a motion vector for prediction elements, and in the next frame, segment 1 can only have the reference frame as a prediction element.

[0052] Further, which prediction elements are associated with a particular segment can vary, for example, from frame to frame throughout the encoding or decoding process. In other embodiments, the prediction elements associated with a particular segment can persist for each frame, be updated periodically, or be updated based on some other factor such as a rate-distortion metric.

[0053] Each prediction element can also have one or more possible values. For example, the prediction mode can be, for example, intra prediction or inter prediction. Further, intra prediction mode can also contain an associated coding mode. Coding modes relay how intra predictions are formed for a given block. Thus, for example, intra predictions can be formulated by copying samples (or filtered samples) from adjacent, previously coded blocks to predict the values in the current block. For example, a chroma component can be intra predicted using one of four prediction modes such as vertical prediction, horizontal prediction, DC prediction or True Motion prediction. Similarly, a luma component can be predicted using vertical prediction, horizontal prediction, DC prediction or True Motion prediction. Other prediction modes are also available (e.g., southwest prediction, southeast prediction, vertical right diagonal prediction, vertical left diagonal prediction, horizontal down prediction, horizontal up prediction, etc.). Rather than copying pixel values, another intra prediction coding mode uses one or more parameterized equations. These parameterized equations can, for example, be an expression representing a curve that has a “best fit” to a defined set of previously coded pixel values.

[0054] The reference frame (in the case of inter prediction) can be, for example, last frame, golden frame, alternate frame or constructed frame. The motion vector (again, in the case of inter prediction) can be any two-dimensional vector.

[0055] The value of the prediction element can be selected for the segment in a variety of different ways. The prediction element can be based on prediction information for one or more blocks in a segment. Prediction information for a block can include a prediction mode (e.g., inter prediction or intra prediction), a reference frame or a motion vector or any other information associated with motion estimation or motion compensation or any other prediction technique.

[0056] In one implementation, the prediction element can be selected based on the value of prediction information that is most common among the blocks in that segment. For example, returning to FIG. 7B, if blocks B0-B3, B4, B7, B8 and B11 each have the last frame as their designated reference frame and block B15 has the golden frame as its designated reference frame, then the last frame can be selected as the value for the reference frame prediction element since it is the most common among the blocks in that segment.

[0057] In another implementation, the prediction element can be selected based on the value of a certain block's prediction information within that segment. For example, if the prediction element is reference frame and block B2 has the last frame as its reference frame, the last frame can be selected as the value of the reference frame prediction element for segment 1. The selection of the block can be random, predefined in the encoder or based on some other factor.

[0058] In another implementation, the value of the prediction element can be selected based on the average value of one or more blocks' prediction information in the segment. For example, if the prediction element is motion vector and block B0 has a motion vector of (0, 0) and block B1 has a motion vector of (4, 4), an average value of the two motion vectors, (2, 2) can be selected as the value of the motion vector prediction element for segment 1. The average can be taken from all of the blocks in the segment or a subset of all the blocks in the segment.

[0059] The value of the prediction element can be an initial value or an absolute value. In the case of an initial value, each individual block within the segment can be associated with a change value. The change value can be an incremental value, a decremental value or any other value that can be combined with the initial value to amount to an absolute value. For example, if the initial value of a motion vector associated at the segment level is (1, 0), an individual block can have an incremental value of (1, 1). Thus, the absolute value of the motion vector for the individual block will

be (2, 1). In other implementations, the absolute value, rather than the initial value, is what is associated at the segment level.

[0060] Further, the values of the prediction element(s) associated with a particular segment can vary, for example, from frame to frame throughout the encoding or decoding process. In other embodiments, the values of prediction element(s) associated with a particular segment can persist for each frame, be updated periodically, or be updated based on some other factor such as a rate-distortion metric.

[0061] The foregoing implementations describing selection of values for the prediction element are merely examples. Other techniques of selecting values are also possible.

[0062] Once the one or more prediction elements are determined at step 154, control moves to decision step 156 to determine whether to apply the prediction element(s) to all blocks in the segment. If the prediction element(s) are to be applied to all blocks, control moves to step 158 to apply the prediction element(s) to all blocks in the segment and moves to step 160 to encode the blocks in the segment using the prediction element(s).

[0063] As one example, if the prediction element for a certain segment is reference frame with a value of “last”, all blocks in that segment are processed and encoded with the last frame as the reference frame. In some instances, certain blocks can initially have a different value for a reference frame (i.e., golden) that can be superseded with the value of the “last” reference frame.

