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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 October 2007 (11.10.2007)

PCT

(10) International Publication Number
WO 2007/114609 A1

(51) International Patent Classification:
H04N 7/24 (2006.01) *H04N 7/36* (2006.01)
H04N 7/32 (2006.01)

(21) International Application Number:
PCT/KR2007/001581

(22) International Filing Date: 30 March 2007 (30.03.2007)

(25) Filing Language: Korean

(26) Publication Language: English

(30) Priority Data:

60/787,171	30 March 2006 (30.03.2006)	US
60/801,398	19 May 2006 (19.05.2006)	US
60/810,642	5 June 2006 (05.06.2006)	US
60/830,601	14 July 2006 (14.07.2006)	US
60/832,153	21 July 2006 (21.07.2006)	US
60/837,925	16 August 2006 (16.08.2006)	US
60/840,032	25 August 2006 (25.08.2006)	US
60/842,152	5 September 2006 (05.09.2006)	US

(71) Applicant *(for all designated States except US)*: **LG ELECTRONICS INC.** [KR/KR]; 20, Yoido-dong, Youngdungpo-gu, Seoul 150-721 (KR).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **JEON, Byeong Moon** [KR/KR]; 306-1005 Hyundai 3-cha Apt., Gwangjang-dong, Gwangjin-gu, Seoul 143-754 (KR). **PARK, Seung Wook** [KR/KR]; 1429-7 Sillim 5-dong, Gwanak-gu, Seoul 151-891 (KR). **KOO, Han Suh** [KR/KR]; 108-8 Nonhyun-dong, Gangnam-gu, Seoul 135-101 (KR). **JEON, Yong Joon** [KR/KR]; B-1405 Royal Palace Houseville, Sunae-dong, Bundang-gu, Seongnam-si, Gyeonggi-do 463-020 (KR).

(74) Agents: **KIM, Yong In** et al.; 15th Floor, Yo Sam Building, 648-23 Yeoksam-dong, Kangnam-gu, Seoul 135-808 (KR).

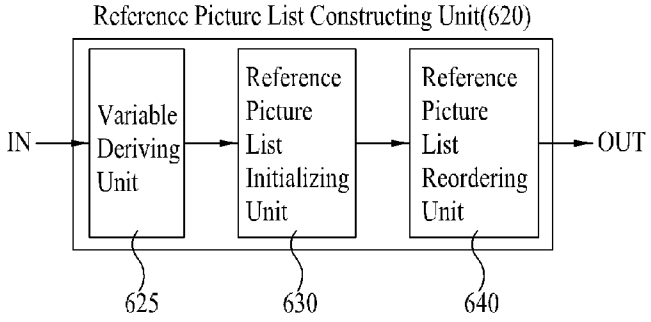
(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, 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, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*
— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: A METHOD AND APPARATUS FOR DECODING/ENCODING A VIDEO SIGNAL



(57) Abstract: The present invention provides a method of decoding a video signal. The method includes the steps of obtaining view information of a picture from the video signal and generating information for reference picture management using the view information.

WO 2007/114609 A1

A METHOD AND APPARATUS FOR DECODING/ENCODING A VIDEO SIGNAL

TECHNICAL FIELD

The present invention relates to a method for decoding/encoding a video signal and apparatus thereof.

BACKGROUND ART

Compression encoding means a series of signal processing techniques for transmitting digitalized information via communication circuit or storing digitalized information in a form suitable for a storage medium. Objects for the compression encoding include audio, video, text, and the like. In particular, a technique for performing compression encoding on a sequence is called video sequence compression. The video sequence is generally characterized in having spatial redundancy and temporal redundancy.

DISCLOSURE OF THE INVENTION

TECHNICAL OBJECT

The object of the present invention is to enhance coding efficiency of a video signal.

TECHNICAL SOLUTION

An object of the present invention is to code a video signal efficiently by defining view information capable of identifying a view of picture.

Another object of the present invention is to code a video signal efficiently by providing a method of managing reference pictures used for inter-view prediction.

Another object of the present invention is to code a video signal efficiently by providing a method of constructing a reference picture list for inter-view prediction.

Another object of the present invention is to code a video signal efficiently by providing a method of reordering a reference picture list for inter-view prediction.

ADVANTAGEOUS EFFECTS

In coding a video signal, the present invention enables coding to be more efficiently performed by providing a method of managing reference pictures used for inter-view prediction. And, the present invention enables coding to be more efficiently performed by providing a method of initializing a reference picture list for inter-view prediction and a method of reordering a reference picture list for inter-view prediction. In performing

inter-view prediction using the present invention, it is able to improve a coding rate by reducing a load of DPB (decoded picture buffer). And, the present invention enables more accurate prediction, thereby reducing the number of bits to be transmitted.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic block diagram of an apparatus for decoding a video signal according to the present invention.

FIG. 2 is a diagram of configuration information for a multi-view video addable to a multi-view video coded bit stream according to an embodiment of the present invention.

FIG. 3 is an internal block diagram of a reference picture list constructing unit 620 according to an embodiment of the present invention.

FIG. 4 is a diagram of a hierarchical structure of level information for providing view scalability of a video signal according to an embodiment of the present invention.

FIG. 5 is a diagram of a NAL-unit configuration including level information within an extension area of a NAL header according to one embodiment of the present invention.

FIG. 6 is a diagram of an overall predictive

structure of a multi-view video signal according to an embodiment of the present invention to explain a concept of an inter-view picture group.

FIG. 7 is a diagram of a predictive structure according to an embodiment of the present invention to explain a concept of a newly defined inter-view picture group.

FIG. 8 is a schematic block diagram of an apparatus for decoding a multi-view video using inter-view picture group identification information according to an embodiment of the present invention.

FIG. 9 is a flowchart of a process for constructing a reference picture list according to an embodiment of the present invention.

FIG. 10 is a diagram to explain a method of initializing a reference picture list when a current slice is a P-slice according to one embodiment of the present invention.

FIG. 11 is a diagram to explain a method of initializing a reference picture list when a current slice is a B-slice according to one embodiment of the present invention.

FIG. 12 is an internal block diagram of a reference picture list reordering unit 630 according to an embodiment

of the present invention.

FIG. 13 is an internal block diagram of a reference index assignment changing unit 643B or 645B according to one embodiment of the present invention.

FIG. 14 is a diagram to explain a process for reordering a reference picture list using view information according to one embodiment of the present invention.

FIG. 15 is an internal block diagram of a reference picture list reordering unit 630 according to another embodiment of the present invention.

FIG. 16 is an internal block diagram of a reference picture list reordering unit 970 for inter-view prediction according to an embodiment of the present invention.

FIG. 17 and FIG. 18 are diagrams of syntax for reference picture list reordering according to one embodiment of the present invention.

FIG. 19 is a diagram of syntax for reference picture list reordering according to another embodiment of the present invention.

FIG. 20 is a diagram for a process for obtaining an illumination difference value of a current block according to one embodiment of the present invention.

FIG. 21 is a flowchart of a process for performing illumination compensation of a current block according to

an embodiment of the present invention.

FIG. 22 is a diagram of a process for obtaining an illumination difference prediction value of a current block using information for a neighboring block according to one embodiment of the present invention.

FIG. 23 is a flowchart of a process for performing illumination compensation using information for a neighboring block according to one embodiment of the present invention.

FIG. 24 is a flowchart of a process for performing illumination compensation using information for a neighboring block according to another embodiment of the present invention.

FIG. 25 is a diagram of a process for predicting a current picture using a picture in a virtual view according to one embodiment of the present invention.

FIG. 26 is a flowchart of a process for synthesizing a picture in a virtual view in performing an inter-view prediction in MVC according to an embodiment of the present invention.

FIG. 27 is a flowchart of a method of executing a weighted prediction according to a slice type in video signal coding according to the present invention.

FIG. 28 is a diagram of macroblock types allowable in

a slice type in video signal coding according to the present invention.

FIG. 29 and FIG. 30 are diagrams of syntax for executing a weighted prediction according to a newly defined slice type according to one embodiment of the present invention.

FIG. 31 is a flowchart of a method of executing a weighted prediction using flag information indicating whether to execute inter-view weighted prediction in video signal coding according to the present invention.

FIG. 32 is a diagram to explain a weighted prediction method according to flag information indicating whether to execute a weighted prediction using information for a picture in a view different from that of a current picture according to one embodiment of the present invention.

FIG. 33 is a diagram of syntax for executing a weighted prediction according to a newly defined flag information according to one embodiment of the present invention.

FIG. 34 is a flowchart of a method of executing a weighted prediction according to a NAL (network abstraction layer) unit type according to an embodiment of the present invention.

FIG. 35 and FIG. 36 are diagrams of syntax for

executing a weighted prediction in case that a NAL unit type is for multi-view video coding according to one embodiment of the present invention.

FIG. 37 is a partial block diagram of a video signal decoding apparatus according to a newly defined slice type according to an embodiment of the present invention.

FIG. 38 is a flowchart to explain a method of decoding a video signal in the apparatus shown in FIG. 37 according to the present invention.

FIG. 39 is a diagram of a macroblock prediction mode according to one embodiment of the present invention.

FIG. 40 and FIG. 41 are diagrams of syntax having slice type and macroblock mode applied thereto according to the present invention.

FIG. 42 is a diagram of embodiments to which the slice types in FIG. 41 are applied.

FIG. 43 is a diagram of various embodiments of the slice type included in the slice types shown in FIG. 41.

FIG. 44 is a diagram of a macroblock allowable for a mixed slice type by prediction of two mixed predictions according to one embodiment of the present invention.

FIGs. 45 to 47 are diagrams of a macroblock type of a macroblock existing in a mixed slice by prediction of two mixed predictions according to one embodiment of the

present invention.

FIG. 48 is a partial block diagram of a video signal encoding apparatus according to a newly defined slice type according to an embodiment of the present invention.

FIG. 49 is a flowchart of a method of encoding a video signal in the apparatus shown in FIG. 48 according to the present invention.

BEST MODE FOR CARRYING OUT THE INVENTION

To achieve these and other advantages and in accordance with the purpose of the present invention, as embodied and broadly described, a method of decoding a video signal includes the steps of obtaining view information for identifying a view of a picture from the video signal and generating information for reference picture management using the view information.

To further achieve these and other advantages and in accordance with the purpose of the present invention, a method of decoding a video signal includes the steps of obtaining view information for identifying a view of a picture from the video signal and deriving a first variable for constructing a reference picture list for inter-view prediction using the view information, wherein the reference picture list for the inter-view prediction is

constructed using the first variable.

MODE FOR INVENTION

Reference will now be made in detail to the preferred embodiments of the present invention, examples of which are illustrated in the accompanying drawings.

The technique of compressing and encoding video signal data considers spatial redundancy, temporal redundancy, scalable redundancy, and inter-view redundancy. And, it is also able to perform a compression coding by considering a mutual redundancy between views in the compression encoding process. The technique for the compression coding, which considers the inter-view redundancy, is just an embodiment of the present invention. And, the technical idea of the present invention is applicable to the temporal redundancy, the scalable redundancy, etc.

Looking into a configuration of a bit stream in H.264/AVC, there exists a separate layer structure called a NAL (network abstraction layer) between a VCL (video coding layer) dealing with a moving picture encoding process itself and a lower system that transports and stores encoded information. An output from an encoding process is VCL data and is mapped by NAL unit prior to transport or

storage. Each NAL unit includes compressed video data or RBSP (raw byte sequence payload: result data of moving picture compression) that is the data corresponding to header information.

The NAL unit basically includes a NAL header and an RBSP. The NAL header includes flag information (nal_ref_idc) indicating whether a slice as a reference picture of the NAL unit is included and an identifier (nal_unit_type) indicating a type of the NAL unit. Compressed original data is stored in the RBSP. And, RBSP trailing bit is added to a last portion of the RBSP to represent a length of the RBSP as an 8-bit multiplication. As the type of the NAL unit, there is IDR (instantaneous decoding refresh) picture, SPS (sequence parameter set), PPS (picture parameter set), SEI (supplemental enhancement information), or the like.

In the standardization, restrictions for various profiles and levels are set to enable implementation of a target product with an appropriate cost. In this case, a decoder should meet the restriction decided according the corresponding profile and level. Thus, two concepts, 'profile' and 'level' are defined to indicate a function or parameter for representing how far the decoder can cope with a range of a compressed sequence. And, a profile

indicator (profile_idc) can identify that a bit stream is based on a prescribed profile. The profile indicator means a flag indicating a profile on which a bit stream is based. For instance, in H.264/AVC, if a profile indicator is 66, it means that a bit stream is based on a baseline profile. If a profile indicator is 77, it means that a bit stream is based on a main profile. If a profile indicator is 88, it means that a bit stream is based on an extended profile. And, the profile identifier can be included in a sequence parameter set.

So, in order to deal with a multi-view video, it needs to be identified whether a profile of an inputted bit stream is a multi-view profile. If the profile of the inputted bit stream is the multi-view profile, it is necessary to add syntax to enable at least one additional information for multi-view to be transmitted. In this case, the multi-view profile indicates a profile mode handling multi-view video as an amendment technique of H.264/AVC. In MVC, it may be more efficient to add syntax as additional information for an MVC mode rather than unconditional syntax. For instance, when a profile indicator of AVC indicates a multi-view profile, if information for a multi-view video is added, it is able to enhance encoding efficiency.

A sequence parameter set indicates header information containing information crossing over coding of an overall sequence such as a profile, a level, and the like. A whole compressed moving picture, i.e., a sequence should begin at a sequence header. So, a sequence parameter set corresponding to header information should arrive at a decoder before data referring to the parameter set arrives. Namely, the sequence parameter set RBSP plays a role as the header information for the result data of the moving picture compression. Once a bit stream is inputted, a profile indicator preferentially identifies that the inputted bit stream is based on which one of a plurality of profiled. So, by adding a part for deciding whether an inputted bit stream relates to a multi-view profile (e.g., 'If (profile_idc==MULTI_VIEW_PROFILE)') to syntax, it is decided whether the inputted bit stream relates to the multi-view profile. Various kinds of configuration information can be added only if the inputted bit stream is approved as relating to the multi-view profile. For instance, it is able to add a number of total views, a number of inter-view reference pictures (List0/1) in case of an inter-view picture group, a number of inter-view reference pictures (List0/1) in case of a non-inter-view picture group, and the like. And, various informations for

view are usable for generation and management of a reference picture list in a decoded picture buffer.

FIG. 1 is a schematic block diagram of an apparatus for decoding a video signal according to the present invention.

Referring to FIG. 1, an apparatus for decoding a video signal according to the present invention includes a NAL parser 100, an entropy decoding unit 200, an inverse quantization/inverse transform unit 300, an intra-prediction unit 400, a deblocking filter unit 500, a decoded picture buffer unit 600, an inter-prediction unit 700, and the like.

The decoded picture buffer unit 600 includes a reference picture storing unit 610, a reference picture list constructing unit 620, a reference picture managing unit 650, and the like. And, the reference picture list constructing unit 620 includes a variable deriving unit 625, a reference picture list initializing unit 630, and a reference picture list reordering unit 640.

And, the inter-prediction unit 700 includes a motion compensation unit 710, an illumination compensation unit 720, an illumination difference prediction unit 730, a view synthesis prediction unit 740, and the like.

The NAL parser 100 carries out parsing by NAL unit to

decode a received video sequence. In general, at least one sequence parameter set and at least one picture parameter set are transferred to a decoder before a slice header and slice data are decoded. In this case, various kinds of configuration informations can be included in a NAL header area or an extension area of a NAL header. Since MVC is an amendment technique for a conventional AVC technique, it may be more efficient to add the configuration informations in case of an MVC bit stream only rather than unconditional addition. For instance, it is able to add flag information for identifying a presence or non-presence of an MVC bit stream in the NAL header area or the extension area of the NAL header. Only if an inputted bit stream is a multi-view video coded bit stream according to the flag information, it is able to add configuration informations for a multi-view video. For instance, the configuration informations can include temporal level information, view level information, inter-view picture group identification information, view identification information, and the like. This is explained in detail with reference to FIG. 2 as follows.

FIG. 2 is a diagram of configuration information for a multi-view video addable to a multi-view video coded bit stream according to one embodiment of the present invention.

Details of configuration information for a multi-view video are explained in the following description.

First of all, temporal level information indicates information for a hierarchical structure to provide temporal scalability from a video signal (①). Through the temporal level information, it is able to provide a user with sequences on various time zones.

View level information indicates information for a hierarchical structure to provide view scalability from a video signal (②). In a multi-view video, it is necessary to define a level for a time and a level for a view to provide a user with various temporal and view sequences. In case of defining the above level information, it is able to use temporal scalability and view scalability. Hence, a user is able to select a sequence at a specific time and view, or a selected sequence may be restricted by a condition.

The level informations can be set in various ways according to a specific condition. For instance, the level information can be set differently according to camera location or camera alignment. And, the level information can be determined by considering view dependency. For instance, a level for a view having I-picture in an inter-view picture group is set to 0, a level for a view having

P-picture in the inter-view picture group is set to 1, and a level for a view having B-picture in the inter-view picture group is set to 2. Moreover, the level informations can be randomly set not based on a special condition. The view level information will be explained in detail with reference to FIG. 4 and FIG. 5 later.

Inter-view picture group identification information indicates information for identifying whether a coded picture of a current NAL unit is an inter-view picture group (③). In this case, the inter-view picture group means a coded picture in which all slices reference only slices with the same picture order count. For instance, the inter-view picture group means an coded picture that refers to slices in a different view only without referring to slices in a current view. In a decoding process of a multi-view video, an inter-view random access may be needed. The inter-view picture group identification information may be necessary to realize an efficient random access. And, inter-view reference information may be necessary for inter-view prediction. So, inter-view picture group identification information can be used to obtain the inter-view reference information. Moreover, the inter-view picture group identification information can be used to add reference pictures for inter-view prediction in

constructing a reference picture list. Besides, the inter-view picture group identification information can be used to manage the added reference pictures for the inter-view prediction. For instance, the reference pictures may be classified into inter-view picture groups and non-inter-view picture groups and the classified reference pictures can be then marked that the reference pictures failing to be used for the inter-view prediction shall not be used. Meanwhile, the inter-view picture group identification information is applicable to a hypothetical reference decoder. Details of the inter-view picture group identification information will be explained with reference to FIG. 6 later.

The view identification information means information for discriminating a picture in a current view from a picture in a different view (④). In coding a video signal, POC (picture order count) or 'frame_num' may be used to identify each picture. In case of a multi-view video sequence, inter-view prediction can be executed. So, identification information to discriminate a picture in a current view from a picture in another view is needed. So, it is necessary to define view identification information for identifying a view of a picture. The view identification information can be obtained from a header

area of a video signal. For instance, the header area can be a NAL header area, an extension area of a NAL header, or a slice header area. Information for a picture in a view different from that of a current picture is obtained using the view identification information and it is able to decode the video signal using the information of the picture in the different view. The view identification information is applicable to an overall encoding/decoding process of the video signal. And, the view identification information can be applied to multi-view video coding using the 'frame_num' that considers a view instead of considering a specific view identifier.

Meanwhile, the entropy decoding unit 200 carries out entropy decoding on a parsed bit stream, and a coefficient of each macroblock, a motion vector, and the like are then extracted. The inverse quantization/ inverse transform unit 300 obtains a transformed coefficient value by multiplying a received quantized value by a constant and then transforms the coefficient value inversely to reconstruct a pixel value. Using the reconstructed pixel value, the intra-prediction unit 400 performs an intra prediction from a decoded sample within a current picture. Meanwhile, the deblocking filter unit 500 is applied to each coded macroblock to reduce block distortion. A filter smoothens a

block edge to enhance an image quality of a decoded frame. A selection of a filtering process depends on boundary strength and gradient of an image sample around a boundary. Pictures through filtering are outputted or stored in the decoded picture buffer unit 600 to be used as reference pictures.