[0064] Otherwise, if the prediction element(s) are not to be applied to all blocks, control moves to step 162 to apply the prediction element(s) to a subset of all blocks in the segment. A subset can be any number of blocks less than the total number of blocks in the segment. Control then moves to step 164 to encode blocks in the subset using the prediction element(s).

[0065] In other words, blocks that are not included in the subset can be considered as having “opted-out” of the values specified for the prediction element(s). Accordingly, a block that has opted-out can have an individual value associated with the block. For example, referring to FIG. 7B, segment 1 can specify a reference frame prediction element to be the “last” frame and the motion vector to be (0,0), but a specific individual block, such as block B1 can specify the golden frame as its reference frame and still use the (0,0) motion vector specified at the segment level. In other instances, block B1 can

also have a different motion vector than (0, 0) in addition to using a different reference frame.

[0066] During encoding, either at step 160 or step 164, the prediction element(s) can be associated with header information for the segment. The header information can contain information particular to the segment, including a unique id identifying the segment, information about the duration (e.g., time or number of frames) the segment persists over multiple frames and/or information how this segment relates to other segments. Other segment data can also be included in the header information for the segment. This header information can be used by the decoder to properly identify, process and decode the segment.

[0067] Utilizing more bits to represent prediction information by, for example, increasing the number or precision of motion vectors, can result in reduction in size of the residual block and can contribute to enhanced quality. Conversely, utilizing fewer bits to represent prediction information can increase the size of the residual block and can contribute to poorer quality. Consequently, there is a tradeoff between creating the best overall visual quality and having a low bitrate. Associating prediction elements and their values at the segment level can achieve an improved bitrate because less prediction information (or in some cases, no prediction information) will be presented at the individual block level. Further, using segmentation maps to assign blocks related in some manner (e.g., similar motion information, color information, etc.) can assist in improving visual quality notwithstanding the reduced bitrate. That is, information that is needed for obtaining improved visual quality can be coded more efficiently using the embodiments described herein. The embodiments, as discussed previously, can provide a mechanism by which individual blocks that are spatially and temporally related can share information (i.e., the prediction elements) in order to improve bitrate whilst preserving visual quality.

[0068] The operation of encoding can be performed in many different ways and can produce a variety of encoded data formats. The above-described embodiments of encoding or decoding may illustrate some exemplary encoding techniques. However, in general, encoding and decoding are understood to include any transformation or any other change of data whatsoever.

[0069] The embodiments of transmitting station 12 and/or receiving station 30 (and the algorithms, methods, instructions etc. stored thereon and/or executed thereby) can be realized in hardware, software, or any combination thereof including, for example, IP cores, ASICs, programmable logic arrays, optical processors, molecular processors,

quantum processors, programmable logic controllers, microcode, firmware, microcontrollers, servers, microprocessors, digital signal processors or any other suitable circuit or other information processing device now existing or hereafter developed. In the claims, the term “processor” should be understood as encompassing any the foregoing, either singly or in combination. The terms “signal” and “data” are used interchangeably. Further, portions of transmitting station 12 and receiving station 30 do not necessarily have to be implemented in the same manner.

[0070] Further, in one embodiment, for example, transmitting station 12 or receiving station 30 can be implemented using a general purpose computer/processor with a computer program that, when executed, carries out any of the respective methods, algorithms and/or instructions described herein. In addition or alternatively, for example, a special purpose computer/processor can be utilized which can contain specialized hardware for carrying out any of the methods, algorithms, or instructions described herein.

[0071] Transmitting station 12 and receiving station 30 can, for example, be implemented on computers in a screen casting system. Alternatively, transmitting station 12 can be implemented on a server and receiving station 30 can be implemented on a device separate from the server, such as a hand-held communications device (i.e. a cell phone). In this instance, transmitting station 12 can encode content using an encoder into an encoded video signal and transmit the encoded video signal to the communications device. In turn, the communications device can then decode the encoded video signal using a decoder. Alternatively, the communications device can decode content stored locally on the communications device (i.e. no transmission is necessary). Other suitable transmitting station 12 and receiving station 30 implementation schemes are available. For example, receiving station 30 can be a personal computer rather than a portable communications device.

[0072] Further, all or a portion of embodiments of the present invention can take the form of a computer program product accessible from, for example, a computer-usable or computer-readable medium. A computer-usable or computer-readable medium can be any device that can, for example, tangibly contain, store, communicate, or transport the program for use by or in connection with any processor. The medium can be, for example, an electronic, magnetic, optical, electromagnetic, or a semiconductor device. Other suitable mediums are also available.