The decoded picture buffer unit 600 plays a role in storing or opening the previously coded pictures to perform an inter prediction. In this case, to store the pictures in the decoded picture buffer unit 600 or to open the pictures, 'frame_num' and POC (picture order count) of each picture are used. So, since there exist pictures in a view different from that of a current picture among the previously coded pictures, view information for identifying a view of a picture may be usable together with the 'frame_num' and the POC. The decoded picture buffer unit 600 includes the reference picture storing unit 610, the reference picture list constructing unit 620, and the reference picture managing unit 650. The reference picture storing unit 610 stores pictures that will be referred to for the coding of the current picture. The reference picture list constructing unit 620 constructs a list of reference pictures for the inter-picture prediction. In multi-view video coding, inter-view prediction may be

needed. So, if a current picture refers to a picture in another view, it may be necessary to construct a reference picture list for the inter-view prediction. In this case, the reference picture list constructing unit 620 can use information for view in generating the reference picture list for the inter-view prediction. Details of the reference picture list constructing unit 620 will be explained with reference to FIG. 3 later.

FIG. 3 is an internal block diagram of a reference picture list constructing unit 620 according to an embodiment of the present invention.

The reference picture list constructing unit 620 includes the variable deriving unit 625, the reference picture list initializing unit 630, and the reference list reordering unit 640.

The variable deriving unit 625 derives variables used for reference picture list initialization. For instance, the variable can be derived using 'frame_num' indicating a picture identification number. In particular, variables FrameNum and FrameNumWrap may be usable for each short-term reference picture. First of all, the variable FrameNum is equal to a value of a syntax element frame_num. The variable FrameNumWrap can be used for the decoded picture buffer unit 600 to assign a small number to each reference

picture. And, the variable FrameNumWrap can be derived from the variable FrameNum. So, it is able to derive a variable PicNum using the derived variable FrameNumWrap. In this case, the variable PicNum can mean an identification number of a picture used by the decoded picture buffer unit 600. In case of indicating a long-term reference picture, a variable LongTermPicNum can be usable.

In order to construct a reference picture list for inter-view prediction, it is able to derive a first variable (e.g., ViewNum) to construct a reference picture list for inter-view prediction. For instance, it is able to derive a second variable (e.g., ViewId) using 'view_id' for identifying a view of a picture. First of all, the second variable can be equal to a value of the syntax element 'view_id'. And, a third variable (e.g., ViewIdWrap) can be used for the decoded picture buffer unit 600 to assign a small view identification number to each reference picture and can be derived from the second variable. In this case, the first variable ViewNum can mean a view identification number of picture used by the decoded picture buffer unit 600. Yet, since a number of reference pictures used for inter-view prediction in multi-view video coding may be relatively smaller than that used for temporal prediction, it may not define another variable to indicate a view

identification number of a long-term reference picture.

The reference picture list initializing unit 630 initializes a reference picture list using the above-mentioned variables. In this case, an initialization process for the reference picture list may differ according to a slice type. For instance, in case of decoding a P-slice, it is able to assign a reference index based on a decoding order. In case of decoding a B-slice, it is able to assign a reference index based on a picture output order. In case of initializing a reference picture list for inter-view prediction, it is able to assign an index to a reference picture based on the first variable, i.e., the variable derived from view information.

The reference picture list reordering unit 640 plays a role in enhancing a compression efficiency by assigning a smaller index to a picture frequently referred to in the initialized reference picture list. This is because a small bit is assigned if a reference index for encoding gets smaller.

And, the reference picture list reordering unit 640 includes a slice type checking unit 642, a reference picture list-0 reordering unit 643, and a reference picture list-1 reordering unit 645. If an initialized reference picture list is inputted, the slice type checking unit 642

checks a type of a slice to be decoded and then decides whether to reorder a reference picture list-0 or a reference picture list-1. So, the reference picture list-0/1 reordering unit 643,645 performs reordering of the reference picture list-0 if the slice type is not an I-slice and also performs reordering of the reference picture list-1 additionally if the slice type is a B-slice. Thus, after an end of the reordering process, a reference picture list is constructed.

The reference picture list0/1 reordering unit 643, 645 includes an identification information obtaining unit 643A,645A and a reference index assignment changing unit 643B,645B respectively. The identification information obtaining unit 643A,645A received identification information (reordering_of_pic_nums_idc) indicating an assigning method of a reference index if reordering of a reference picture list is carried out according to flag information indicating whether to execute the reordering of the reference picture list. And, the reference index assignment changing unit 643B,645B reorders the reference picture list by changing an assignment of a reference index according to the identification information.

And, the reference picture list reordering unit 640 is operable by another method. For instance, reordering can

be executed by checking a NAL unit type transferred prior to passing through the slice type checking unit 642 and then classifying the NAL unit type into a case of MVC NAL and a case of non-MVC NAL.

The reference picture managing unit 650 manages reference pictures to execute inter prediction more flexibly. For instance, a memory management control operation method and a sliding window method are usable. This is to manage a reference picture memory and a non-reference picture memory by unifying the memories into one memory and realize an efficient memory management with a small memory. In multi-view video coding, since pictures in a view direction have the same picture order count, information for identifying a view of each of the pictures is usable in marking the pictures in a view direction. And, reference pictures managed in the above manner can be used by the inter-prediction unit 700.

The inter-prediction unit 700 carries out inter prediction using reference pictures stored in the decoded picture buffer unit 600. An inter-coded macroblock can be divided into macroblock partitions. And, each of the macroblock partitions can be predicted from one or two reference pictures. The inter-prediction unit 700 includes the motion compensation unit 710, the illumination

compensation unit 720, the illumination difference prediction unit 730, the view synthesis prediction unit 740, the weighted prediction unit 750, and the like.

The motion compensation unit 710 compensates for a motion of a current block using informations transferred from the entropy decoding unit 200. Motion vectors of neighboring blocks of the current block are extracted from a video signal, and then a motion vector predictor of the current block are derived from the motion vectors of the neighboring blocks. And, the motion of the current block is compensated using the derived motion vector predictor and a differential motion vector extracted from the video signal. And, it is able to perform the motion compensation using one reference picture or a plurality of pictures. In multi-view video coding, in case that a current picture refers to pictures in different views, it is able to perform motion compensation using reference picture list information for the inter-view prediction stored in the decoded picture buffer unit 600. And, it is also able to perform motion compensation using vie information for identifying a view of the reference picture. A direct mode is an coding mode for predicting motion information of a current block from motion information for an encoded block. Since this method is able to save a number of bits required for coding the

motion information, compression efficiency is enhanced. For instance, a temporal direction mode predicts motion information for a current block using a correlation of motion information in a temporal direction. Using a method similar to this method, the present invention is able to predict motion information for a current block using a correlation of motion information in a view direction.

Meanwhile, in case that an inputted bit stream corresponds to a multi-view video, since the respective view sequences are obtained by different camera, an illumination difference is generated by internal and external factors of the cameras. To prevent this, the illumination compensation unit 720 compensates the illumination difference. In performing the illumination compensation, it is able to use flag information indicating whether to perform illumination compensation on a specific layer of a video signal. For instance, it is able to perform an illumination compensation using flag information indicating whether to perform the illumination compensation on a corresponding slice or macroblock. In performing the illumination compensation using the flag information, the illumination compensation is applicable to various macroblock types (e.g., inter 16x16 mode, B-skip mode, direct mode, etc.).

In performing illumination compensation, it is able to use information for a neighboring block or information for a block in a view different from that of a current block to reconstruct the current block. And, it is also able to use an illumination difference value of the current block. In this case, if the current block refers to blocks in a different view, it is able to perform illumination compensation using the reference picture list information for the inter-view prediction stored in the decoded picture buffer unit 600. In this case, the illumination difference value of the current block indicates a difference between an average pixel value of the current block and an average pixel value of a reference block corresponding to the current block. For example of using the illumination difference value, the illumination difference prediction value of the current block is obtained using neighboring blocks of the current block and a difference value (illumination difference residual) between the illumination difference value and the illumination difference prediction value is used. Hence, the decoding unit is able to reconstruct the illumination difference value of the current block using the illumination difference residual and the illumination difference prediction value. In obtaining an illumination difference prediction value of a

current block, it is able to use information for a neighboring block. For instance, it is able to predict an illumination difference value of a current block using an illumination difference value of a neighbor block. Prior to the prediction, it is checked whether a reference index of the current block is equal to that of the neighboring block. According to a result of the checking, it is then decided what kind of a neighboring block or a value will be used.

The view synthesis prediction unit 740 is used to synthesize pictures in a virtual view using pictures in a view neighbor to a view of a current picture and predict the current picture using the synthesized pictures in the virtual view. The decoding unit is able to decide whether to synthesize a picture in a virtual view according to an inter-view synthesis prediction identifier transferred from an encoding unit. For instance, if view_synthesize_pred_flag = 1 or view_syn_pred_flag = 1, a slice or macroblock in a virtual view is synthesized. In this case, when the inter-view synthesis prediction identifier informs that a virtual view will be generated, it is able to generate a picture in the virtual view using view information for identifying a view of picture. And, in predicting a current picture from the synthesized pictures in the virtual view, it is able to use the view information

to use the picture in the virtual view as a reference picture.

The weighted prediction unit 750 is used to compensate for a phenomenon that an image quality of a sequence is considerably degraded in case of encoding the sequence of which brightness temporarily varies. In MVC, weighted prediction can be performed to compensate for a brightness difference from a sequence in a different view as well as it is performed for a sequence of which brightness temporarily varies. For instance, the weighted prediction method can be classified into explicit weighted prediction method and implicit weighted prediction method.

In particular, the explicit weighted prediction method can use one reference picture or two reference pictures. In case of using one reference picture, a prediction signal is generated from multiplying a prediction signal corresponding to motion compensation by a weight coefficient. In case of using two reference pictures, a prediction signal is generated from adding an offset value to a value resulting from multiplying a prediction signal corresponding to motion compensation by a weight coefficient.

And, the implicit weighted prediction performs a weighted prediction using a distance from a reference

picture. As a method of obtaining the distance from the reference picture, it is able to use POC (picture order count) indicating a picture output order for example. In this case, the POC may be obtained by considering identification of a view of each picture. In obtaining a weight coefficient for a picture in a different view, it is able to use view information for identifying a view of a picture to obtain a distance between views of the respective pictures.

In video signal coding, depth information is usable for a specific application or another purpose. In this case, the depth information may mean information capable of indicating an inter-view disparity difference. For instance, it is able to obtain a disparity vector by inter-view prediction. And, the obtained disparity vector should be transferred to a decoding apparatus for diparity compensation of a current block. Yet, if a depth map is obtained and then transferred to the decoding apparatus, the disparity vector can be inferred from the depth map (or disparity map) without transferring the disparity vector to the decoding apparatus. In this case, it is advantageous in that the number of bits of depth information to be transferred to the decoding apparatus can be reduced. So, by deriving the disparity vector from the depth map, it is

able to provide a new disparity compensating method. Thus, in case of using a picture in a different view in the course of deriving the disparity vector from the depth map, view information for identifying a view of picture can be used.

The inter-predicted or intra-predicted pictures through the above-explained process are selected according to a prediction mode to reconstruct a current picture. In the following description, various embodiments providing an efficient decoding method of a video signal are explained.

FIG. 4 is a diagram of a hierarchical structure of level information for providing view scalability of a video signal according to one embodiment of the present invention.

Referring to FIG. 4, level information for each view can be decided by considering inter-view reference information. For instance, since it is impossible to decode a P-picture and a B-picture without an I-picture, it is able to assign 'level=0' to a base view of which inter-view picture group is the I-picture, 'level=1' to a base view of which inter-view picture group is the P-picture, and 'level=2' to a base view of which inter-view picture group is the B-picture. Yet, it is also able to decide level information randomly according to a specific standard.

Level information can be randomly decided according

to a specific standard or without a standard. For instance, in case that level information is decided based on a view, it is able to set a view V0 as a base view to view level 0, a view of pictures predicted using pictures in one view to view level 1, and a view of pictures predicted using pictures in a plurality of views to view level 2. In this case, at least one view sequence to have compatibility with a conventional decoder (e.g., H.264/AVC, MPEG-2, MPEG-4, etc.) may be needed. This base view becomes a base of multi-view coding, which may correspond to a reference view for prediction of another view. A sequence corresponding to a base view in MVC (multi-view video coding) can be configured into an independent bit stream by being encoded by a conventional sequence encoding scheme (MPEG-2, MPEG-4, H.263, H.264, etc.). A sequence corresponding to a base view is compatible with H.264/AVC or may not. Yet, a sequence in a view compatible with H.264/AVC corresponds to a base view.

As can be seen in FIG. 4, it is able to set a view V2 of pictures predicted using pictures in the view V0, a view V4 of pictures predicted using pictures in the view V2, a view V6 of pictures predicted using pictures in the view V4, and a view V7 of pictures predicted using pictures in the view V6 to view level 1. And, it is able to set a view V1

of pictures predicted using pictures in the views V0 and V2 and a view V3 predicted in the same manner, and a view V5 predicted in the same manner to view level 2. So, in case that a user's decoder is unable to view a multi-view video sequence, it decodes sequences in the view corresponding to the view level 0 only. In case that the user's decoder is restricted by profile information, it is able to decode the information of a restricted view level only. In this case, a profile means that technical elements for algorithm in a video encoding/decoding process are standardized. In particular, the profile is a set of technical elements required for decoding a bit sequence of a compressed sequence and can be a sort of a sub-standardization.

According to another embodiment of the present invention, level information may vary according to a location of a camera. For instance, assuming that views V0 and V1 are sequences obtained by a camera located in front, that views V2 and V3 are sequences located in rear, that views V4 and V5 are sequences located in left, and that views V6 and V7 are sequences located in right, it is able to set the views V0 and V1 to view level 0, the views V2 and V3 to view level 1, the views V4 and V5 to view level 2, and the views V6 and V7 to view level 3. Alternatively, level information may vary according to camera alignment.

Alternatively, level information can be randomly decided not based on a specific standard.

FIG. 5 is a diagram of a NAL-unit configuration including level information within an extension area of a NAL header according to one embodiment of the present invention.

Referring to FIG. 5, a NAL unit basically includes a NAL header and an RBSP. The NAL header includes flag information (nal_ref_idc) indicating whether a slice becoming a reference picture of the NAL unit is included and an identifier (nal_unit_type) indicating a type of the NAL unit. And, the NAL header may further include level information (view_level) indicating information for a hierarchical structure to provide view scalability.

Compressed original data is stored in the RBSP, and RBSP trailing bit is added to a last portion of the RBSP to represent a length of the RBSP as an 8-bit multiplication number. As the types of the NAL unit, there are IDR (instantaneous decoding refresh), SPS (sequence parameter set), PPS (picture parameter set), SEI (supplemental enhancement information), etc.

The NAL header includes information for a view identifier. And, a video sequence of a corresponding view level is decoded with reference to the view identifier in

the course of performing decoding according to a view level.

The NAL unit includes a NAL header 51 and a slice layer 53. The NAL header 51 includes a NAL header extension 52. And, the slice layer 53 includes a slice header 54 and a slice data 55.

The NAL header 51 includes an identifier (nal_unit_type) indicating a type of the NAL unit. For instance, the identifier indicating the NAL unit type may be an identifier for both scalable coding and multi-view video coding. In this case, the NAL header extension 52 can include flag information discriminating whether a current NAL is the NAL for the scalable video coding or the NAL for the multi-view video coding. And, the NAL header extension 52 can include extension information for the current NAL according to the flag information. For instance, in case that the current NAL is the NAL for the multi-view video coding according to the flag information, the NAL header extension 52 can include level information (view_level) indicating information for a hierarchical structure to provide view scalability.

FIG. 6 is a diagram of an overall predictive structure of a multi-view video signal according to one embodiment of the present invention to explain a concept of an inter-view picture group.

Referring to FIG. 6, T0 to T100 on a horizontal axis indicate frames according to time and S0 to S7 on a vertical axis indicate frames according to view. For instance, pictures at T0 mean frames captured by different cameras on the same time zone T0, while pictures at S0 mean sequences captured by a single camera on different time zones. And, arrows in the drawing indicate predictive directions and predictive orders of the respective pictures. For instance, a picture P0 in a view S2 on a time zone T0 is a picture predicted from I0, which becomes a reference picture of a picture P0 in a view S4 on the time zone T0. And, it becomes a reference picture of pictures B1 and B2 on time zones T4 and T2 in the view S2, respectively.

In a multi-view video decoding process, an inter-view random access may be needed. So, an access to a random view should be possible by minimizing the decoding effort. In this case, a concept of an inter-view picture group may be needed to realize an efficient access. The inter-view picture group means a coded picture in which all slices reference only slices with the same picture order count. For instance, the inter-view picture group means an coded picture that refers to slices in a different view only without referring to slices in a current view. In FIG. 6, if a picture I0 in a view S0 on a time zone T0 is an inter-

view picture group, all pictures in different views on the same time zone, i.e., the time zone T0, become inter-view picture groups. For another instance, if a picture I0 in a view S0 on a time zone T8 is an inter-view picture group, all pictures in different views on the same time zone, i.e., the time zone T8, are inter-view picture groups. Likewise, all pictures in T16, …, T96, and T100 become inter-view picture groups as well.

FIG. 7 is a diagram of a predictive structure according to an embodiment of the present invention to explain a concept of a newly defined inter-view picture group.

In an overall predictive structure of MVC, GOP can begin with an I-picture. And, the I-picture is compatible with H.264/AVC. So, all inter-view picture groups compatible with H.264/AVC can always become the I-picture. Yet, in case that the I-pictures are replaced by a P-picture, more efficient coding is enabled. In particular, more efficient coding is enabled using the predictive structure enabling GOP to begin with the P-picture compatible with H.264/AVC.

In this case, if the inter-view picture group is re-defined, all slices become encoded picture capable of referring to not only a slice in a frame on a same time

zone but also a slice in the same view on a different time zone. Yet, in case of referring to a slice on a different time zone in a same view, it can be restricted to the inter-view picture group compatible with H.264/AVC only. For instance, a P-picture on a timing point T8 in a view S0 in FIG. 6 can become a newly defined inter-view picture group. Likewise, a P-picture on a timing point T96 in a view S0 or a P-picture on a timing point T100 in a view S0 can become a newly defined inter-view picture group. And, the inter-view picture group can be defined only if it is a base view.

After the inter-view picture group has been decoded, all of the sequentially coded pictures are decoded from pictures decoded ahead of the inter-view picture group in an output order without inter-prediction.

Considering the overall coding structure of the multi-view video shown in FIG. 6 and FIG. 7, since inter-view reference information of an inter-view picture group differs from that of a non-inter-view picture group, it is necessary to discriminate the inter-view picture group and the non-inter-view picture group from each other according to the inter-view picture group identification information.

The inter-view reference information means the information capable of recognizing a predictive structure

between inter-view pictures. This can be obtained from a data area of a video signal. For instance, it can be obtained from a sequence parameter set area. And, the inter-view reference information can be recognized using the number of reference pictures and view information for the reference pictures. For instance, the number of total views is obtained and the view information for identifying each view can be then obtained based on the number of the total views. And, it is able to obtain the number of the reference pictures for a reference direction for each view. According to the number of the reference pictures, it is able to obtain the view information for each of the reference pictures. In this manner, the inter-view reference information can be obtained. And, the inter-view reference information can be recognized by discriminating an inter-view picture group and a non-inter-view picture group. This can be recognized using inter-view picture group identification information indicating whether a coded slice in a current NAL is an inter-view picture group. Details of the inter-view picture group identification information are explained with reference to FIG. 8 as follows.

FIG. 8 is a schematic block diagram of an apparatus for decoding a multi-view video using inter-view picture

group identifying information according to one embodiment of the present invention.

Referring to FIG. 8, a decoding apparatus according to one embodiment of the present invention includes a bit stream deciding unit 81, an inter-view picture group identification information obtaining unit 82, and a multi-view video decoding unit 83.

If a bit stream is inputted, the bit stream deciding unit 81 decides whether the inputted bit stream is a coded bit stream for a scalable video coding or a coded bit stream for multi-view video coding. This can be decided by flag information included in the bit stream.

The inter-view picture group identification information obtaining unit 82 is able to obtain inter-view picture group identification information if the inputted bit stream is the bit stream for a multi-view video coding as a result of the decision. If the obtained inter-view picture group identification information is 'true', it means that a coded slice of a current NAL is an inter-view picture group. If the obtained inter-view picture group identification information is 'false', it means that a coded slice of a current NAL is a non-inter-view picture group. The inter-view picture group identification information can be obtained from an extension area of a NAL

header or a slice layer area.

The multi-view video decoding unit 83 decodes a multi-view video according to the inter-view picture group identification information. According to an overall coding structure of a multi-view video sequence, inter-view reference information of an inter-view picture group differs from that of a non-inter-view picture group. So, it is able to use the inter-view picture group identification information in adding reference pictures for inter-view prediction to generate a reference picture list for example. And, it is also able to use the inter-view picture group identification information to manage the reference pictures for the inter-view prediction. Moreover, the inter-view picture group identification information is applicable to a hypothetical reference decoder.

As another example of using the inter-view picture group identification information, in case of using information in a different view for each decoding process, inter-view reference information included in a sequence parameter set is usable. In this case, information for discriminating whether a current picture is an inter-view picture group or a non-inter-view picture group, i.e., inter-view picture group identification information may be required. So, it is able to use different inter-view

reference information for each decoding process.

FIG. 9 is a flowchart of a process for generating a reference picture list according to an embodiment of the present invention.

Referring to FIG. 9, the decoded picture buffer unit 600 plays role in storing or opening previously coded pictures to perform inter-picture prediction.

First of all, pictures coded prior to a current picture are stored in the reference picture storing unit 610 to be used as reference pictures (S91).

In multi-view video coding, since some of the previously coded pictures are in a view different from that of the current picture, view information for identifying a view of a picture can be used to utilize these pictures as reference pictures. So, the decoder should obtain view information for identifying a view of a picture (S92). For instance, the view information can include 'view_id' for identifying a view of a picture.

The decoded picture buffer unit 600 needs to derive a variable used therein to generate a reference picture list. Since inter-view prediction may be required fro multi-view video coding, if a current picture refers to a picture in a different view, it may be necessary to generate a reference picture list for inter-view prediction. In this case, the

decoded picture buffer unit 600 needs to derive a variable used to generate the reference picture list for the inter-view prediction using the obtained view information (S93).

A reference picture list for temporal prediction or a reference picture list for inter-view prediction can be generated by a different method according to a slice type of a current slice (S94). For instance, if a slice type is a P/SP slice, a reference picture list 0 is generated (S95). In case that a slice type is a B-slice, a reference picture list 0 and a reference picture list 1 are generated (S96). In this case, the reference picture list 0 or 1 can include the reference picture list for the temporal prediction only or both of the reference picture list for the temporal prediction and the reference picture list for the inter-view prediction. This will be explained in detail with reference to FIG. 8 and FIG. 9 later.

The initialized reference picture list undergoes a process for assigning a smaller number to a frequently referred picture to further enhance a compression rate (S97). And, this can be called a reordering process for a reference picture list, which will be explained in detail with reference to FIGs. 12 to 19 later. The current picture is decoded using the reordered reference picture list and the decoded picture buffer unit 600 needs to manage the

decoded reference pictures to operate a buffer more efficiently (S98). The reference pictures managed by the above process are read by the inter-prediction unit 700 to be used for inter-prediction. In multi-view video coding, the inter-prediction can include inter-view prediction. In this case, the reference picture list for the inter-view prediction is usable.

Detailed examples for a method of generating a reference picture list according to a slice type are explained with reference to FIG. 10 and FIG. 11 as follows.

FIG. 10 is a diagram to explain a method of initializing a reference picture list when a current slice is a P-slice according to one embodiment of the present invention.

Referring to Fig. 10, a time is indicated by T0, T1, ..., TN, while a view is indicated by V0, V1, ..., V4. For instance, a current picture indicates a picture at a time T3 in a view V4. And, a slice type of the current picture is a P-slice. 'PN' is an abbreviation of a variable PicNum, 'LPN' is an abbreviation of a variable LongTermPicNum, and 'VN' is an abbreviation of a variable ViewNum. A numeral attached to an end portion of each of the variables indicates an index indicating a time of each picture (for PN or LPN) or a view of each picture (for VN). This is

applicable to FIG. 11 in the same manner.

A reference picture list for temporal prediction or a reference picture list for inter-view prediction can be generated in a different way according to a slice type of a current slice. For instance, a slice type in FIG. 12 is a P/SP slice. In this case, a reference picture list 0 is generated. In particular, the reference picture list 0 can include a reference picture list for temporal prediction and/or a reference picture list for inter-view prediction. In the present embodiment, it is assumed that a reference picture list includes both a reference picture list for temporal prediction and a reference picture list for inter-view prediction.

There are various methods for ordering reference pictures. For instance, reference pictures can be aligned according to in order of decoding or picture output. Alternatively, reference pictures can be aligned based on a variable derived using view information. Alternatively, reference pictures can be aligned according to inter-view reference information indicating an inter-view prediction structure.

In case of a reference picture list for temporal prediction, short-term reference pictures and long-term reference pictures can be aligned based on a decoding order.

For instance, they can be aligned according to a value of a variable PicNum or LongTermPicNum derived from a value indicating a picture identification number (e.g., frame_num or Longtermframeidx). First of all, short-term reference pictures can be initialized prior to long-tern reference pictures. An order of aligning the short-term reference pictures can be set from a reference picture having a highest value of variable PicNum to a reference picture having a lowest variable value. For instance, the short-term reference pictures can be aligned in order of PN1 having a highest variable, PN2 having an intermediate variable, and PNO having a lowest variable among PN0 to PN2. An order of aligning the long-term reference pictures can be set from a reference picture having a lowest value of variable LongTermPicNum to a reference picture having a highest variable value. For instance, the long-term reference pictures can be aligned in order of LPN0 having a highest variable and LPN1 having a lowest variable.

In case of a reference picture list for inter-view prediction, reference pictures can be aligned based on a first variable ViewNum derived using view information. In particular, reference pictures can be aligned in order of a reference picture having a highest first variable (ViewNum) value to a reference picture having a lowest first variable

(ViewNum) value. For instance, reference pictures can be aligned in order of VN3 having a highest variable, VN2, VN1, and VN0 having a lowest variable among VN0, VN1, VN2, and VN3.

Thus, both of the reference picture list for the temporal prediction and the reference picture list for the inter-view prediction can be managed as one reference picture list. Alternatively, both of the reference picture list for the temporal prediction and the reference picture list for the inter-view prediction can be managed as separate reference picture lists, respectively. In case of managing both of the reference picture list for the temporal prediction and the reference picture list for the inter-view prediction as one reference picture list, they can be initialized according to an order or simultaneously. For instance, in case of initializing both of the reference picture list for the temporal prediction and the reference picture list for the inter-view prediction according to an order, the reference picture list for the temporal prediction is preferentially initialized and the reference picture list for the inter-view prediction is then initialized in addition. This concept is applicable to FIG. 11 as well.

A case that a slice type of a current picture is a B-

slice is explained with reference to FIG. 11 as follows.

FIG. 11 is a diagram to explain a method o initializing a reference picture list when a current slice is a B-slice according to one embodiment of the present invention.

Referring to FIG. 9, in case that a slice type is a B-slice, a reference picture list 0 and a reference picture list 1 are generated. In this case, the reference picture list 0 or the reference picture list 1 can include a reference picture list for temporal prediction only or both a reference picture list for temporal prediction and a reference picture list for inter-view prediction.

In case of the reference picture list for the temporal prediction, a short-term reference picture aligning method may differ from a long-term reference picture aligning method. For instance, in case of short-term reference pictures, reference pictures can be aligned according to a picture order count (hereinafter abbreviated POC). In case of long-term reference pictures, reference pictures can be aligned according to a variable (LongtermPicNum) value. And, the short-term reference pictures can be initialized prior to the long-term reference pictures.

In order of aligning short-term reference pictures of

the reference picture list 0, reference pictures are preferentially aligned from a reference picture having a highest POC value to a reference picture having a lowest POC value among reference pictures having POC values smaller than that of a current picture, and then aligned from a reference picture having a lowest POC value to a reference picture having a highest POC value among reference pictures having POC values greater than that of the current picture. For instance, reference pictures can be preferentially aligned from PN1 having a highest POC value in reference pictures PN0 and PN1 having POC values smaller than that of a current picture to PN0, and then aligned from PN3 having a lowest POC value in reference pictures PN3 and PN4 having a POC value smaller than that of a current picture to PN4.

In order of aligning long-term reference pictures of the reference picture list 0, reference pictures are aligned from a reference picture having a lowest variable LongtermPicNum to a reference picture having a highest variable. For instance, reference pictures are aligned from LPN0 having a lowest value in LPN0 and LPN1 to LPN1 having a second lowest variable.

In case of the reference picture list for the inter-view prediction, reference pictures can be aligned based on

a first variable ViewNum derived using view information. For instance, in case of the reference picture list 0 for the inter-view prediction, reference pictures can be aligned from a reference picture having a highest first variable value among reference pictures having first variable values lower than that of a current picture to a reference picture having a lowest first variable value. The reference pictures are then aligned from a reference picture having a lowest first variable value among reference pictures having first variable values greater than that of the current picture to a reference picture having a highest first variable value. For instance, reference pictures are preferentially aligned from VN1 having a highest first variable value in VN0 and VN1 having first variable values smaller than that of a current picture to VN0 having a lowest first variable value and then aligned from VN3 having a lowest first variable value in VN3 and VN4 having first variable values greater than that of the current picture to VN4 having a highest first variable value.

In case of the reference picture list 1, the above-explained aligning method of the reference list 0 is similarly applicable.

First of all, in case of the reference picture list

for the temporal prediction, in order of aligning short-term reference pictures of the reference picture list 1, reference pictures are preferentially aligned from a reference picture having a lowest POC value to a reference picture having a highest POC value among reference pictures having POC values greater than that of a current picture and then aligned from a reference picture having a highest POC value to a reference picture having a lowest POC value among reference pictures having POC values smaller than that of the current picture. For instance, reference pictures can be preferentially aligned from PN3 having a lowest POC value in reference pictures PN3 and PN4 having POC values greater than that of a current picture to PN4 and then aligned from PN1 having a highest POC value in reference pictures PN0 and PN1 having POC values greater than that of the current picture to PN0.

In order of aligning long-term reference pictures of the reference picture list 1, reference pictures are aligned from a reference picture having a lowest variable LongtermPicNum to a reference picture having a highest variable. For instance, reference pictures are aligned from LPN0 having a lowest value in LPN0 and LPN1 to LPN1 having a lowest variable.

In case of the reference picture list for the inter-

view prediction, reference pictures can be aligned based on a first variable ViewNum derived using view information. For instance, in case of the reference picture list 1 for the inter-view prediction, reference pictures can be aligned from a reference picture having a lowest first variable value among reference pictures having first variable values greater than that of a current picture to a reference picture having a highest first variable value. The reference pictures are then aligned from a reference picture having a highest first variable value among reference pictures having first variable values smaller than that of the current picture to a reference picture having a lowest first variable value. For instance, reference pictures are preferentially aligned from VN3 having a lowest first variable value in VN3 and VN4 having first variable values greater than that of a current picture to VN4 having a highest first variable value and then aligned from VN1 having a highest first variable value in VN0 and VN1 having first variable values smaller than that of the current picture to VN0 having a lowest first variable value.

The reference picture list initialized by the above process is transferred to the reference picture list reordering unit 640. The initialized reference picture list

is then reordered for more efficient coding. The reordering process is to reduce a bit rate by assigning a small number to a reference picture having highest probability in being selected as a reference picture by operating a decoded picture buffer. Various methods of reordering a reference picture list are explained with reference to FIGs 12 to 19 as follows.

FIG. 12 is an internal block diagram of the reference picture list reordering unit 640 according to one embodiment of the present invention.

Referring to FIG. 12, the reference picture list reordering unit 640 basically includes a slice type checking unit 642, a reference picture list 0 reordering unit 643, and a reference picture list 1 reordering unit 645.

In particular, the reference picture list 0 reordering unit 643 includes a first identification information obtaining unit 643A, and a first reference index assignment changing unit 643B. And, the reference picture list 1 reordering unit 645 includes a second identification obtaining unit 645A and a second reference index assignment changing unit 645B.

The slice type checking unit 642 checks a slice type of a current slice. It is then decided whether to reorder a

reference picture list 0 and/or a reference picture list 1 according to the slice type. For instance, if a slice type of a current slice is an I-slice, both of the reference picture list 0 and the reference picture list 1 are not reordered. If a slice type of a current slice is a P-slice, the reference picture list 0 is reordered only. If a slice type of a current slice is a B-slice, both of the reference picture list 0 and the reference picture list 1 are reordered.

The reference picture list 0 reordering unit 643 is activated if flag information for executing reordering of the reference picture list 0 is 'true' and if the slice type of the current slice is not the I-slice. The first identification information obtaining unit 643A obtains identification information indicating a reference index assigning method. The first reference index assignment changing unit 643B changes a reference index assigned to each reference picture of the reference picture list 0 according to the identification information.

Likewise, the reference picture list 1 reordering unit 645 is activated if flag information for executing reordering of the reference picture list 1 is 'true' and if the slice type of the current slice is the B-slice. The second identification information obtaining unit 645A

obtains identification information indicating a reference index assigning method. The second reference index assignment changing unit 645B changes a reference index assigned to each reference picture of the reference picture list 1 according to the identification information.

So, reference picture list information used for actual inter-prediction is generated through the reference picture list 0 reordering unit 643 and the reference picture list 1 reordering unit 645.

A method of changing a reference index assigned to each reference picture by the first or second reference index assignment changing unit 643B or 645B is explained with reference to FIG. 13 as follows.

FIG. 13 is an internal block diagram of a reference index assignment changing unit 643B or 645B according to one embodiment of the present invention. In the following description, the reference picture list 0 reordering unit 643 and the reference picture list 1 reordering unit 645 shown in FIG. 12 are explained together.

Referring to FIG. 13, each of the first and second reference index assignment changing units 643B and 645B includes a reference index assignment changing unit for temporal prediction 644A, a reference index assignment changing unit for long-term reference picture 644B, a

reference index assignment changing unit for inter-view prediction 644C, and a reference index assignment change terminating unit 644D. According to identification informations obtained by the first and second identification information obtaining units 643A and 645A, parts within the first and second reference index assignment changing units 643B and 645B are activated, respectively. And, the reordering process keeps being executed until identification information for terminating the reference index assignment change is inputted.

For instance, if identification information for changing assignment of a reference index for temporal prediction is received from the first or second identification information obtaining unit 643A or 645A, the reference index assignment changing unit for temporal prediction 644A is activated. The reference index assignment changing unit for temporal prediction 644A obtains a picture number difference according to the received identification information. In this case, the picture number difference means a difference between a picture number of a current picture and a predicted picture number. And, the predicted picture number may indicate a number of a reference picture assigned right before. So, it is able to change the assignment of the reference index

using the obtained picture number difference. In this case, the picture number difference can be added/subtracted to/from the predicted picture number according to the identification information.

For another instance, if identification information for changing assignment of a reference index to a designated long-term reference picture is received, the reference index assignment changing unit for a long-term reference picture 644B is activated. The reference index assignment changing unit for a long-term reference picture 644B obtains a long-term reference picture number of a designated picture according to the identification number.

For another instance, if identification information for changing assignment of a reference index for inter-view prediction is received, the reference index assignment changing unit for inter-view prediction 644C is activated. The reference index assignment changing unit for inter-view prediction 644C obtains view information difference according to the identification information. In this case, the view information difference means a difference between a view number of a current picture and a predicted view number. And, the predicted view number may indicate a view number of a reference picture assigned right before. So, it is able to change assignment of a reference index using the

obtained view information difference. In this case, the view information difference can be added/subtracted to/from the predicted view number according to the identification information.

For another instance, if identification information for terminating a reference index assignment change is received, the reference index assignment change terminating unit 644D is activated. The reference index assignment change terminating unit 644D terminates an assignment change of a reference index according to the received identification information. So, the reference picture list reordering unit 640 generates reference picture list information.

Thus, reference pictures used for inter-view prediction can be managed together with reference pictures used for temporal prediction. Alternatively, reference pictures used for inter-view prediction can be managed separate from reference pictures used for temporal prediction. For this, new informations for managing the reference pictures used for the inter-view prediction may be required. This will be explained with reference to FIGs. 15 to 19 later.

Details of the reference index assignment changing unit for inter-view prediction 644C are explained with

reference to FIG. 14 as follows.

FIG. 14 is a diagram to explain a process for reordering a reference picture list using view information according to one embodiment of the present invention.

Referring to FIG. 14, if a view number VN of a current picture is 3, if a size of a decoded picture buffer DPBsize is 4, and if a slice type of a current slice is a P-slice, a reordering process for a reference picture list 0 is explained as follows.

First of all, an initially predicted view number is ' 3' that is the view number of the current picture. And, an initial alignment of the reference picture list 0 for inter-view prediction is ' 4, 5, 6, 2' (①). In this case, if identification information for changing assignment of a reference index for inter-view prediction by subtracting a view information difference is received, '1' is obtained as the view information difference according to the received identification information. A newly predicted view number (=2) is calculated by subtracting the view information difference (=1) from the predicted view number (=3). In particular, a first index of the reference picture list 0 for the inter-view prediction is assigned to a reference picture having the view number 2. And, a picture previously assigned to the first index can be moved to a most rear

part of the reference picture list 0. So, the reordered reference picture list 0 is ' 2, 5, 6, 4' (②). Subsequently, if identification information for changing assignment of a reference index for inter-view prediction by subtracting the view information difference is received, '-2' is obtained as the view information difference according to the identification information. A newly predicted view number (=4) is then calculated by subtracting the view information difference (=-2) from the predicted view number (=2). In particular, a second index of the reference picture list 0 for the inter-view prediction is assigned to a reference picture having a view number 4. Hence, the reordered reference picture list 0 is ' 2, 4, 6, 5' (③). Subsequently, if identification information for terminating the reference index assignment change is received, the reference picture list 0 having the reordered reference picture list 0 as an end is generated according to the received identification information (④). Hence, the order of the finally generated reference picture list 0 for the inter-view prediction is ' 2, 4, 6, 5' .

For another instance of reordering the rest of the pictures after the first index of the reference picture list 0 for the inter-view prediction has been assigned, a picture assigned to each index can be moved to a position

right behind that of the corresponding picture. In particular, a second index is assigned to a picture having a view number 4, a third index is assigned to a picture (view number 5) to which the second index was assigned, and a fourth index is assigned to a picture (view number 6) to which the third index was assigned. Hence, the reordered reference picture list 0 becomes '2, 4, 5, 6'. And, a subsequent reordering process can be executed in the same manner.

The reference picture list generated by the above-explained process is used for inter-prediction. Both of the reference picture list for the inter-view prediction and the reference picture list for the temporal prediction can be managed as one reference picture list. Alternatively, each of the reference picture list for the inter-view prediction and the reference picture list for the temporal prediction can be managed as a separate reference picture list. This is explained with reference to FIGs. 15 to 19 as follows.

FIG. 15 is an internal block diagram of a reference picture list reordering unit 640 according to another embodiment of the present invention.

Referring to FIG. 15, in order to manage a reference picture list for inter-view prediction as a separate

reference picture list, new informations may be needed. For instance, a reference picture list for temporal prediction is reordered, and a reference picture list for inter-view prediction is then reordered in some cases.

The reference picture list reordering unit 640 basically includes a reference picture list reordering unit for temporal prediction 910, a NAL type checking unit 960, and a reference picture list reordering unit for inter-view prediction 970.

The reference picture list reordering unit for temporal prediction 910 includes a slice type checking unit 642, a third identification information obtaining unit 920, a third reference index assignment changing unit 930, a fourth identification information obtaining unit 940, and a fourth reference index assignment changing unit 950. The third reference index assignment changing unit 930 includes a reference index assignment changing unit for temporal prediction 930A, a reference index assignment changing unit for a long-term reference picture 930B, and a reference index assignment change terminating unit 930C. Likewise, the fourth reference index assignment changing unit 950 includes a reference index assignment changing unit for temporal prediction 950A, a reference index assignment changing unit for long-term reference picture 950B, and a

reference index assignment change terminating unit 950C.

The reference picture list reordering unit for temporal prediction 910 reorders reference pictures used for temporal prediction. Operations of the reference picture list reordering unit for temporal prediction 910 are identical to those of the aforesaid reference picture list reordering unit 640 shown in FIG. 10 except informations for the reference pictures for the inter-view prediction. So, details of the reference picture list reordering unit for temporal prediction 910 are omitted in the following description.