[0073] The above-described embodiments have been described in order to allow easy understanding of the present invention and do not limit the present invention. On the contrary, the invention is intended to cover various modifications and equivalent arrangements included within the scope of the appended claims, which scope is to be accorded the broadest interpretation so as to encompass all such modifications and equivalent structure as is permitted under the law.

What is claimed is:

1. A method for encoding a video signal having at least one frame with a plurality of blocks, comprising:
 - assigning at least some of the plurality of blocks to a segment;
 - determining at least one prediction element for the segment;
 - applying the at least one prediction element to a first block and at least some of the other blocks in the segment; and
 - encoding the first block and the other blocks in the segment.
2. The method of claim 1, wherein the at least one prediction element is a prediction mode, a reference frame or a motion vector.
3. The method of claim 1 or claim 2, wherein the at least one prediction element includes a first prediction element and at least a second prediction element different than the first element.
4. The method of claim 3, further comprising at least one of:
 - applying the first prediction element and not the second prediction element to a second block in the segment; or
 - applying the first prediction element and the second prediction element to the first block in the segment.
5. The method of claim 3, wherein the first prediction element is at least one of a prediction mode, a reference frame or a motion vector and the second prediction element is at least one of a prediction mode, reference frame or a motion vector.
6. The method of claim 1 or claim 2, wherein the at least some of the blocks are assigned to the segment based on at least one of:
 - blocks having a similar motion vector in the at least one frame;
 - blocks having a similar color value in the at least one frame; or
 - blocks in a similar spatial location in the at least one frame.

7. The method of claim 6, wherein spatial location includes blocks that are at least one of neighboring blocks or blocks with similar texture characteristics.

8. The method of claim 1 or claim 2, further comprising:
identifying prediction information for a first block in the segment, wherein the at least one prediction element is based on the first block's prediction information.

9. The method of claim 1 or claim 2, further comprising:
identifying prediction information for a first block and other blocks in the segment;
assigning blocks having similar prediction information to groups; and
determining which group has the greatest number blocks, wherein the at least one prediction element is based on the prediction information associated with the group having the greatest number of blocks.

10. The method of claim 1 or claim 2, further comprising:
identifying prediction information for a first block and at least one other block in the segment, wherein the at least one prediction element is based at least on the first and the at least one other block's prediction information.

11. The method of claim 10, wherein the prediction information of the first and the at least one other block each includes a motion vector, further comprising:
determining an average of the first block and the at least one other block's motion vectors; and
selecting the average as the at least one prediction element for the segment.

12. The method of claim 1 or claim 2, wherein a value of the at least one prediction element is one of an absolute value and an initial value; and wherein when the value is the initial value, applying the at least one prediction element comprises:
identifying a block-specific change value for the first block;
combining the initial value and the block-specific value to the first block;
and
applying the combination to the first block.

13. An apparatus for encoding a video signal having at least one frame with a plurality of blocks, comprising:

- a memory; and
- a processor configured to execute instructions stored in the memory to:
 - assign at least some of the plurality of blocks to a segment;
 - determine at least one prediction element for the segment;
 - apply at least one prediction element to a first block and at least some of the other blocks in the segment; and
 - encode the first block and the other blocks in the segment.

14. The apparatus of claim 13, wherein the at least one prediction element is a prediction mode, a reference frame or a motion vector.

15. The apparatus of claim 13 or claim 14, wherein the processor is further configured to execute instructions stored in the memory to:

- apply the at least one prediction element to all blocks in the segment.

16. The apparatus of claim 13 or claim 14, wherein the at least one prediction element includes a first prediction element and at least a second prediction element different than the first element and wherein the processor is further configured to execute instructions stored in the memory to:

- apply the first prediction element and not the second prediction element to a second block in the segment.

17. The apparatus of claim 13 or claim 14, wherein the processor is further configured to execute instructions stored in the memory to:

- identify prediction information for a first block and other blocks in the segment;
- assign blocks having similar prediction information to groups; and
- determine which group has the greatest number blocks, wherein the at least one prediction element is based on the prediction information associated with the group having the greatest number of blocks.

18. A method for decoding a video signal having at least one frame with a plurality of blocks, comprising:

assigning at least some of the plurality of blocks to a segment;
obtaining at least one prediction element for the segment;
applying the at least one prediction element to a first block and at least some of the other blocks in the segment; and
decoding the first block using the at least one prediction element.