The NAL type checking unit 960 checks a NAL type of a received bit stream. If the NAL type is a NAL for multi-view video coding, reference pictures used for the inter-view prediction are reordered by the reference picture list reordering unit for temporal prediction 970. The generated reference picture list for the inter-view prediction are used for inter-prediction together with the reference picture list generated by the reference picture list reordering unit for temporal prediction 910. Yet, if the NAL type is not the NAL for the multi-view video coding, the reference picture list for the inter-view prediction is not reordered. In this case, a reference picture list for temporal prediction is generated only. And, the inter-view

prediction reference picture list reordering unit 970 reorders reference pictures used for inter-view prediction. This is explained in detail with reference to FIG. 16 as follows.

FIG. 16 is an internal block diagram of the reference picture list reordering unit 970 for inter-view prediction according to one embodiment of the present invention.

Referring to FIG. 16, the reference picture list reordering unit for inter-view prediction 970 includes a slice type checking unit 642, a fifth identification information obtaining unit 971, a fifth reference index assignment changing unit 972, a sixth identification information obtaining unit 973, and a sixth reference index assignment changing unit 974.

The slice type checking unit 642 checks a slice type of a current slice. If so, it is then decided whether to execute reordering of a reference picture list 0 and/or a reference picture list 1 according to the slice type. Details of the slice type checking unit 642 can be inferred from FIG. 10, which are omitted in the following description.

Each of the fifth and sixth identification information obtaining units 971 and 973 obtains identification information indicating a reference index

assigning method. And, each of the fifth and sixth reference index assignment changing units 972 and 974 changes a reference index assigned to each reference picture of the reference picture list 0 and/or 1. In this case, the reference index can mean a view number of a reference picture only. And, the identification information indicating the reference index assigning method may be flag information. For instance, if the flag information is true, an assignment of a view number is changed. If the flag information is false, a reordering process of a view number can be terminated. If the flag information is true, each of the fifth and sixth reference index assignment changing units 972 and 974 can obtain a view number difference according to the flag information. In this case, the view number difference means a difference between a view number of a current picture and a view number of a predicted picture. And, the view number of the predicted picture may mean a view number of a reference picture assigned right before. It is then able to change view number assignment using the view number difference. In this case, the view number difference can be added/subtracted to/from the view number of the predicted picture according to the identification information.

Thus, to manage the reference picture list for the

inter-view prediction as a separate reference picture list, it is necessary to newly define a syntax structure. As one embodiment of the contents explained in FIG. 15 and FIG. 16, the syntax is explained with reference to FIG. 17, FIG. 18, and FIG. 19 as follows.

FIG. 17 and FIG. 18 are diagrams of syntax for reference picture list reordering according to one embodiment of the present invention.

Referring to FIG. 17, an operation of the reference picture list reordering unit for the temporal prediction 910 shown in FIG. 15 is represented as syntax. Compared to the blocks shown in FIG. 15, the slice type checking unit 642 corresponds to S1 and S6 and the fourth identification information obtaining unit 940 corresponds to S7. The internal blocks of the third reference index assignment changing unit 930 correspond to S3, S4, and S5, respectively. And, the internal blocks of the fourth reference index assignment changing unit 950 correspond to S8, S9, and S10, respectively.

Referring to FIG. 18, operations of the NAL type checking unit 960 and the inter-view reference picture list reordering unit 970 are represented as syntax. Compared to the respective blocks shown in FIG. 15 and FIG. 16, the NAL type checking unit 960 corresponds to S11, the slice type

checking unit 642 corresponds to S13 and S16, the fifth identification information obtaining unit 971 corresponds to S14, and the sixth identification information obtaining unit 973 corresponds to S17. The fifth reference index assignment changing unit 972 corresponds to S15 and the sixth reference index assignment changing unit 974 corresponds to S18.

FIG. 19 is a diagram of syntax for reference picture list reordering according to another embodiment of the present invention.

Referring to FIG. 19, operations of the NAL type checking unit 960 and the inter-view reference picture list reordering unit 970 are represented as syntax. Compared to the respective blocks shown in FIG. 15 and FIG. 16, the NAL type checking unit 960 corresponds to S21, the slice type checking unit 642 corresponds to S22 and S25, the fifth identification information obtaining unit 971 corresponds to S23, and the sixth identification information obtaining unit 973 corresponds to S26. The fifth reference index assignment changing unit 972 corresponds to S24 and the sixth reference index assignment changing unit 974 corresponds to S27.

As mentioned in the foregoing description, the reference picture list for the inter-view prediction can be

used by the inter-prediction unit 700 and is usable for performing illumination compensation as well. The illumination compensation is applicable in the course of performing motion estimation/motion compensation. In case that a current picture uses a reference picture in a different view, it is able to perform the illumination compensation more efficiently using the reference picture list for the inter-view prediction. The illumination compensations according to embodiments of the present invention are explained as follows.

FIG. 20 is a diagram for a process for obtaining a illumination difference value of a current block according to one embodiment of the present invention.

Illumination compensation means a process for decoding an adaptively motion compensated video signal according to illumination change. And, it is applicable to a predictive structure of a video signal, for example, inter-view prediction, intra-view prediction, and the like.

Illumination compensation means a process for decoding a video signal using a illumination difference residual and a illumination difference prediction value corresponding to a block to be decoded. In this case, the illumination difference prediction value can be obtained from a neighboring block of a current block. A process for

obtaining a illumination difference prediction value from the neighboring block can be decided using reference information for the neighbor block, and a sequence and direction can be taken into consideration in the course of searching neighbor blocks. The neighboring block means an already decoded block and also means a block decoded by considering redundancy within the same picture for a view or time or a sequence decoded by considering redundancy within different pictures.

In comparing similarities between a current block and a candidate reference block, an illumination difference between the two blocks should be taken into consideration. In order to compensate for the illumination difference, new motion estimation/compensation is executed. New SAD can be found using Formula 1.

[Formula 1]

$$M_{cur} = \frac{1}{S \times T} \sum_{i=m}^{m+S-1} \sum_{j=n}^{n+T-1} f(i, j)$$

$$M_{ref}(p, q) = \frac{1}{S \times T} \sum_{i=p}^{p+S-1} \sum_{j=q}^{q+T-1} r(i, j)$$

[Formula 2]

$$NewSAD(x, y) = \sum_{i=m}^{m+S-1} \sum_{j=n}^{n+T-1} \left| \left(f(i, j) - M_{cur} \right) - \left(r(i + x, j + y) - M_{ref}(m + x, n + y) \right) \right|$$

In this case, 'Mcurr' indicates an average pixel

value of a current block and 'Mref' indicates an average pixel value of a reference block. 'f(i,j)' indicates a pixel value of a current block and 'r(i+x, j+y)' indicates a pixel value of a reference block. By performing motion estimation based on the new SAD according to the Formula 2, it is able to obtain an average pixel difference value between the current block and the reference block. And, the obtained average pixel difference value may be called an illumination difference value (IC_offset).

In case of performing motion estimation to which illumination compensation is applied, an illumination difference value and a motion vector are generated. And, the illumination compensation is executed according to Formula 3 using the illumination difference value and the motion vector.

[Formula 3]

$$NewR(i,j) = \{f(i,j) - M_{cur}\} - \{r(i+x', j+y') - M_{ref}(m+x', n+y')\}$$
$$= \{f(i,j) - r(i+x', j+y')\} - \{M_{cur} - M_{ref}(m+x', n+y')\}$$
$$= \{f(i,j) - r(i+x', j+y')\} - IC_offset$$

In this case, NewR(i,j) indicates an illumination-compensated error value (residual) and (x′ , y′) indicates a motion vector.

An illumination difference value (Mcurr- Mref) should be transferred to the decoding unit. The decoding unit

carries out the illumination compensation in the following manner.

[Formula 4]

$$f'(i,j) = \{NewR''(x',y',i,j) + r(i+x', j+y')\} + \{M_{cur} - M_{ref}(m+x', n+y')\}$$
$$= \{NewR''(x',y',i,j) + r(i+x', j+y')\} + IC_offset$$

In Formula 4, NewR″ (i,j) indicates a reconstructed illumination-compensated error value (residual) and f′ (i,j) indicates a pixel value of a reconstructed current block.

In order to reconstruct a current block, an illumination difference value should be transferred to the decoding unit. And, the illumination difference value can be predicted from information of neighboring blocks. In order to further reduce a bit number to code the illumination difference value, it is able to send a difference value (RIC_offset) between the illumination difference value of the current block (IC_offset) and the illumination difference value of the neighboring block (predIC_offset) only. This is represented as Formula 5.

[Formula 5]

RIC_offset = IC_offset − predIC_offset

FIG. 21 is a flowchart of a process for performing illumination compensation of a current block according to an embodiment of the present invention.

Referring t FIG. 21, first of all, an illumination difference value of a neighboring block indicating an average pixel difference value between the neighboring block of a current block and a block referred to by the neighbor block is extracted from a video signal (S2110).

Subsequently, an illumination difference prediction value for illumination compensation of the current block is obtained using the illumination difference value (S2120). So, it is able to reconstruct an illumination difference value of the current block using the obtained illumination difference prediction value.

In obtaining the illumination difference prediction value, it is able to use various methods. For instance, before the illumination difference value of the current block is predicted from the illumination difference value of the neighboring block, it is checked whether a reference index of the current block is equal to that of the neighboring block. It is then able to decide what kind of a neighboring block or a value will be used according to a result of the checking. For another instance, in obtaining the illumination difference prediction value, flag information (IC_flag) indicating whether to execute an illumination compensation of the current block can be used. And, flag information for the current block can be

predicted using the information of the neighboring blocks as well. For another instance, it is able to obtain the illumination difference prediction value using both of the reference index checking method and the flag information predicting method. These are explained in detail with reference to FIGs. 22 to 24 as follows.

FIG. 22 is a block diagram of a process for obtaining an illumination difference prediction value of a current block using information for a neighbor block according to one embodiment of the present invention.

Referring to FIG. 22, it is able to use information for a neighboring block in obtaining an illumination difference prediction value of a current block. In the present disclosure, a block can include a macroblock or a sub-macroblock. For instance, it is able to predict an illumination difference value of the current block using an illumination difference value of the neighboring block. Prior to this, it is checked whether a reference index of the current block is equal to that of the neighboring block. According to a result of the checking, it is then able to decide what kind of a neighboring block or a value will be used. In FIG. 22, 'refIdxLX' indicates a reference index of a current block, 'refIdxLXN' indicates a reference index of a block-N. In this case, 'N' is a mark of a block neighbor

to the current block and indicates A, B, or C. And, 'PredIC_offsetN' indicates an illumination difference value for illumination compensation of a neighbor block-N. If it is unable to use a block-C that is located at an upper right end of the current block, it is able to use a block-D instead of the block-C. In particular, information for the block-D is usable as information for the block-C. If it is unable to use both of the block-B and the block-C, it is able to use a block-A instead. Namely, it is able to use the information for the block-A as the information for the block-B or the block-C.

For another instance, in obtaining the illumination difference prediction value, it is able to use flag information (IC_flag) indicating whether to execute an illumination compensation of the current block. Alternatively, it is able to use both of the reference index checking method and the flag information predicting method in obtaining the illumination difference prediction value. In this case, if the flag information for the neighbor block indicates that the illumination compensation is not executed, i.e., if IC_falg==0, the illumination difference value 'PredIC_offsetN' of the neighbor block is set to 0.

FIG. 23 is a flowchart of a process for performing

illumination compensation using information for a neighbor block according to one embodiment of the present invention.

Referring to FIG. 23, the decoding unit extracts an average pixel value of a reference block, a reference index of a current block, a reference index of the reference block, and the like from a video signal and is then able to obtain an illumination difference prediction value of the current block using the extracted information. The decoding unit obtains a difference value (illumination difference residual) between an illumination difference value of the current block and the illumination difference prediction value and is then able to reconstruct an illumination difference value of the current block using the obtained illumination difference residual and the illumination difference prediction value. In this case, it is able to use information for a neighbor block to obtain the illumination difference prediction value of the current block. For instance, it is able to predict an illumination difference value of the current block using the illumination difference value of the neighbor block. Prior to this, it is checked whether a reference index of the current block is equal to that of the neighbor block. According to a result of the checking, it is then able to decide what kind of a neighboring block or a value will be

used.

In particular, an illumination difference value of a neighbor block indicating an average pixel difference value between the neighbor block of a current block and a block referred to by the neighbor block is extracted from a video signal (S2310).

Subsequently, it is checked whether a reference index of the current block is equal to a reference index of one of a plurality of neighbor blocks (S2320).

As a result of the checking step S2320, if there exists at least one neighbor block having the same reference index as that of the current block, it is checked whether there exist one corresponding neighbor block or not (S2325).

As a result of the checking step S2325, if there exists only one neighbor block having the same reference index of the current block, an illumination difference value of the neighbor block having the same reference index of the current block is assigned to an illumination difference prediction value of the current block (S2330). In particular, it is 'PredIC_offset = PredIC_offsetN'.

If the neighbor block having the same reference index as that of the current block fails to exist as a result of the checking step S2320 or if there exist at

least tow neighbor blocks having the same reference index as that of the current block as a result of the checking step S2325, a median of illumination difference values (PredIC_offsetN, N = A, B, or C) of the neighbor blocks is assigned to an illumination difference prediction value of the current block (S650). In particular, it is 'PredIC_offset = Median(PredIC_offsetA, PredIC_offsetB, PredIC_offsetC)'.

FIG. 24 is a flowchart of a process for performing illumination compensation using information for a neighbor block according to another embodiment of the present invention.

Referring to FIG. 24, a decoding unit has to reconstruct an illumination difference value of a current block to carry out illumination compensation. In this case, it is able to use information for a neighbor block to obtain an illumination difference prediction value of the current block. For instance, it is able to predict an illumination difference value of the current block using the illumination difference value of the neighbor block. Prior to this, it is checked whether a reference index of the current block is equal to that of the neighbor block. According to a result of the checking, it is then able to decide what kind of a neighboring block or a value will be

used.

In particular, an illumination difference value of a neighbor block indicating an average pixel difference value between the neighbor block of a current block and a block referred to by the neighbor block is extracted from a video signal (S2410).

Subsequently, it is checked whether a reference index of the current block is equal to a reference index of one of a plurality of neighbor blocks (S2420).

As a result of the checking step S720, if there exists at least one neighbor block having the same reference index as that of the current block, it is checked whether there exist one corresponding neighbor block or not (S2430).

As a result of the checking step S2430, if there exists only one neighbor block having the same reference index as that of the current block, an illumination difference value of the neighbor block having the same reference index as that of the current block is assigned to an illumination difference prediction value of the current block (S2440). In particular, it is 'PredIC_offset = PredIC_offsetN'.

If the neighbor block having the same reference index as that of the current block fails to exist as a

result of the checking step S2420, the illumination difference prediction value of the current block is set to 0 (S2460). In particular, it is 'PredIC_offset = 0'.

If there exist at least two neighbor blocks having the same reference index as that of the current block as a result of the checking step S2430, the neighbor block having a reference index different from that of the current block is set to 0 and a median of illumination difference values of the neighbor blocks including the value set to 0 is assigned to the illumination difference prediction value of the current block (S2450). In particular, it is 'PredIC_offset = Median(PredIC_offsetA, PredIC_offsetB, PredIC_offsetC)'. Yet, in case that there exists the neighbor block having the reference index different from that of the current block, the value '0' can be included in PredIC_offsetA, PredIC_offsetB, or PredIC_offsetC.

Meanwhile, view information for identifying a view of a picture and a reference picture list for inter-view prediction are applicable to synthesizing a picture in a virtual view. In a process for synthesizing a picture in a virtual view, a picture in a different view may be referred to. So, if the view information and the reference picture list for the inter-view prediction are used, it is able to synthesize a picture in a virtual view more efficiently. In

the following description, methods of synthesizing a picture in a virtual view according to embodiments of the present invention are explained.

FIG. 25 is a block diagram of a process for predicting a current picture using a picture in a virtual view according to one embodiment of the present invention.

Referring to FIG. 25, in performing inter-view prediction in multi-view video coding, it is able to predict a current picture using a picture in a view different from that of the current view as a reference picture. Yet, a picture in a virtual view is obtained using pictures in a view neighbor to that of a current picture and the current picture is then predicted using the obtained picture in the virtual view. If so, the prediction can be more accurately performed. In this case, a view identifier indicating a view of a picture can be used to utilize pictures in neighbor views or pictures in a specific view. In case that the virtual view is generated, there must exist specific syntax for indicating whether to generate the virtual view. If the syntax indicates that the virtual view shall be generated, it is able to generate the virtual view using the view identifier. The pictures in the virtual view obtained by the view synthesis prediction unit 740 are usable as reference pictures. In this case, the

view identifier can be assigned to the pictures in the virtual view. In a process for performing motion vector prediction to transfer a motion vector, neighbor blocks of a current block can refer to the pictures obtained by the view synthesis prediction unit 740. In this case, to use the picture in the virtual view as the reference picture, a view identifier indicating a view of a picture can be utilized.

FIG. 26 is a flowchart of a process for synthesizing a picture of a virtual view in performing inter-view prediction in MVC according to an embodiment of the present invention.

Referring to FIG. 26, a picture in a virtual view is synthesized using pictures in a view neighbor to that of a current picture. The current picture is then predicted using the synthesized picture in the virtual view. If so, it is able to achieve more accurate prediction. In case that a picture in a virtual view is synthesized, there exists specific syntax indicating whether to execute a prediction of a current picture by synthesizing the picture in the virtual view. If it is decided whether to execute the prediction of the current picture, more efficient coding is possible. The specific syntax is defined as an inter-view synthesis prediction identifier, which is

explained as follows. For instance, a picture in a virtual view is synthesized by a slice layer to define 'view_synthesize_pred_flag' indicating whether to execute a prediction of a current picture. And, a picture in a virtual view is synthesized by a macroblock layer to define 'view_syn_pred_flag' indicating whether to execute a prediction of a current picture. If 'view_synthesize_pred_flag = 1', a current slice synthesizes a slice in a virtual view using a slice in a view neighbor to that of the current slice. It is then able to predict the current slice using the synthesized slice. If 'view_synthesize_pred_flag = 0', a slice in a virtual view is not synthesized. Likewise, if 'view_syn_pred_flag = 1', a current macroblock synthesizes a macroblock in a virtual view using a macroblock in a view neighbor to that of the current macroblock. It is then able to predict the current macroblock using the synthesized macroblock. If 'view_syn_pred_flag = 0', a macroblock in a virtual view is not synthesized. Hence, in the present invention, the inter-view synthesis prediction identifier indicating whether to obtain a picture in a virtual view is extracted from a video signal. It is then able to obtain the picture in the virtual view using the inter-view synthesis prediction identifier.

As mentioned in the foregoing description, view information for identifying a view of a picture and a reference picture list for inter-view prediction can be used by the inter-prediction unit 700. And, they can be used in performing weighted prediction as well. The weighted prediction is applicable to a process for performing motion compensation. In doing so, if a current picture uses a reference picture in a different view, it is able to perform the weighted prediction more efficiently using the view information and the reference picture list for the inter-view prediction. Weighted prediction methods according to embodiments of the present invention are explained as follows.

FIG. 27 is a flowchart of a method of executing weighted prediction according to a slice type in video signal coding according to the present invention.

Referring to FIG. 27, weighted prediction is a method of scaling a sample of motion compensated prediction data within a P-slice or B-slice macroblock. A weighted prediction method includes an explicit mode for performing weighted prediction for a current picture using a weighted coefficient information obtained from information for reference pictures and an implicit mode for performing weighted prediction for a current picture using a weighted

coefficient information obtained from information for a distance between the current picture and one of reference pictures. The weighted prediction method can be differently applied according to a slice type of a current macroblock. For instance, in the explicit mode, the weighted coefficient information can be varied according to whether a current macroblock, on which weighted prediction is performed, is a macroblock of a P-slice or a macroblock of a B-slice. And, the weighted coefficient of the explicit mode can be decided by an encoder and can be transferred by being included in a slice header. On the other hand, in the implicit mode, a weighted coefficient can be obtained based on a relatively temporal position of List 0 and List 1. For instance, if a reference picture is temporarily close to a current picture, a great weighted coefficient is applicable. If a reference picture is temporarily distant from to a current picture, a small weighted coefficient is applicable.

First of all, a slice type of a macroblock to apply weighted prediction thereto is extracted from a video signal (S2710).

Subsequently, weighted prediction can be performed on a macroblock according to the extracted slice type (S2720).

In this case, the slice type can include a

macroblock to which inter-view prediction is applied. The inter-view prediction means that a current picture is predicted using information for a picture in a view different from that of the current picture. For instance, the slice type can include a macroblock to which temporal prediction for performing prediction using information for a picture in a same view as that of a current picture is applied, a macroblock to which the inter-view prediction is applied, and a macroblock to which both of the temporal prediction and the inter-view prediction are applied. And, the slice type can include a macroblock to which temporal prediction is applied only, a macroblock to which inter-view prediction is applied only, or a macroblock to which both of the temporal prediction and the inter-view prediction are applied. Moreover, the slice type can include two of the macroblock types or all of the three macroblock types. This will be explained in detail with reference to FIG. 28 later. Thus, in case that a slice type including a inter-view prediction applied macroblock is extracted from a video signal, weighted prediction is performed using information for a picture in a view different from that of a current picture. In doing so, a view identifier for identifying a view of a picture can be utilized to use information for a picture in a different

view.

FIG. 28 is a diagram of macroblock types allowable in a slice type in video signal coding according to one embodiment of the present invention.

Referring to FIG. 28, if a P-slice type by inter-view prediction is defined as VP (View_P), an intra-macroblock I, a macroblock P predicted from one picture in a current view, or a macroblock VP predicted from one picture in a different view is allowable for the P-slice type by the inter-view prediction (2810).

In case that a B-slice type by inter-view prediction is defined as VB (View_B), a macroblock P or B predicted from at least one picture in a current view or a macroblock VP or VB predicted from at least one picture in a different view is allowable (2820).

In case that a slice type, on which prediction is performed using temporal prediction, inter-view prediction, or both of the temporal prediction and the inter-view prediction, is defined as 'Mixed', an intra-macroblock I, a macroblock P or B predicted from at least one picture in a current view, a macroblock VP or VB predicted from at least one picture in a different view, or a macroblock 'Mixed' predicted using both of the picture in the current view and the picture in the different view is allowable for the

mixed slice type (2830). In this case, in order to use the picture in the different view, it is able to use a view identifier for identifying a view of a picture.

FIG. 29 and FIG. 30 are diagrams of syntax for executing weighted prediction according to a newly defined slice type according to one embodiment of the present invention.

As mentioned in the foregoing description of FIG. 28, if the slice type is decided as VP, VB, or Mixed, the syntax for performing the conventional weighted prediction (e.g., H.264) can be modified into FIG. 29 or FIG. 30.

For instance, if a slice type is P-slice by temporal prediction, a part ' if(slice_type != VP || slice_type != VB)' is added (2910).

If a slice type is a B-slice by temporal prediction, the if-statement can be modified into ' if(slice_type == B || slice_type == Mixed)' (2920).

By newly defining a VP slice type and a VB slice type, a format similar to FIG. 29 can be newly added (2930, 2940). In this case, since information for a view is added, syntax elements include 'view' parts, respectively. For example, there is 'luma_log2_view_weight_denom, chroma_log2_view_weight_denom'.

FIG. 31 is a flowchart of a method of executing

weighted prediction using flag information indicating whether to execute inter-view weighted prediction in video signal coding according to the present invention.

Referring to FIG. 31, in video signal coding to which the present invention is applied, in case of using flag information indicating whether weighted prediction will be executed, more efficient coding is enabled.

The flag information can be defined based on a slice type. For instance, there can exist flag information indicating whether weighted prediction will be applied to a P-slice or a SP-slice or flag information indicating whether weighted prediction will be applied to a B-slice.

In particular, the flag information can be defined as 'weighted_pred_flag' or 'weighted_bipred_idc'. If 'weighted_pred_flag = 0', it indicates that weighted prediction is not applied to the P-slice and the SP-slice. If 'weighted_pred_flag = 1', it indicates that weighted prediction is applied to the P-slice and the SP-slice. If 'weighted_bipred_idc = 0', it indicates that default weighted prediction is applied to the B-slice. If 'weighted_bipred_idc = 1', it indicates that explicit weighted prediction is applied to the B-slice. If 'weighted_bipred_idc = 2', it indicates that implicit weighted prediction is applied to the B-slice.

In multi-view video coding, flag information indicating whether weighted prediction will be executed using information for an inter-view picture can be defined based on a slice type.

First of all, a slice type and flag information indicating whether inter-view weighted prediction will be executed are extracted from a video signal (S3110, S3120). In this case, the slice type can include a macroblock to which temporal prediction for performing prediction using information for a picture in a same view as that of a current picture is applied and a macroblock to which inter-view prediction for performing prediction using information for a picture in a view different from that of a current picture is applied.

It is then able to decide a weighted prediction mode based on the extracted slice type and the extracted flag information (S3130).

Subsequently, it is able to perform weighted prediction according to the decided weighted prediction mode (S3140). In this case, the flag information can include flag information indicating whether weighted prediction will be executed using information for a picture in a view different from that of a current picture as well as the aforesaid 'weighted_pred_flag' and

'weighted_bipred_flag'. This will be explained in detail with reference to FIG. 32 later.

Hence, in case that a slice type of a current macroblock is a slice type including a macroblock to which inter-view prediction is applied, more efficient coding is enabled rather than a case of using flag information indicating whether weighted prediction will be executed using information for a picture in a different view.

FIG. 32 is a diagram to explain a weight predicting method according to flag information indicating whether to execute weighted prediction using information for a picture in a view different from that if a current picture according to one embodiment of the present invention.

Referring to FIG. 32, for example, flag information indicating whether weighted prediction will be executed using information for a picture in a view different from that of a current picture can be defined as 'view_weighted_pred_flag' or 'view_weighted_bipred_flag'.

If 'view_weighted_pred_flag = 0', it indicates that weighted prediction is not applied to a VP-slice. If 'view_weighted_pred_flag = 1', explicit weighted prediction is applied to a VP-slice. If 'view_weighted_bipred_flag = 0', it indicates that default weighted prediction is applied to a VB-slice. If 'view_weighted_bipred_flag = 1',

it indicates that explicit weighted prediction is applied to a VB-slice. If 'view_weighted_bipred_flag = 2', it indicates that implicit default weighted prediction is applied to a VB-slice.

In case that implicit weighted prediction is applied to a VB-slice, a weight coefficient can be obtained from a relative distance between a current view and a different view. In case that implicit weighted prediction is applied to a VB-slice, weighted prediction can be performed using a view identifier identifying a view of a picture or a picture order count (POC) rendered by considering discrimination of each view.

The above flag informations can be included in a picture parameter set (PPS). In this case, the picture parameter set (PPS) means header information indicating an encoding mode of all pictures (e.g., entropy encoding mode, quantization parameter initial value by picture unit, etc.). Yet, the picture parameter set is not attached to all of the pictures. If a picture parameter set does not exist, a picture parameter set existing right before is used as header information.

FIG. 33 is a diagram of syntax for executing weighted prediction according to newly defined flag information according to one embodiment of the present

invention.

Referring to FIG. 33, in multi-view video coding to which the present invention is applied, in case that a slice type including a macroblock applied to inter-view prediction and flag information indicating whether weighted prediction will be executed using information for a picture in a view different from that of a current picture are defined, it is necessary to decide what kind of weighted prediction will be executed according to a slice type.

For instance, if a slice type, as shown in FIG. 33, extracted from a video signal is a P-slice or a SP-slice, weighted prediction can be executed if 'weighted_pred_flag = 1'. In case that a slice type is a B-slice, weighted prediction can be executed if 'weighted_bipred_flag = 1'. In case that a slice type is a VP-slice, weighted prediction can be executed if 'view_weighted_pred_flag = 1'. In case that a slice type is a VB-slice, weighted prediction can be executed if 'view_weighted_bipred_flag = 1'.

FIG. 34 is a flowchart of a method of executing weighted prediction according to a NAL (network abstraction layer) unit according to an embodiment of the present invention.

Referring to FIG. 34, first of all, a NAL unit type

(nal_unit_type) is extracted from a video signal (S910). In this case, the NAL unit type means an identifier indicating a type of a NAL unit. For instance, if 'nal_unit_type = 5', a NAL unit is a slice of an IDR picture. And, the IDR (instantaneous decoding refresh) picture means a head picture of a video sequence.

Subsequently, it is checked whether the extracted NAL unit type is a NAL unit type for multi-view video coding (S3420).

If the NAL unit type is the NAL unit type for multi-view video coding, weighted prediction is carried out using information for a picture in a view different from that of a current picture (S3430). The NAL unit type can be a NAL unit type applicable to both scalable video coding and multi-view video coding or a NAL unit type for multi-view video coding only. Thus, if the NAL unit type is for multi-view video coding, the weighted prediction should be executed using the information for the picture in the view different from that of the current picture. So, it is necessary to define new syntax. This will be explained in detail with reference to FIG. 35 and FIG. 36 as follows.

FIG. 35 and FIG. 36 are diagrams of syntax for executing weighted prediction in case that a NAL unit type is for multi-view video coding according to one embodiment

of the present invention.

First of all, if a NAL unit type is a NAL unit type for multi-view video coding, syntax for executing conventional weighted prediction (e.g., H.264) can be modified into the syntax shown in FIG. 35 or FIG. 36. For instance, a reference number 3510 indicates a syntax part for performing conventional weighted prediction and a reference number 3520 indicates a syntax part for performing weighted prediction in multi-view video coding. So, the weighted prediction is performed by the syntax part 3520 only if the NAL unit type is the NAL unit type for multi-view video coding. In this case, since information for a view is added, each syntax element includes a 'view' portion. For instance, there is 'luma_view_log2_weight_denom, chroma_view_log2_weight_denom' or the like. And, a reference number 3530 in FIG. 36 indicates a syntax part for performing conventional weighted prediction and a reference number 3540 in FIG. 36 indicates a syntax part indicates a syntax part for performing weighted prediction in multi-view video coding. So, the weighted prediction is performed by the syntax part 3540 only if the NAL unit type is the NAL unit type for multi-view video coding. Likewise, since information for a view is added, each syntax element

includes a 'view' portion. For instance, there is 'luma_view_weight_l1_flag, chroma_view_weight_l1_flag' or the like. Thus, if a NAL unit type for multi-view video coding is defined, more efficient coding is enabled un a manner of performing weighted prediction using information for a picture in a view different from that of a current picture.

FIG. 37 is a block diagram of an apparatus for decoding a video signal according to an embodiment of the present invention.

Referring to FIG. 37, an apparatus for decoding a video signal according to the present invention includes a slice type extracting unit 3710, a prediction mode extracting unit 3720 and a decoding unit 3730.

FIG. 38 is a flowchart of a method of decoding a video signal in the decoding apparatus shown in FIG. 37 according to one embodiment of the present invention.

Referring to FIG. 38, a method of decoding a video signal according to one embodiment of the present invention includes a step S3810 of extracting a slice type and a macroblock prediction mode, and a step S3820 of decoding a current macroblock according to the slice type and/or macroblock prediction mode.

First, a prediction scheme used by an embodiment of

the present invention is explained to help in the understanding of the present invention. The prediction scheme may be classified into an intra-view prediction (e.g., prediction between pictures in a same view) and an inter-view prediction (e.g., prediction between pictures in different views). And, the intra-view prediction can be the same prediction scheme as a general temporal prediction.

According to the present invention, the slice type extracting unit 3710 extracts a slice type of a slice including a current macroblock (S3810).

In this case, a slice type field (slice_type) indicating a slice type for intra-view prediction and/or a slice type field (view_slice_type) indicating a slice type for inter-view prediction may be provided as part of the video signal syntax to provide the slice type. This will be described in greater deal below with respect to FIGs. 6(a) and 6(b). And, each of the slice type (slice_type) for intra-view prediction and the slice type (view_slice_type) for inter-view prediction may indicate, for example, an I-slice type (I_SLICE), a P-slice type (P_SLICE), or a B-slice type (B_SLICE).

For instance, if 'slice_type' of a specific slice is a B-slice and 'view_slice_type' is a P-slice, a macroblock in the specific slice is decoded by a B-slice

(B_SLICE) coding scheme in an intra-view direction (i.e., a temporal direction) and/or by a P-slice (P_SLICE) coding scheme in a view direction.

Meanwhile, the slice type is able to include a P-slice type (VP) for inter-view prediction, a B-slice type (VB) for inter-view prediction and a mixed slice type (Mixed) by prediction resulting from mixing both prediction types. Namely, the mixed slice type provides for prediction using a combination of intra-view and inter-view prediction.

In this case, a P-slice type for inter-view prediction means a case that each macroblock or macroblock partition included in a slice is predicted from one picture in a current view or one picture in a different view. A B-slice type for inter-view prediction means a case that each macroblock or macroblock partition included in a slice is predicted from 'one or two pictures in a current view' or 'one picture in a different view or two pictures in different views, respectively'. And, a mixed slice type for prediction resulting from mixing both predictions means a case that each macroblock or macroblock partition included in a slice is predicted from 'one or two pictures in a current view', 'one picture in a different view or two pictures in different views, respectively', or 'one or two

pictures in a current view and one picture in a different view or two pictures in different views, respectively'.

In other words, a referred picture and allowed macroblock type differ in each slice type, which will be explained in detail with reference to FIG. 43 and FIG. 44 later.

And, the syntax among the aforesaid embodiments of the slice type will be explained in detail with reference to FIG. 40 and FIG. 41 later.

The prediction mode extracting unit 3720 may extract a macroblock prediction mode indicator indicating whether the current macroblock is a macroblock by intra-view prediction, a macroblock by inter-view prediction or a macroblock by prediction resulting from mixing both types of prediction (S3820). For this, the present invention defines a macroblock prediction mode (mb_pred_mode). One embodiment of the macroblock prediction modes will be explained in detail with reference to FIGs. 39,40 and FIGs. 41 later.

The decoding unit 3730 decodes the current macroblock according to the slice type and/or the macroblock prediction mode to receive/produce the current macroblock (S3820). In this case, the current macroblock can be decoded according to the macroblock type of the

current macroblock decided from the macroblock type information. And, the macroblock type can be decided according to the macroblock prediction mode and the slice type.

In case that the macroblock prediction mode is a mode for intra-view prediction, the macroblock type is decided according to a slice type for intra-view prediction and the current macroblock is then decoded by intra-view prediction according to the decided macroblock type.

In case that the macroblock prediction mode is a mode for inter-view prediction, the macroblock type is decided according to a slice type for inter-view prediction and the current macroblock is then decoded by the inter-view prediction according to the decided macroblock type.

In case that the macroblock prediction mode is a mode for prediction resulting from mixing both predictions, the macroblock type is decided according to a slice type for intra-view prediction and a slice type for inter-view prediction, and the current macroblock is then decoded by the prediction resulting from mixing both predictions according to each of the decided macroblock types.

In this case, the macroblock type depends on a macroblock prediction mode and a slice type. In particular, a prediction scheme to be used for a macroblock type may be

determined from a macroblock prediction mode, and a macroblock type is then decided from macroblock type information by a slice type according to the prediction scheme. Namely, one of or both of the extracted slice_type and view_slice_type are selected based on the macroblock prediction mode.

For instance, if a macroblock prediction mode is a mode for inter-view prediction, a macroblock type may be decided from a macroblock table of slice types (I, P, B) corresponding to a slice type (view_slice_type) for inter-view prediction. The relation between a macroblock prediction mode and a macroblock type will be explained in detail with reference to FIGs. 39,40 and FIGs. 41 later.

FIG. 39 is a diagram of a macroblock prediction modes according to example embodiments of the present invention.

In FIG. 39(a), a table corresponding to one embodiment of macroblock prediction modes (mb_pred_mode) according to the present invention is shown.

In case that intra-view prediction, i.e., temporal prediction is used for a macroblock only, '0' is assigned to a value of the 'mb_pred_mode'. In case that inter-view prediction is used for a macroblock only, '1' is assigned to a value of the 'mb_pred_mode'. In case that both

temporal and inter-view prediction is used for a macroblock, '2' is assigned to a value of the 'mb_pred_mode'.

In this case, if a value of the 'mb_pred_mode' is '1', i.e., if the 'mb_pred_mode' indicates the inter-view prediction, view direction List0 (ViewList0) or view direction List1 (ViewList1) is defined as a reference picture list for the inter-view prediction.

In FIG. 39(b), the relation between a macroblock prediction mode and a macroblock type according to another embodiment is shown.

If a value of 'mb_pred_mode' is '0', temporal prediction is used only. And, a macroblock type is decided according to a slice type (slice_type) for intra-view prediction.

If a value of 'mb_pred_mode' is '1', inter-view prediction is used only. And, a macroblock type is decided according to a slice type (view_slice_type) for inter-view prediction.

If a value of 'mb_pred_mode' is '2', mixed prediction of both temporal and intra-view prediction is used. And, two macroblock types are decided according to a slice type (slice_type) for intra-view prediction and a slice type (view_slice_type) for inter-view prediction.

Based on the macroblock prediction mode, the

macroblock type is given based on the slice type as shown in tables 1-3 below. [Please insert tables 7-12 - 7-14 in N6540 here as tables 1-3]

In other words, in this embodiment, a prediction scheme used for a macroblock and a slice type referred to are decided by a macroblock prediction mode. And, a macroblock type is decided according to the slice type.

FIG. 40 and FIG. 41 are diagrams of example embodiments of the syntax of a portion of the video signal received by the apparatus for decoding the video signal. As shown, the syntax has slice type and macroblock prediction mode information according to an embodiment of the present invention.

In FIG. 40, an example syntax is shown. In the syntax, the field 'slice_type' and the field 'view_slice_type' provide slice types and the field 'mb_pred_mode' provides a macroblock prediction mode.

According to the present invention, the 'slice_type' field provides a slice type for intra-view prediction and the 'view_slice_type' field provides a slice type for inter-view prediction. Each slice type can become I-slice type, P-slice type or B-slice type. If a value of the 'mb_pred_mode' is '0' or '1', one macroblock type is decided. Yet, in case that a value of the 'mb_pred_mode' is

'2', it can be seen that another macroblock type (or two types) is further decided. In other words, the syntax shown in (a) of FIG. 40 indicates that 'view_slice_type' is added to further apply the conventional slice types (I, P, B) to multi-view video coding.

In FIG. 41, another example syntax is shown. In the syntax, a 'slice_type' field is employed to provide a slice type and a 'mb_pred_mode' field is employed to provide a macroblock prediction mode.

According to the present invention, the 'slice_type' field may include, among others, a slice type (VP) for inter-view prediction, a slice type-B (VB) for inter-view prediction and a mixed slice type (Mixed) for prediction resulting from mixing both intra-view and inter-view predictions.

If a value in the 'mb_pred_mode' field is '0' or '1', one macroblock type is decided. Yet, in case that a value of the 'mb_pred_mode' field is '2', it can be seen that an additional (i.e., total of two) macroblock type is decided. In this embodiment, the slice type information exists in a slice header, which will be explained in detail with respect to FIGs. 42. In other words, the syntax shown in FIG. 41 indicates that VP, VB and Mixed slice types are added to the conventional slice type (slice_type).

FIGs. 42 are diagrams of examples for applying the slice types shown in FIG. 41.

The diagram in FIG. 42(a) shows that a P-slice type (VP) for inter-view prediction, a B-slice type (VB) for inter-view prediction and a mixed slice type (Mixed) for prediction resulting from mixing both predictions may exist as the slice type, in addition to other slice types, in a slice header. In particular, the slice types VP, VB and Mixed according to an example embodiment are added to the slice types that may exist in a general slice header.

The diagram in FIG. 42(b) shows that a P-slice type (VP) for inter-view prediction, a B-slice type (VB) for inter-view prediction and a mixed slice type (Mixed) for prediction resulting from mixing both predictions may exist as the slice type in a slice header for multi-view video coding (MVC). In particular, the slice types according to an example embodiment are defined in a slice header for multi-view video coding.

The diagram in FIG. 42(c) shows that a slice type (VP) for inter-view prediction, a B-slice type (VB) for inter-view prediction and a mixed slice type (Mixed) for prediction resulting from mixing both predictions may exist as the slice type, in addition to existing slice type for scalable video coding, in a slice header for scalable video

coding (SVC). In particular, the slice types VP, VB and Mixed according to an example embodiment are added to slice types that may exist in a slice header of the scalable video coding (SVC) standard.