19. The method of claim 18 wherein assigning the at least some of the plurality of blocks to the segment comprises assigning the at least some of the plurality of blocks to the segment based on one or more of motion information, color information or spatial information.

20. The method of claim 18, further comprising:
obtaining the segment from a segmentation map coded within a bitstream.

1/7

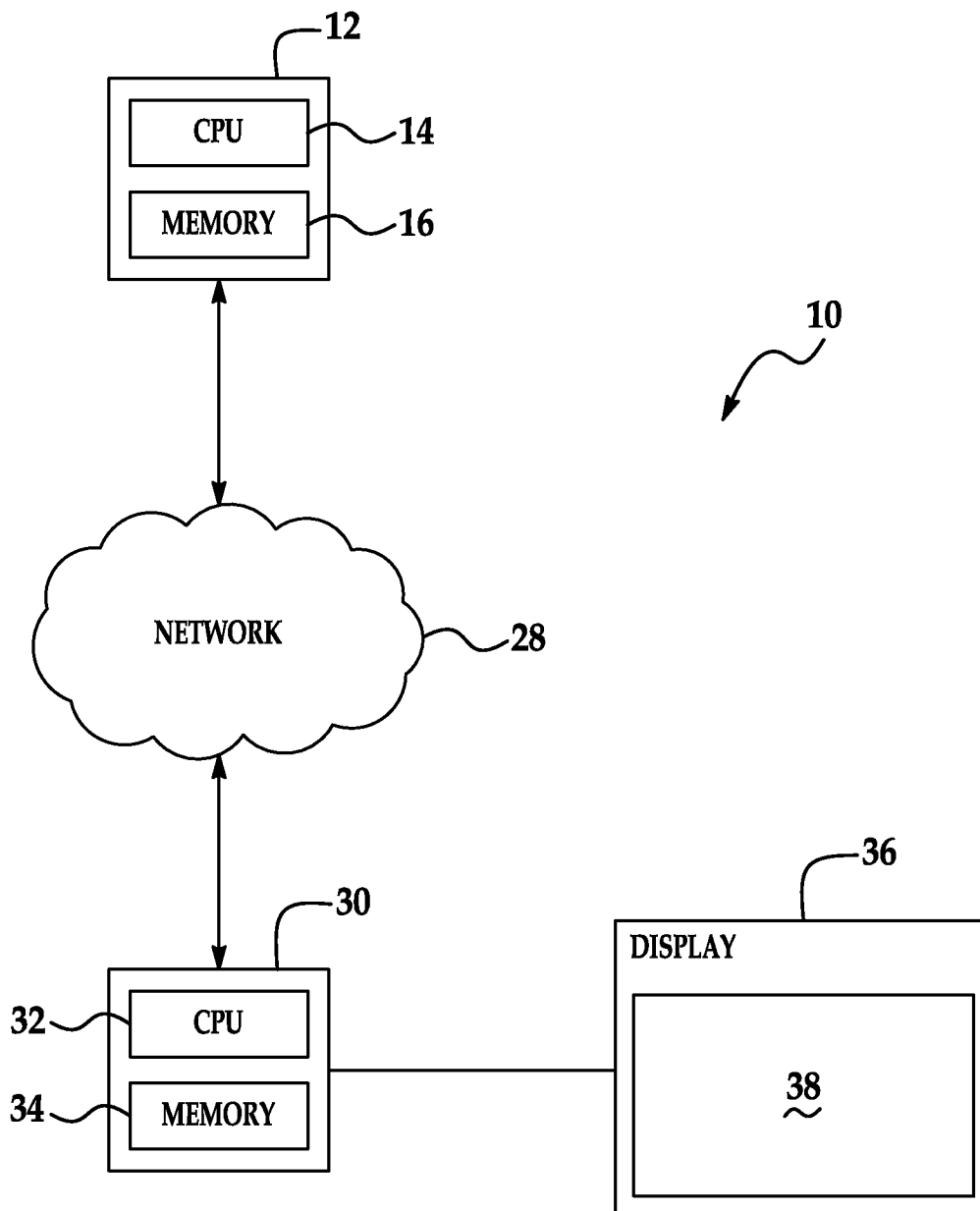


FIG. 1

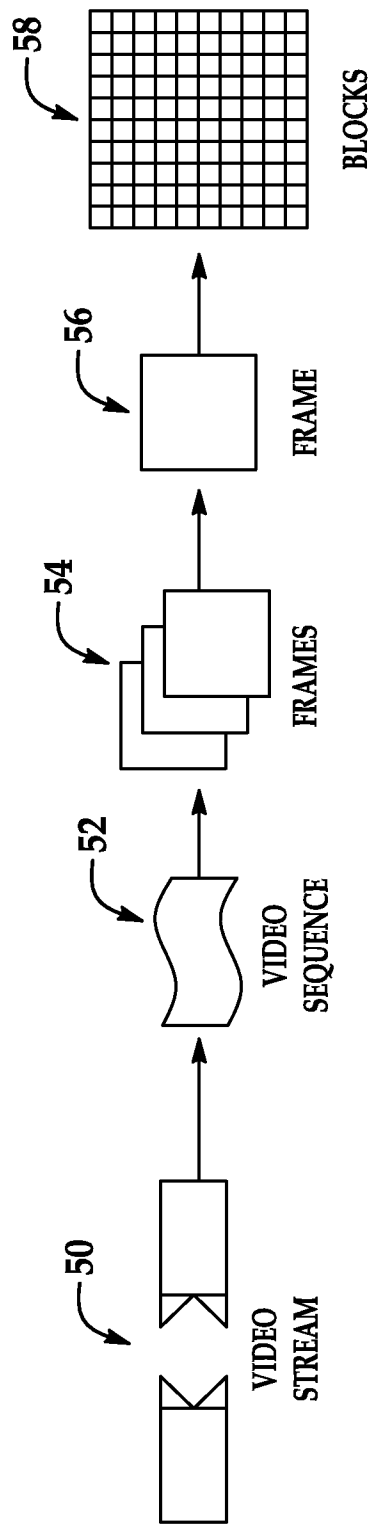


FIG. 2

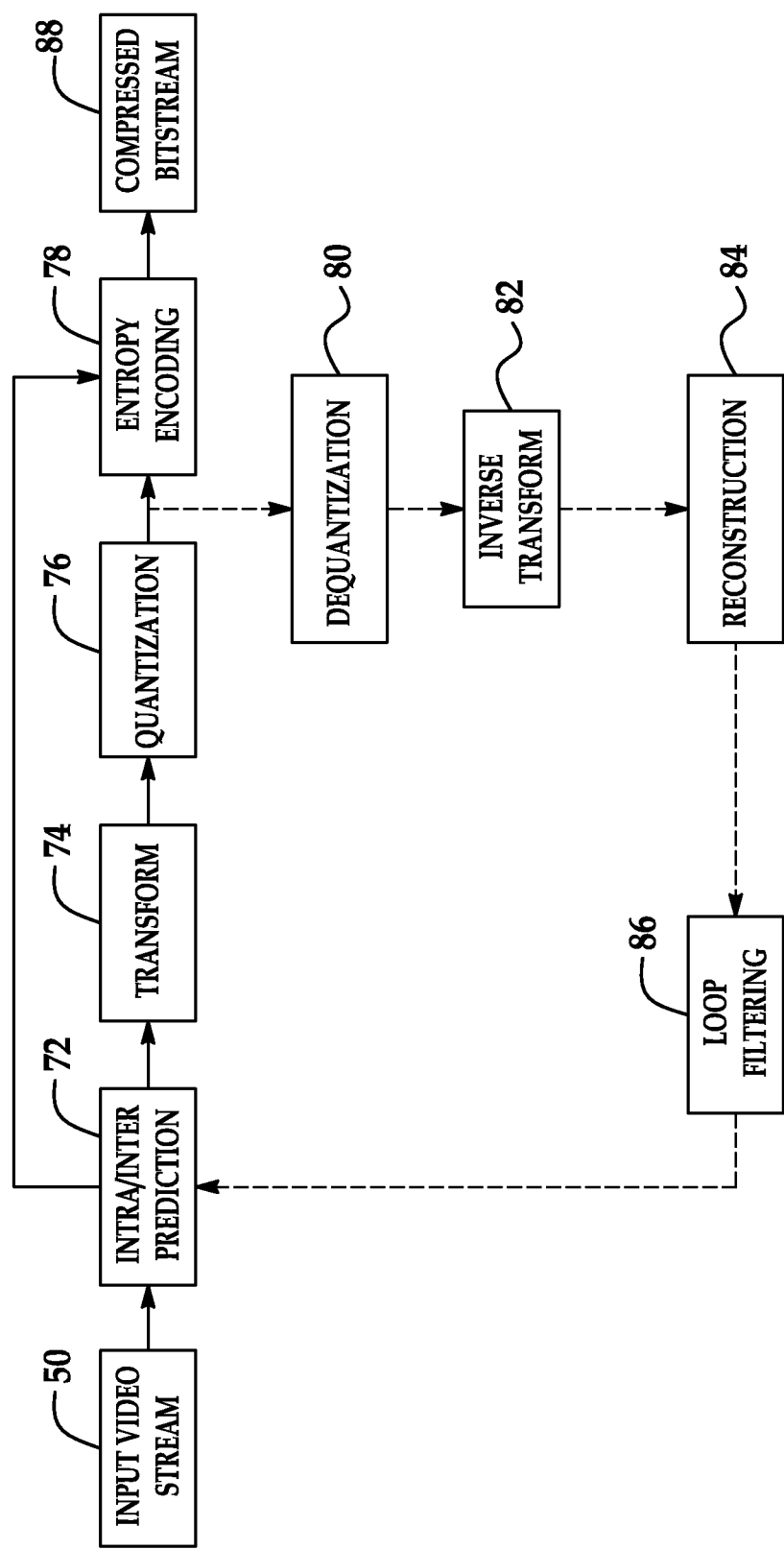


FIG. 3

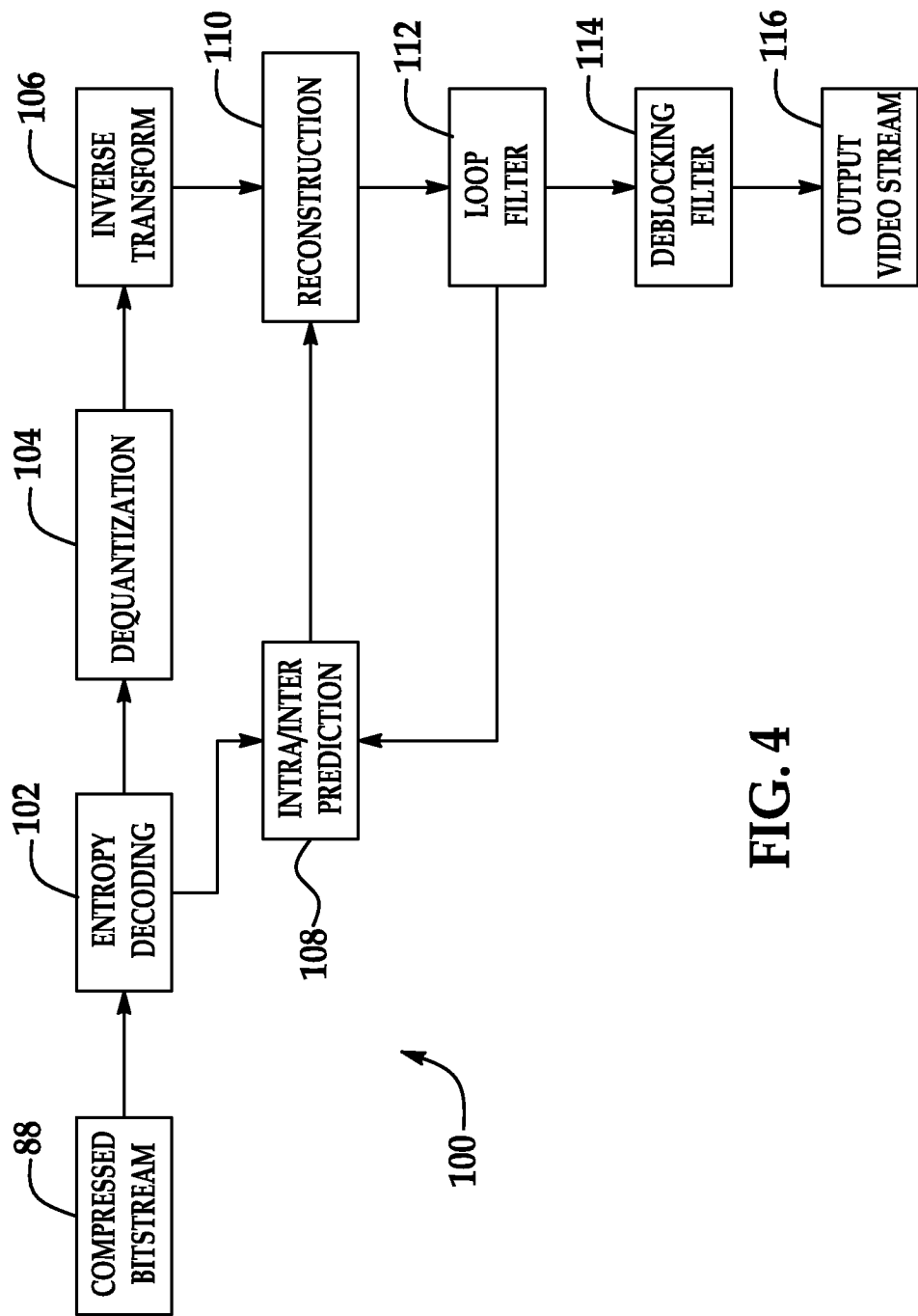


FIG. 4

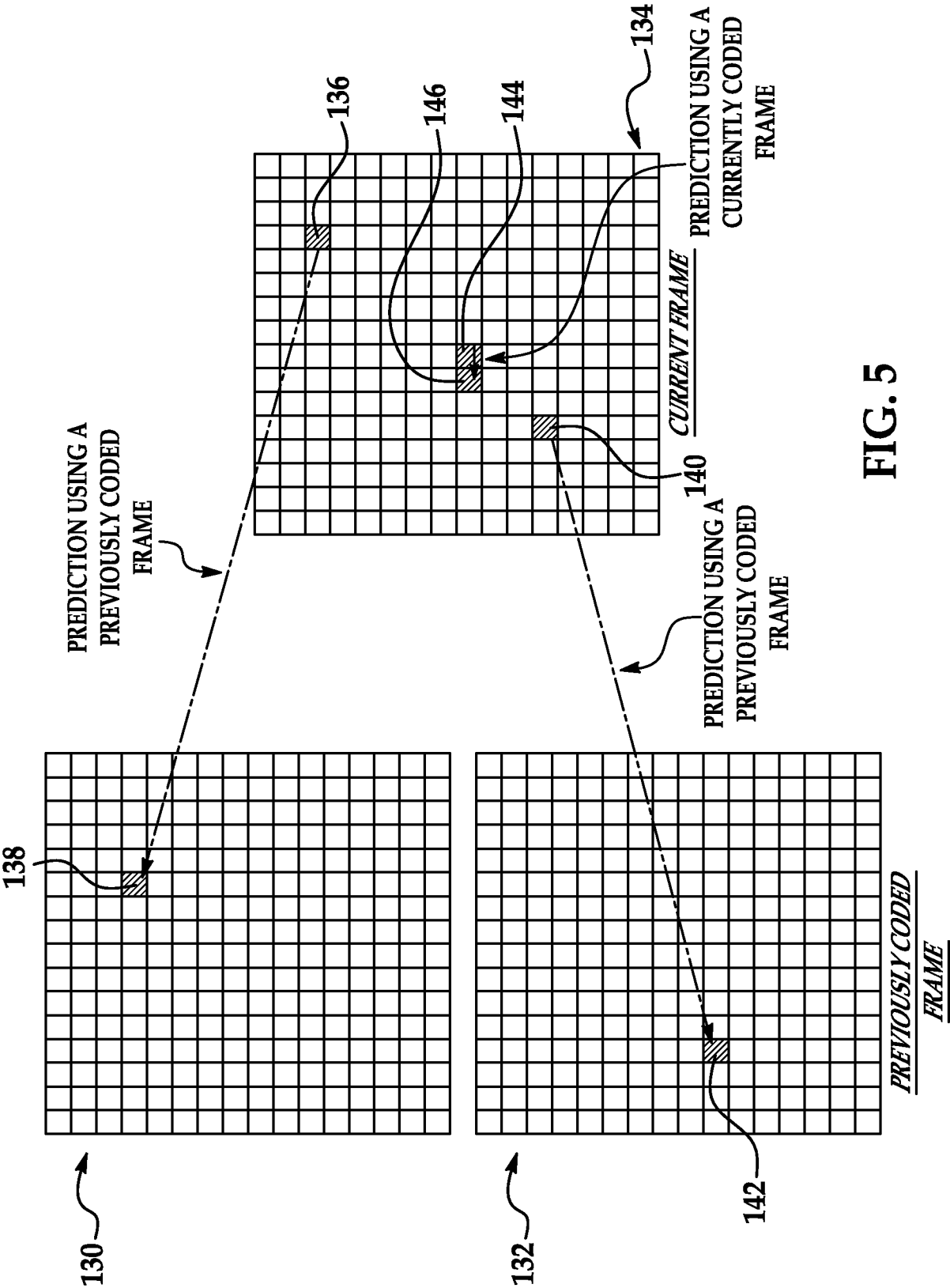


FIG. 5

6/7

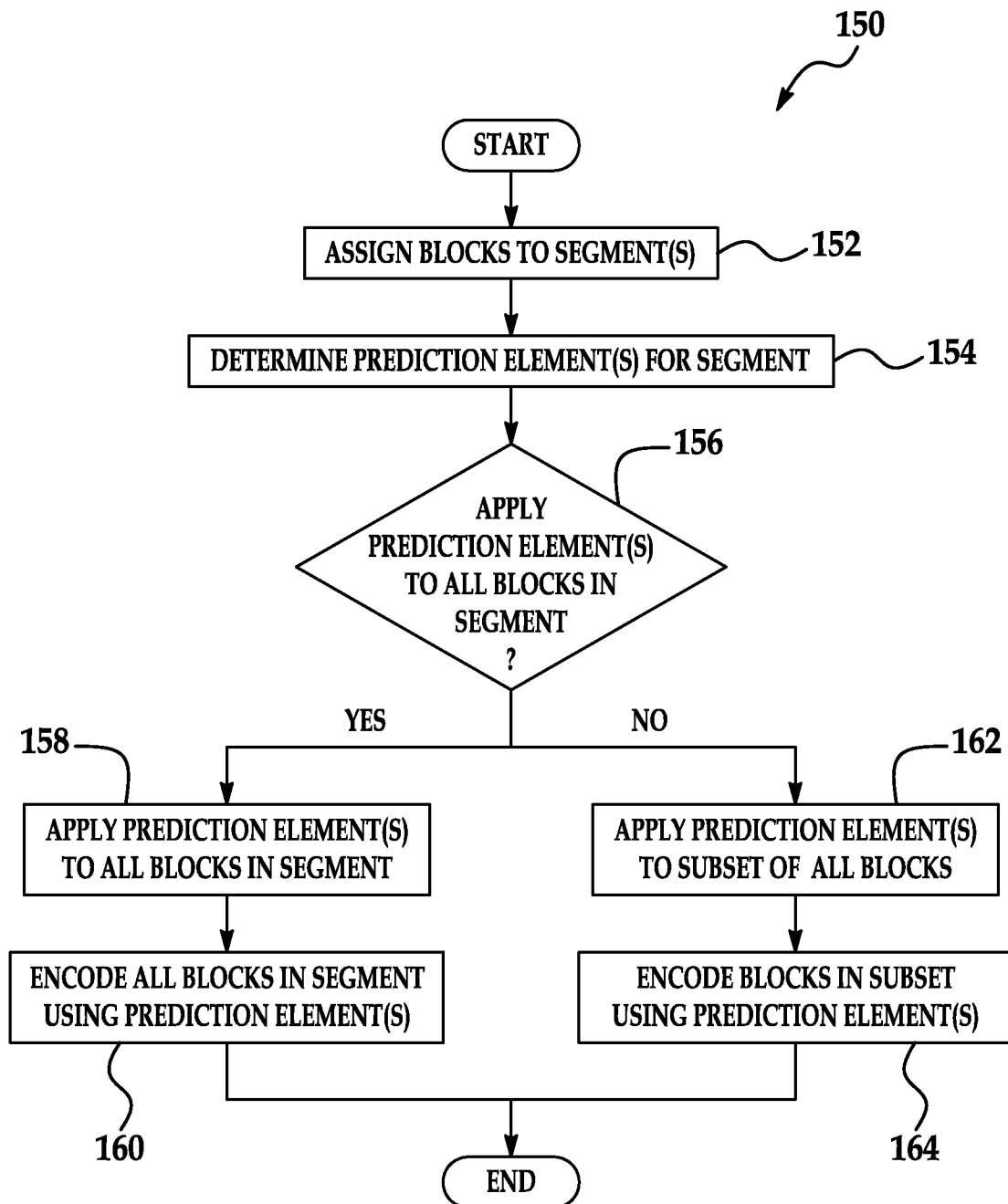


FIG. 6

MV=NEW B0	MV=B0 B1	MV=B0 B2	MV=B0 B3
MV=B0 B4	MV=NEW B5	MV=B5 B6	MV=B0 B7
MV=B0 B8	MV=B5 B9	MV=B5 B10	MV=B0 B11
MV=NEW B12	MV=NEW B13	MV=NEW B14	MV=B0 B15

200

FIG. 7A

1 B0	1 B1	1 B2	1 B3
1 B4	2 B5	2 B6	1 B7
1 B8	2 B9	2 B10	1 B11
B12	B13	B14	1 B15

200

202

FIG. 7B