FIG. 43 is a diagram of various slice type examples included in the slice type shown in FIG. 41.

In FIG. 43(a), a case that a slice type is predicted from one picture in a different view is shown. So, a slice type becomes a slice type (VP) for inter-view prediction.

In FIG. 43(b), a case that a slice type is predicted from two pictures in different views, respectively is shown. So, a slice type becomes a B-slice type (VB) for inter-view prediction.

In FIGs. 43(c) and 43(f), a case that a slice type is predicted from one or two pictures in a current view and one picture in a different view is shown. So, a slice type becomes a mixed slice type (Mixed) for prediction resulting from mixing both predictions. Also, in FIGs. 43(d) and 43(e), a case that a slice type is predicted from one or two pictures in a current view and two pictures in different views is shown. So, a slice type also becomes a mixed slice type (Mixed).

FIG. 44 is a diagram of a macroblock allowed for

the slice types shown in FIG. 41.

Referring to FIG. 44, an intra macroblock (I), a macroblock (P) predicted from one picture in a current view or a macroblock (VP) predicted from one picture in a different view is allowed for a P-slice type (VP) by inter-view prediction.

An intra macroblock (I), a macroblock (P or B) predicted from one or two pictures in a current view or a macroblock VP or VB predicted from one picture in a different view or two pictures in different views, respectively, are allowed for a B-slice type (VB) by inter-view prediction.

And, an intra macroblock (I); a macroblock (P or B) predicted from one or two pictures in a current view; a macroblock (VP or VB) predicted from one picture in a different view or two pictures in different views, respectively, or a macroblock (Mixed) predicted from one or two pictures in a current view, one picture in a different view or two pictures in different views, respectively, are allowed for a mixed slice type (Mixed).

FIGs. 45-47 are diagrams of a macroblock type of a macroblock existing in a mixed slice type (Mixed) according to embodiments of the present invention.

In FIGs. 45(a) and 45(b), configuration schemes for

a macroblock type (mb_type) and sub-macroblock type (sub_mb_type) of a macroblock existing in a mixed slice are shown, respectively.

In FIGs. 46 and 47, binary representation of predictive direction(s) of a macroblock existing in a mixed slice and actual predictive direction(s) of the mixed slice are shown, respectively.

According to an embodiment of the present invention, a macroblock type (mb_type) is prepared by considering both a size (Partition_Size) of a macroblock partition and a predictive direction (Direction) of a macroblock partition.

And, a sub-macroblock type (sub_mb_type) is prepared by considering both a size (Sub_Partition_Size) of a sub-macroblock partition and a predictive direction (Sub_Direction) of each sub-macroblock partition.

Referring to FIG. 45(a), 'Direction0' and 'Direction1' indicate a predictive direction of a first macroblock partition and a predictive direction of a second macroblock partition, respectively. In particular, in case of a 8x16 macroblock, 'Direction0' indicates a predictive direction for a left 8x16 macroblock partition and 'Direction1' indicates a predictive direction for a right 8x16 macroblock partition. A configurational principle of macroblock type (mb_type) is explained in detail as follows.

First, the first two bits indicate a partition size (Partition_Size) of a corresponding macroblock and a value of 0~3 is available for the first two bits. And, four bits following the first two bits indicate a predictive direction (Direction) in case that a macroblock is divided into partitions.

For instance, in case of a 16x16 macroblock, four bits indicating a predictive direction of the macroblock are attached to a rear of the first two bits. In case of a 16x8 macroblock, four bits following the first two bits indicate a predictive direction (Direction0) of a first partition and another four bits are attached to the former four bits to indicate a predictive direction (Direction1) of a second partition. Likewise, in case of a 8x16 macroblock, eight bits are attached to a rear of the first two bits. In this case, the first four bits of the eight bits attached to the first two bits indicate a predictive direction of a first partition and a next four bits indicate a predictive direction of a second partition.

Referring to FIG. 45(b), a predictive direction (Sub_Direction) of a sub-macroblock is used in a same manner as a predictive direction (Direction) of the macroblock partition shown in FIG. 45(a). A configuration principle of sub-macroblock type (sub_mb_type) is explained

in detail as follows.

First, the first two bits indicate a partition size (Partition_Size) of a corresponding macroblock and the second two bits, next to the former two bits, indicate a partition size (Sub_Partition_Size) of a sub-macroblock of the corresponding macroblock. A value of 0~3 is available for each of the first and second two bits. Subsequently, four bits attached next to the second two bits indicate a predictive direction (Sub_Direction) in case that a macroblock is divided into sub-macroblock partitions. For instance, if a size (Partition_Size) of a partition of a macroblock is 8x8 and if a size (Sub_Partition_Size) of a partition of a sub-macroblock is 4x8, the first two bits have a value of 3, the second two bits have a value of 2, the first four bits next to the second two bits indicate a predictive direction for a left 4x8 block of two 4x8 blocks, and the second four bits next to the first four bits indicate a predictive direction for a right 4x8 block.

Referring to FIG. 46, a predictive direction of a macroblock is constructed with four bits. And, it can be seen that each binary representation becomes '1' according to a case of referring to a picture at the left (L), top (T), right (R) or bottom (B) position of a current picture.

Referring to FIG. 47, for example, in case that a

predictive direction is top (T), a picture located at a top in a view direction of a current picture is referred to. In case that a predictive direction corresponds to all directions (LTRB), it can be seen that pictures in all directions (LTRB) of a current picture are referred to.

FIG. 48 is a block diagram of an apparatus for encoding a video signal according to an embodiment of the present invention.

Referring to FIG. 48, an apparatus for encoding a video signal according to an embodiment of the present invention. The apparatus includes a macroblock type deciding unit 4810, a macroblock generating unit 4820 and an encoding unit 4830.

FIG. 49 is a flowchart of a method of encoding a video signal in the encoding apparatus shown in FIG. 48 according to an embodiment of the present invention.

Referring to FIG. 49, a method of encoding a video signal according to an embodiment of the present invention includes a step S4910 of deciding a first macroblock type for intra-view prediction and a second macroblock type for inter-view prediction, a step S4920 of generating a first macroblock having the first macroblock type and a second macroblock having the second macroblock type, a step S4930 of generating a third macroblock using the first and second

macroblocks, and a step S4940 of encoding a macroblock type of a current macroblock and a macroblock prediction mode.

According to the present invention, the macroblock type deciding unit 4810 decides a first macroblock type for intra-view prediction and a second macroblock type for inter-view prediction (S4910) as described in detail above.

Subsequently, the macroblock generating unit 4820 generates a first macroblock having the first macroblock type and a second macroblock having the second macroblock type (S4920) using well-known prediction techniques, and then generates a third macroblock using the first and second macroblocks (S4930). In this case, the third macroblock is generated according to a mean value between the first and second macroblocks.

Finally, the encoding unit 4830 encodes a macroblock type (mb_type) of a current macroblock and a macroblock prediction mode (mb_pred_mode) of the current macroblock by comparing encoding efficiencies of the first to third macroblocks (S4940).

In this case, there are various methods to measure the encoding efficiencies. In particular, a method using RD (rate-distortion) cost is used in this embodiment of the present invention. As is well-known, in the RD cost method, a corresponding cost is calculated with two components: an

encoding bit number generated from encoding a corresponding block and a distortion value indicating an error from an actual sequence.

The first and second macroblock types may be decided in a manner of selecting a macroblock type having a minimum value of the above-explained RD cost. For instance, a macroblock type having a minimum value of the RD cost among macroblock types by intra-view prediction is decided as the first macroblock type. And, a macroblock type having a minimum value of the RD cost among macroblock types by inter-view prediction is decided as the second macroblock type.

In the step of encoding the macroblock type and the macroblock prediction mode, the macroblock type an prediction mode associated with the one of the first and second macroblocks having the smaller RD cost may be selected. Subsequently, the RD cost of the third macroblock is determined. Finally, the macroblock type and macroblock prediction mode of the current macroblock are encoded by comparing the RD cost of the selected first or second macroblock and the RD cost of the third macroblock to each other.

If the RD cost of the selected first or second macroblock is equal to or greater than the RD cost of the

third macroblock, the macroblock type becomes a macroblock type corresponding to the selected first or second macroblock.

For instance, if the RD cost of the first macroblock is smaller than that of the second and third macroblocks, the current macroblock is set as the first macroblock type. And, the macroblock prediction mode (i.e., intra-view) becomes a prediction scheme of a macroblock corresponding to the RD cost.

For instance, if the RD cost of the second macroblock is smaller than that of the first and third macroblocks, an inter-view prediction scheme as a prediction scheme of the second macroblock becomes the macroblock prediction mode of the current macroblock.

Meanwhile, if the RD cost of the third macroblock is smaller than the RD costs of the first and second macroblocks, macroblock types correspond to both the first and second macroblock types. In particular, intra-view prediction and inter-view prediction macroblock types become macroblock types of the current macroblock. And, the macroblock prediction mode becomes a mixed prediction scheme resulting from mixing intra-view and inter-view predictions.

Accordingly, the present invention provides at

least the following effect or advantage.

The present invention is able to exclude the redundancy information between views due to various prediction schemes between views and such information as slice types, macroblock types and macroblock prediction modes; thereby enhancing performance of encoding/decoding efficiency.

INDUSTRIAL APPLICABILITY

While the present invention has been described and illustrated herein with reference to the preferred embodiments thereof, it will be apparent to those skilled in the art that various modifications and variations can be made therein without departing from the spirit and scope of the invention. Thus, it is intended that the present invention covers the modifications and variations of this invention that come within the scope of the appended claims and their equivalents.

WHAT IS CLAIMED IS:

1. A method of decoding a video signal, comprising the steps of:

obtaining view information for identifying a view of a picture from the video signal; and

generating information for reference picture management using the view information.

2. The method of claim 1, the step of generating the information for the reference picture management, comprising the step of constructing a reference picture list for inter-view prediction using the view information.

3. The method of claim 2, the step of constructing the reference picture list for the inter-view prediction, further comprising the step of deriving a first variable for constructing the reference picture list for the inter-view prediction, wherein the reference picture list for the inter-view prediction is constructed using the first variable.

4. The method of claim 3, wherein in the first variable deriving step, a second variable set from the view

information and a third variable derived from the second variable are used.

5. The method of claim 3, the step of constructing the reference picture list for the inter-view prediction, comprising the step of initializing the reference picture list for the inter-view prediction.

6. The method of claim 5, wherein in initializing the reference picture list for the inter-view prediction, the reference picture list for the inter-view prediction is ordered based on the first variable.

7. The method of claim 5, wherein in initializing the reference picture list for the inter-view prediction, the reference picture list for the inter-view prediction is ordered based on a decoding sequence.

8. The method of claim 5, wherein in initializing the reference picture list for the inter-view prediction, the reference picture list for the inter-view prediction is ordered by a different scheme according to a slice type.

9. The method of claim 8, wherein in initializing

the reference picture list for the inter-view prediction, if the slice type is a P/SP slice, a reference picture list 0 for the inter-view prediction is constructed using the first variable and wherein the reference picture list 0 for the inter-view prediction is ordered in order of reference pictures from a reference picture having a highest first variable to a picture having a lowest first variable.

10. The method of claim 8, wherein in initializing the reference picture list for the inter-view prediction, if the slice type is a B slice, a reference picture list 0 for the inter-view prediction is constructed using the first variable and wherein the reference picture list 0 for the inter-view prediction is ordered in order of reference pictures from a reference picture having a highest first variable to a reference picture having a lowest first variable and then ordered in order of the reference pictures from a reference picture having the lowest first variable higher than the first variable of a current picture to a reference picture having the highest first variable.

11. The method of claim 8, wherein in initializing the reference picture list for the inter-view prediction,

if the slice type is a B slice, a reference picture list 1 for the inter-view prediction is constructed using the first variable and wherein the reference picture list 1 for the inter-view prediction is ordered in order of reference pictures from a reference picture having a lowest first variable higher than the first variable of a current picture to a reference picture having a highest first variable and then ordered in order of the reference pictures from a reference picture having the highest first variable lower than the first variable of the current picture to a reference picture having the lowest first variable.

FIG. 1

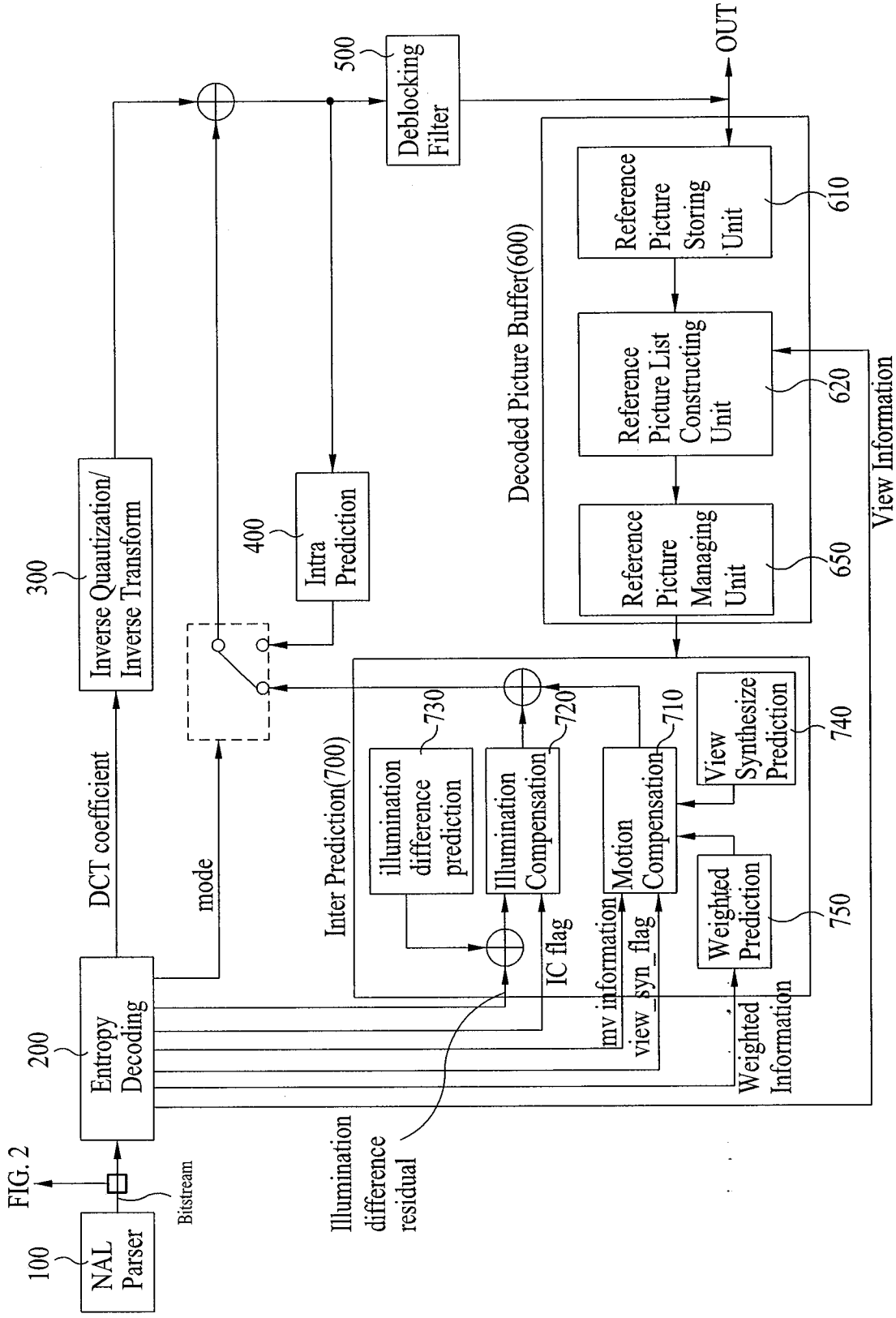

100
NAL Parser
Bitstream
200
Entropy Decoding
DCT coefficient
mode
300
Inverse Quautization/ Inverse Transform
400
Intra Prediction
500
Deblocking Filter
OUT
Decoded Picture Buffer(600)
Reference Picture Storing Unit
610
Reference Picture List Constructing Unit
620
Reference Picture Managing Unit
650
View Information
Inter Prediction(700)
illumination difference prediction
730
Illumination Compensation
720
Motion Compensation
710
View Synthesize Prediction
740
Weighted Prediction
750
IC flag
mv information
view_syn_flag
Weighted Information
Illumination difference residual

FIG. 2

FIG. 2

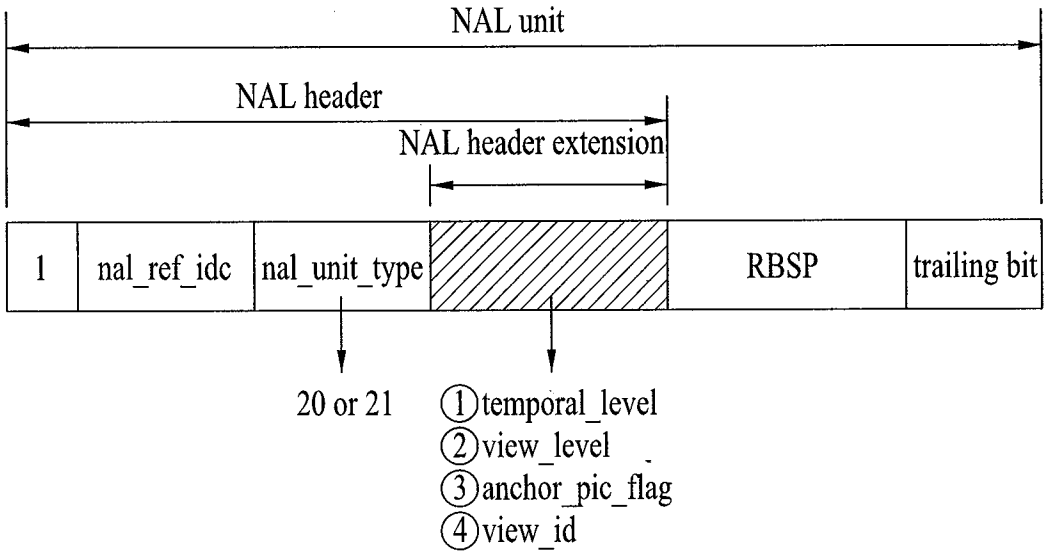

NAL unit
NAL header
NAL header extension
1
nal_ref_idc
nal_unit_type
RBSP
trailing bit
20 or 21
①temporal_level
②view_level
③anchor_pic_flag
④view_id

FIG. 3

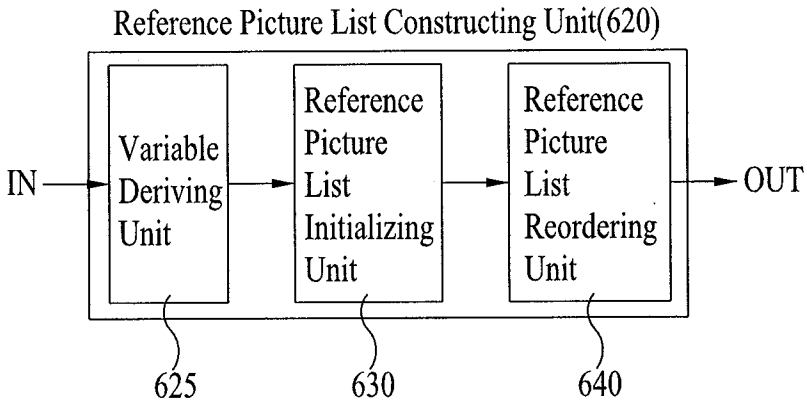

Reference Picture List Constructing Unit(620)
IN
Variable Deriving Unit
Reference Picture List Initializing Unit
Reference Picture List Reordering Unit
OUT
625
630
640

FIG. 4

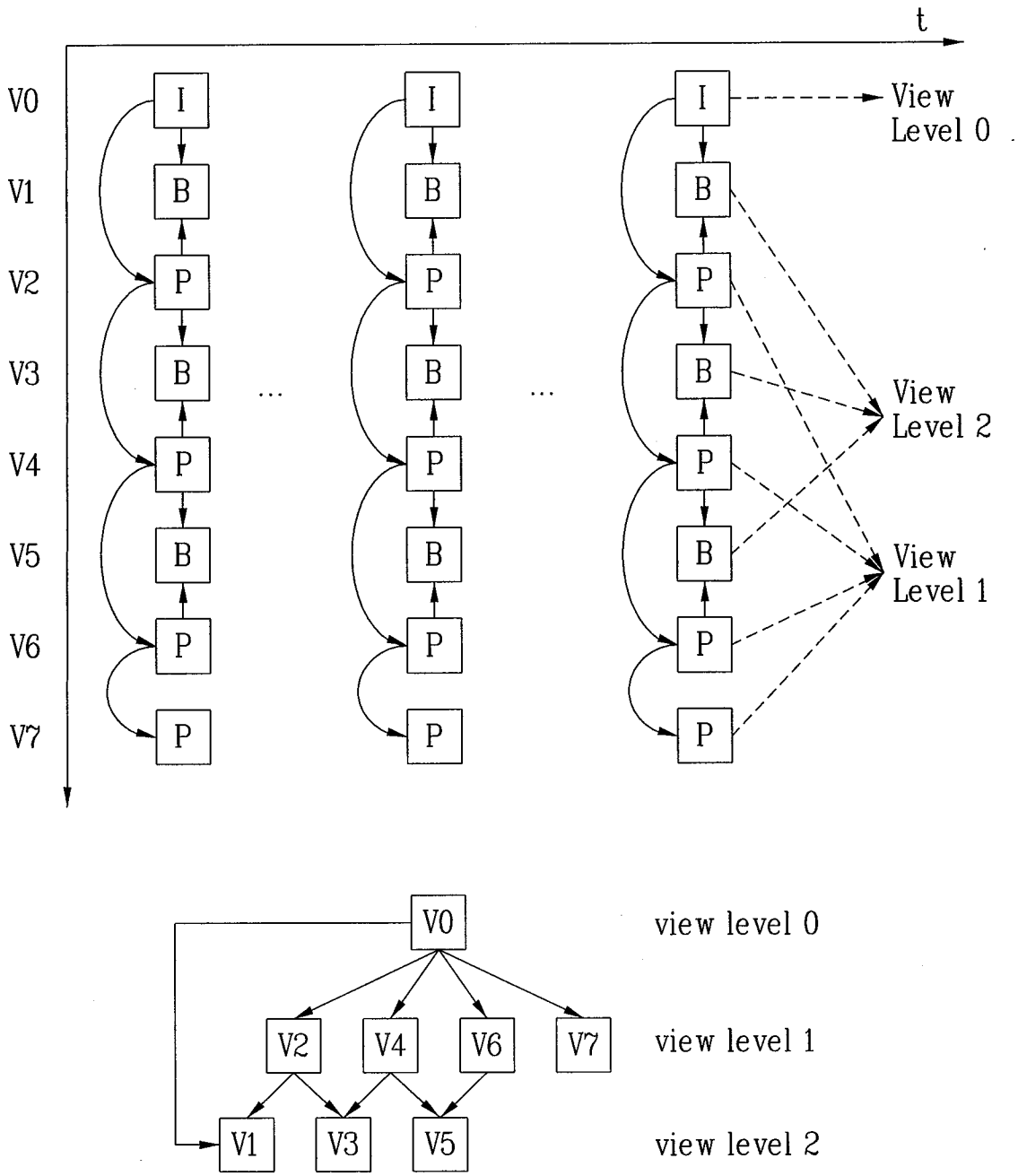

t
V0
V1
V2
V3
V4
V5
V6
V7
I
B
P
View Level 0
View Level 2
View Level 1
V0
V2
V4
V6
V7
V1
V3
V5
view level 0
view level 1
view level 2

FIG. 5

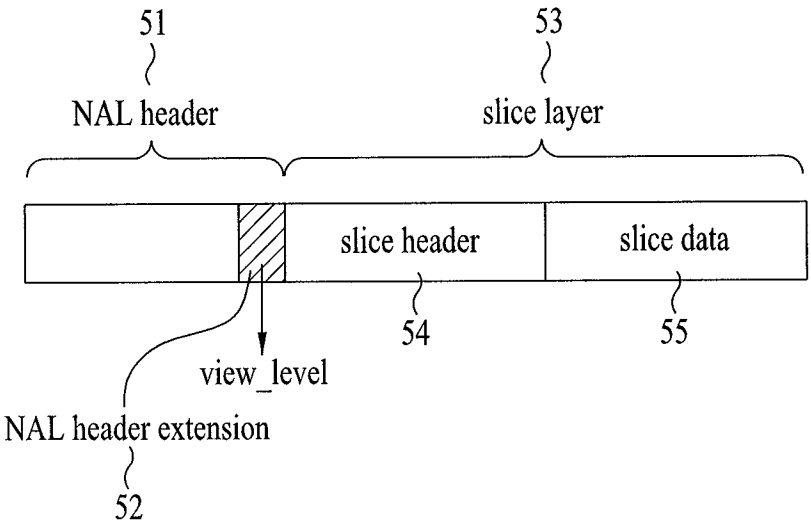

51
53
NAL header
slice layer
slice header
slice data
54
55
view_level
NAL header extension
52

FIG. 6

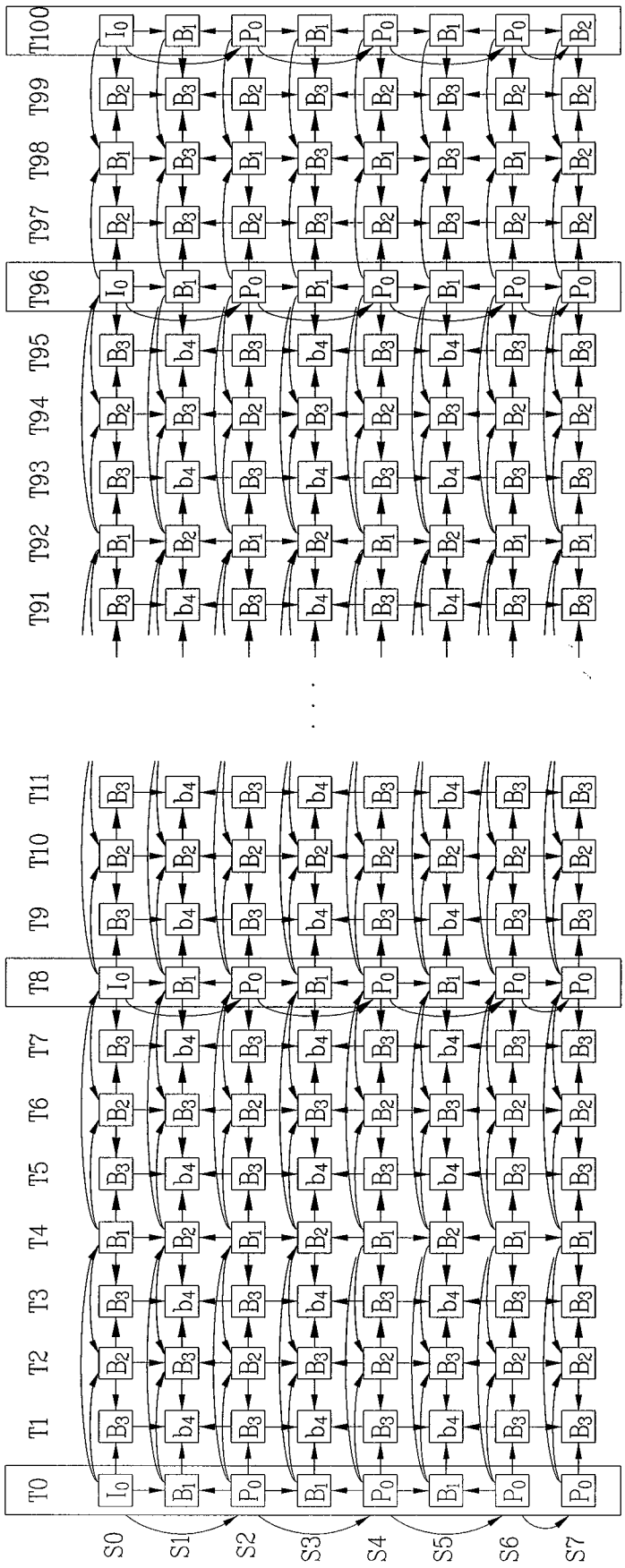

T0
T1
T2
T3
T4
T5
T6
T7
T8
T9
T10
T11
T91
T92
T93
T94
T95
T96
T97
T98
T99
T100
S0
S1
S2
S3
S4
S5
S6
S7

FIG. 7

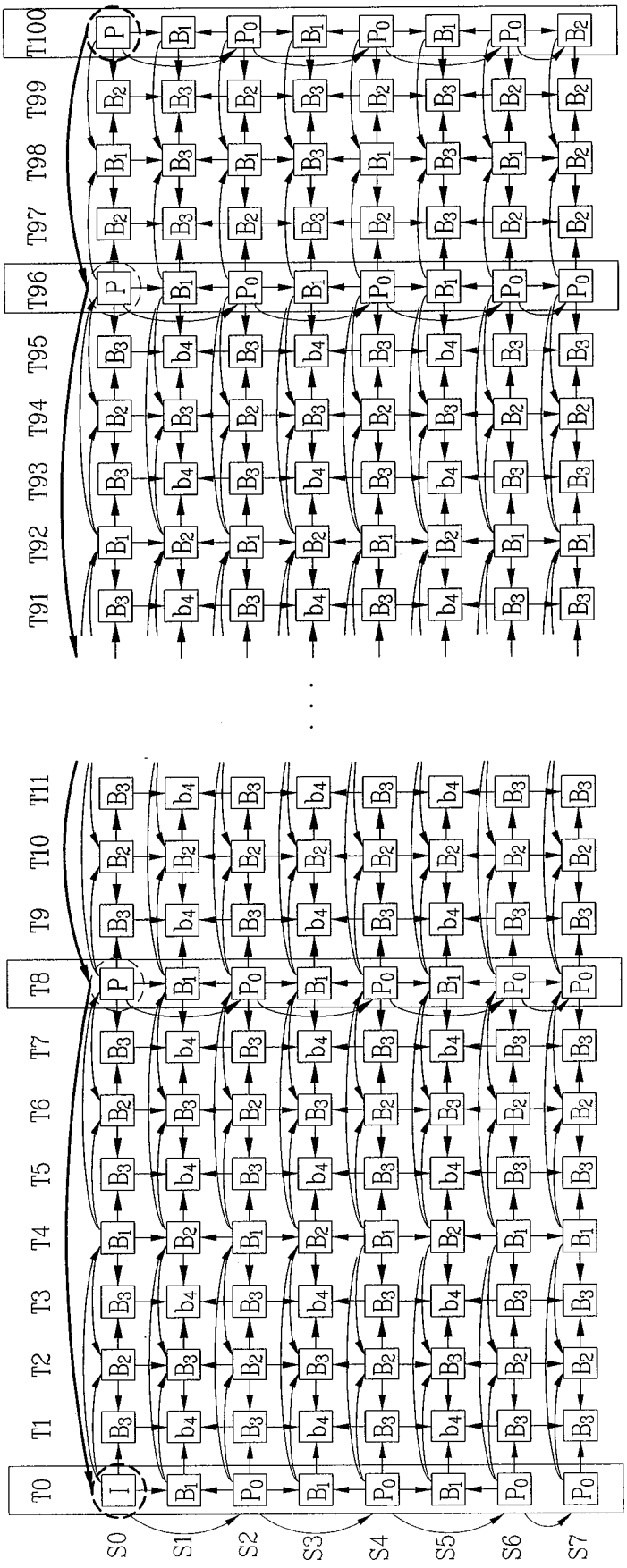

T0
T1
T2
T3
T4
T5
T6
T7
T8
T9
T10
T11
T91
T92
T93
T94
T95
T96
T97
T98
T99
T100
S0
S1
S2
S3
S4
S5
S6
S7

FIG. 8

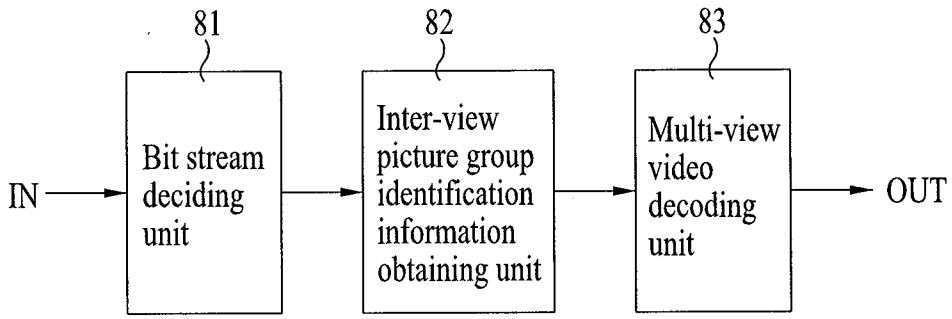

81
82
83
IN
Bit stream deciding unit
Inter-view picture group identification information obtaining unit
Multi-view video decoding unit
OUT

FIG. 9

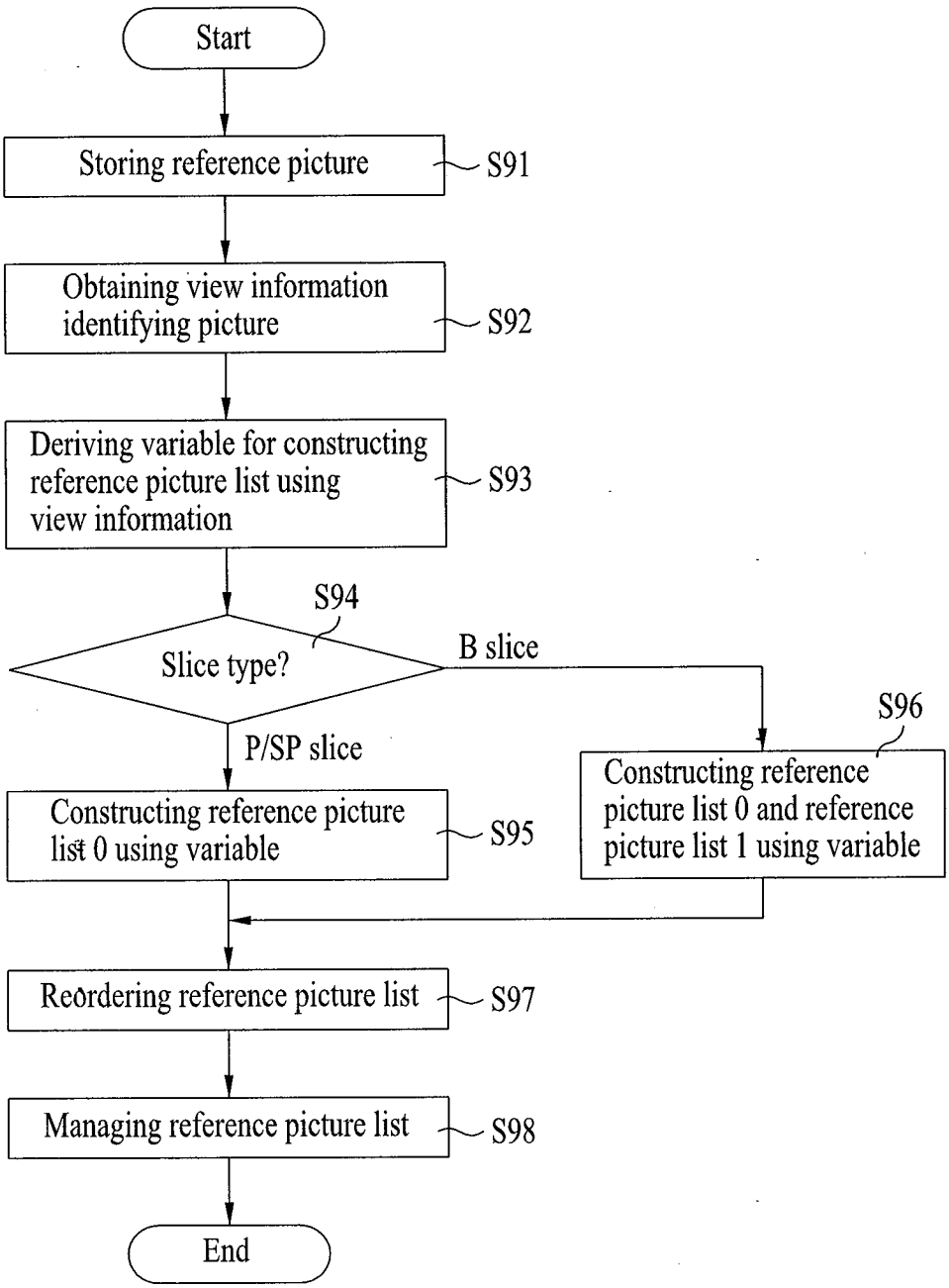
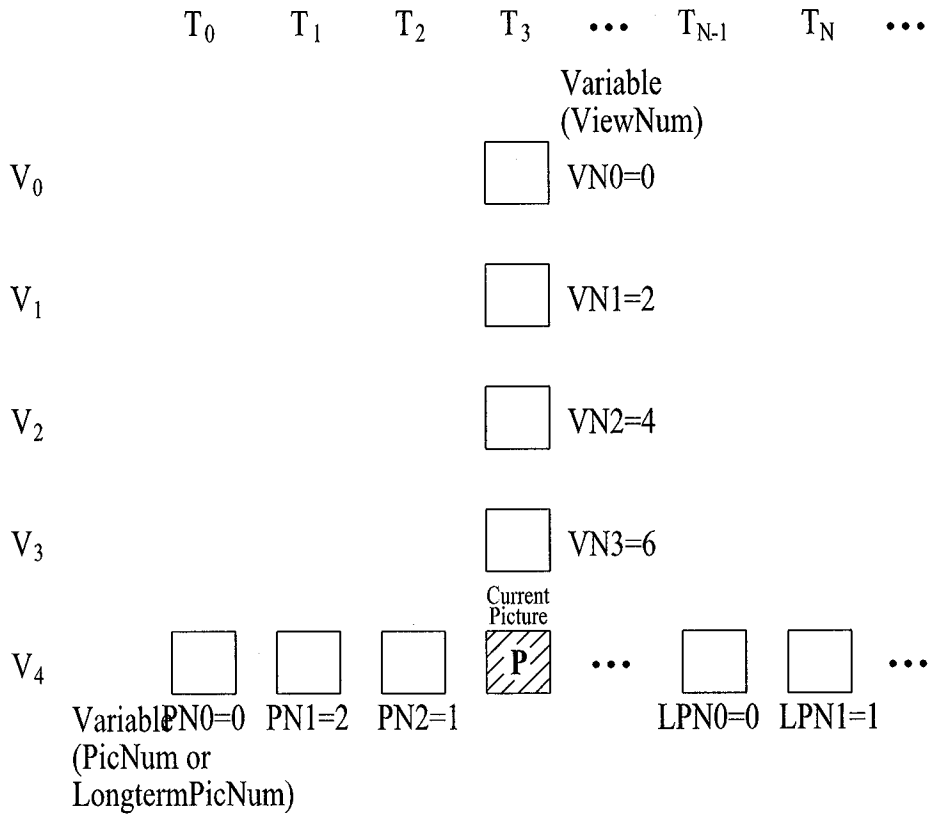

Start
Storing reference picture
S91
Obtaining view information identifying picture
S92
Deriving variable for constructing reference picture list using view information
S93
S94
Slice type?
B slice
P/SP slice
S96
Constructing reference picture list 0 and reference picture list 1 using variable
Constructing reference picture list 0 using variable
S95
Reordering reference picture list
S97
Managing reference picture list
S98
End

FIG. 10


T0 T1 T2 T3 ••• TN-1 TN •••
Variable (ViewNum)
V0
VN0=0
V1
VN1=2
V2
VN2=4
V3
VN3=6
Current Picture
V4
P
•••
•••
Variable (PicNum or LongtermPicNum)
PN0=0
PN1=2
PN2=1
LPN0=0
LPN1=1

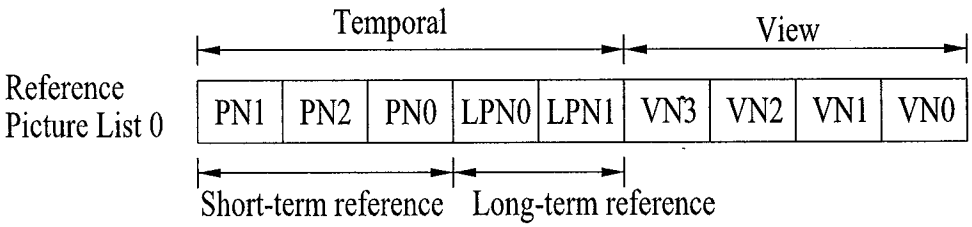

Temporal
View
Reference Picture List 0
PN1
PN2
PN0
LPN0
LPN1
VN3
VN2
VN1
VN0
Short-term reference
Long-term reference

FIG. 11

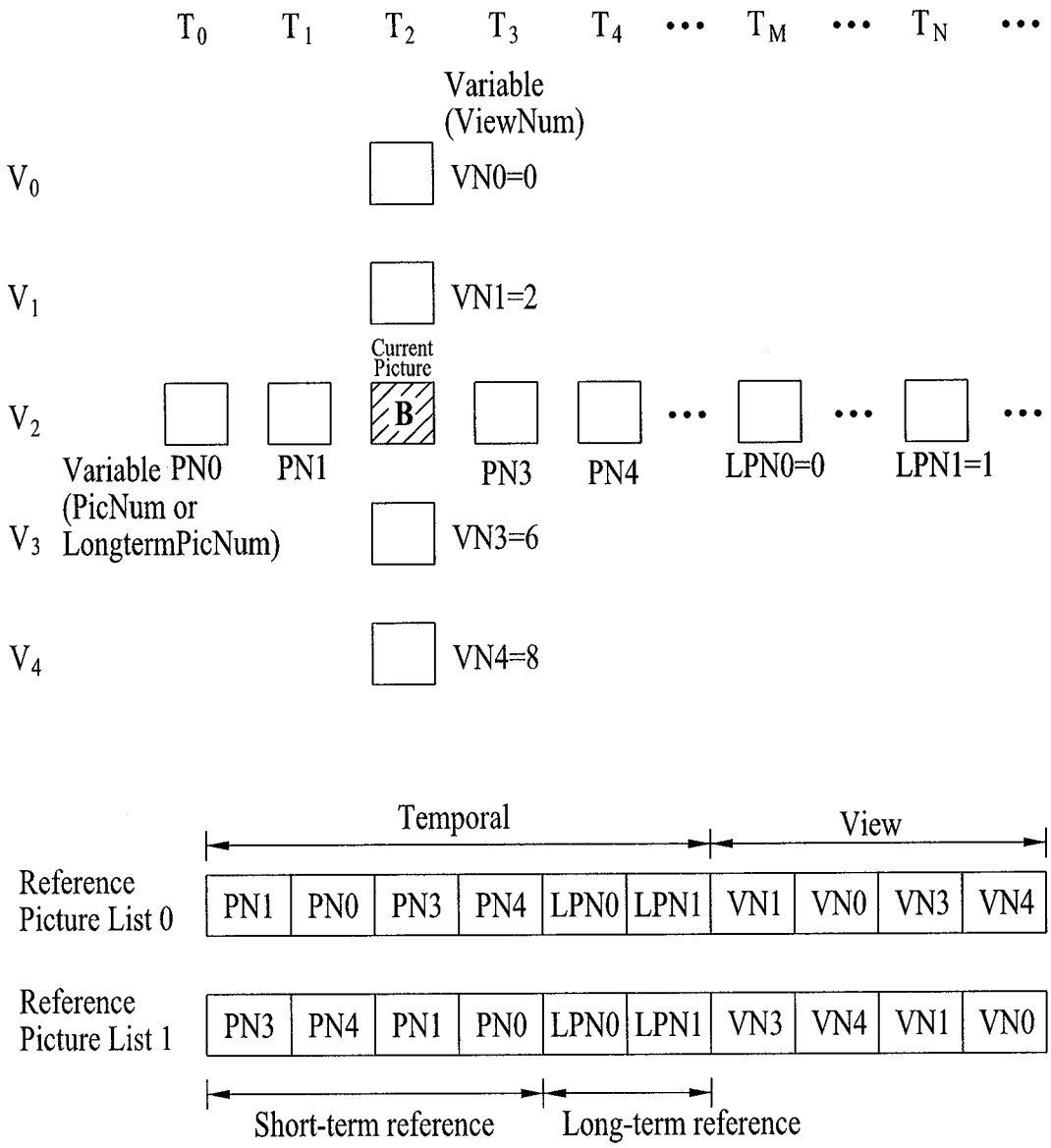

T0 T1 T2 T3 T4 ... TM ... TN ...
Variable (ViewNum)
V0
VN0=0
V1
VN1=2
Current Picture
V2
B
Variable PN0 PN1 PN3 PN4 LPN0=0 LPN1=1
(PicNum or LongtermPicNum)
V3
VN3=6
V4
VN4=8
Temporal
View
Reference Picture List 0
PN1 PN0 PN3 PN4 LPN0 LPN1 VN1 VN0 VN3 VN4
Reference Picture List 1
PN3 PN4 PN1 PN0 LPN0 LPN1 VN3 VN4 VN1 VN0
Short-term reference
Long-term reference

FIG. 12

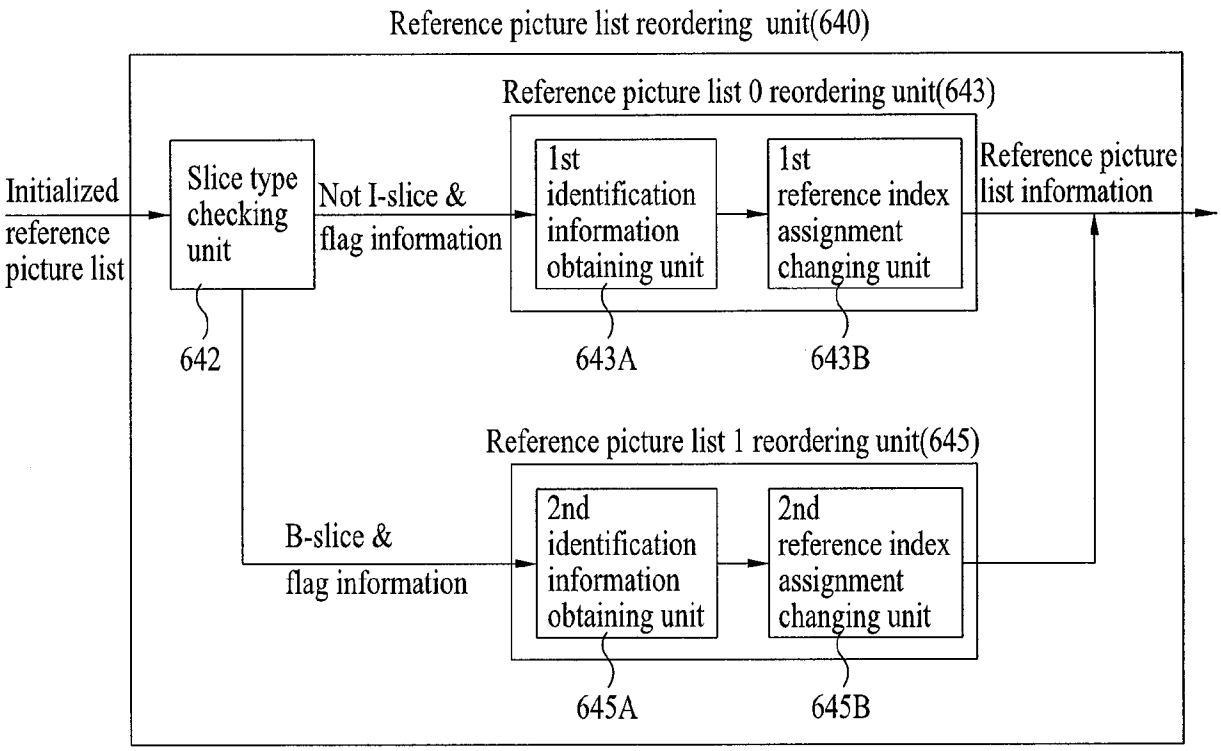

Reference picture list reordering unit(640)
Reference picture list 0 reordering unit(643)
Initialized reference picture list
Slice type checking unit
Not I-slice & flag information
1st identification information obtaining unit
1st reference index assignment changing unit
Reference picture list information
642
643A
643B
Reference picture list 1 reordering unit(645)
B-slice & flag information
2nd identification information obtaining unit
2nd reference index assignment changing unit
645A
645B

FIG. 13

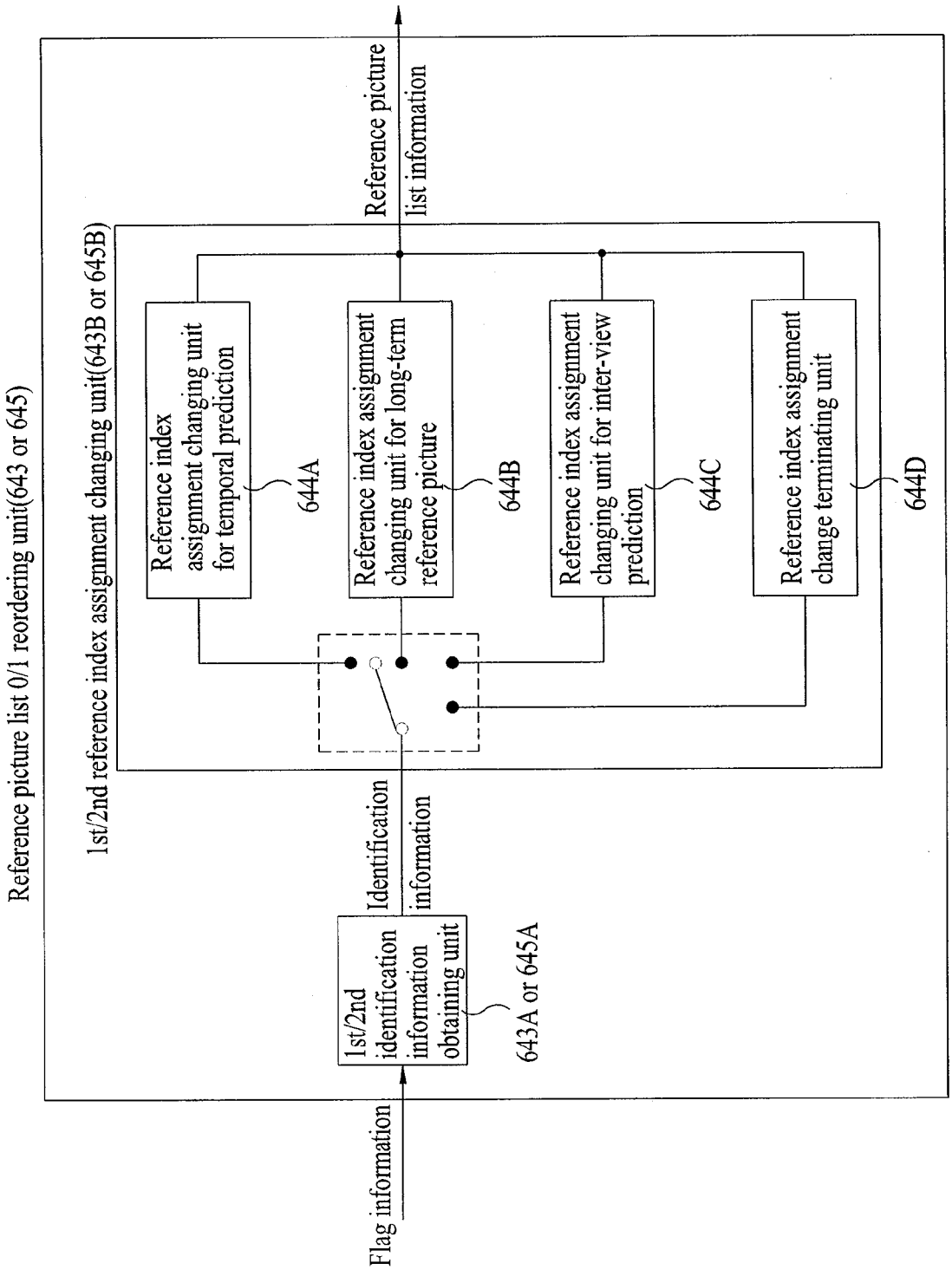

Reference picture list 0/1 reordering unit(643 or 645)
1st/2nd reference index assignment changing unit(643B or 645B)
Reference picture list information
Reference index assignment changing unit for temporal prediction
644A
Reference index assignment changing unit for long-term reference picture
644B
Reference index assignment changing unit for inter-view prediction
644C
Reference index assignment change terminating unit
644D
Identification information
1st/2nd identification information obtaining unit
643A or 645A
Flag information

FIG. 14

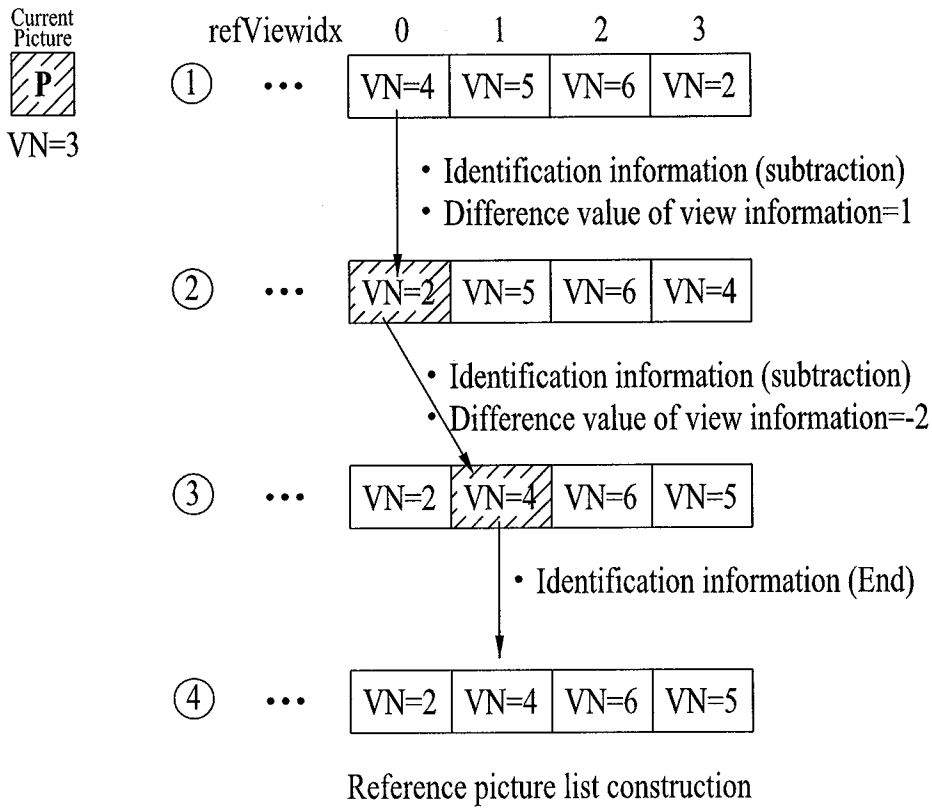

Current Picture
P
VN=3
refViewidx 0 1 2 3
①
VN=4 VN=5 VN=6 VN=2
• Identification information (subtraction)
• Difference value of view information=1
②
VN=2 VN=5 VN=6 VN=4
• Identification information (subtraction)
• Difference value of view information=-2
③
VN=2 VN=4 VN=6 VN=5
• Identification information (End)
④
VN=2 VN=4 VN=6 VN=5
Reference picture list construction

FIG. 15

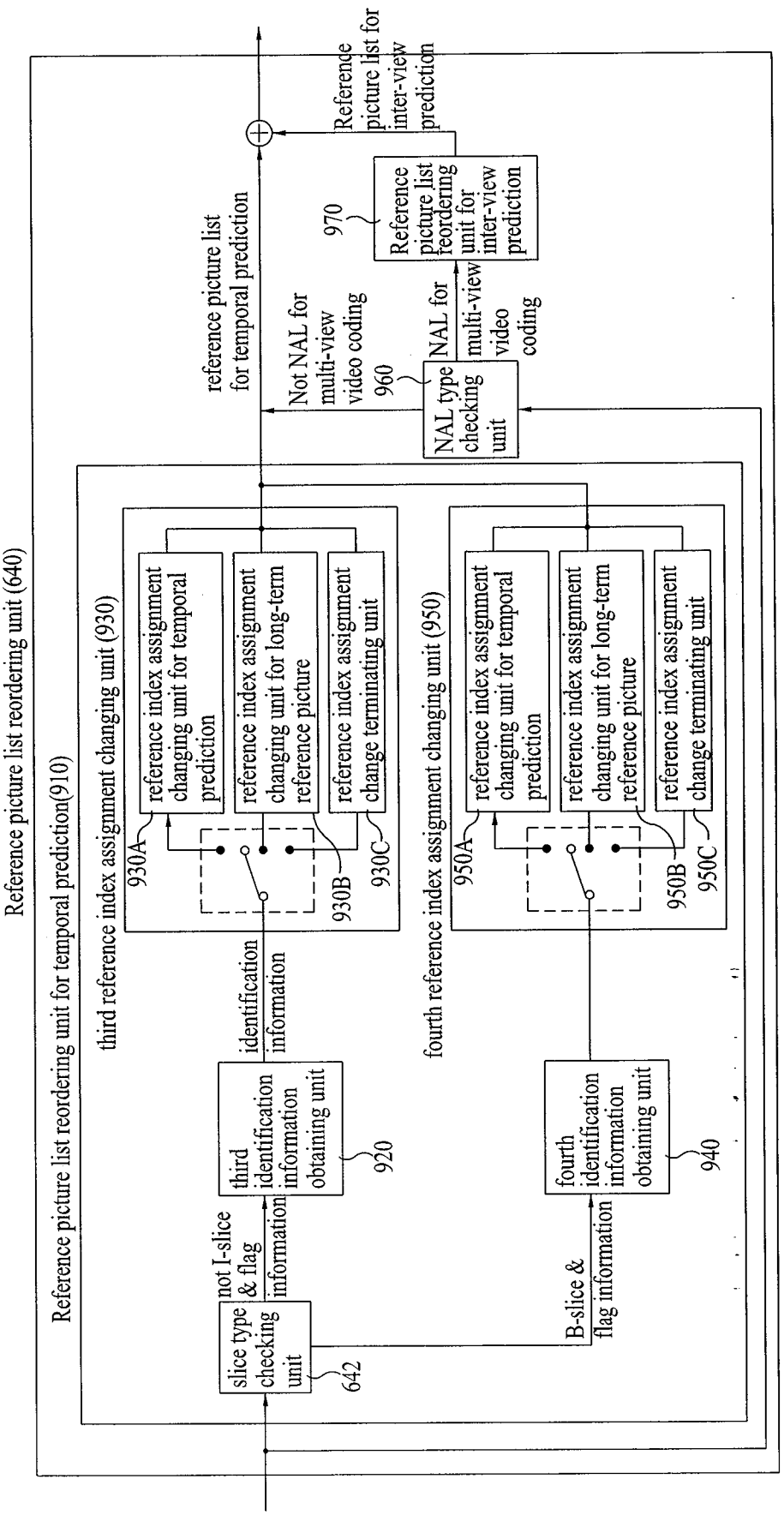

Reference picture list reordering unit (640)
Reference picture list reordering unit for temporal prediction(910)
slice type checking unit
642
not I-slice & flag information
B-slice & flag information
third identification information obtaining unit
920
fourth identification information obtaining unit
940
identification information
third reference index assignment changing unit (930)
930A
reference index assignment changing unit for temporal prediction
930B
reference index assignment changing unit for long-term reference picture
930C
reference index assignment change terminating unit
fourth reference index assignment changing unit (950)
950A
reference index assignment changing unit for temporal prediction
950B
reference index assignment changing unit for long-term reference picture
950C
reference index assignment change terminating unit
NAL type checking unit
960
Not NAL for multi-view video coding
NAL for multi-view video coding
Reference picture list reordering unit for inter-view prediction
970
reference picture list for temporal prediction
Reference picture list for inter-view prediction

FIG. 16

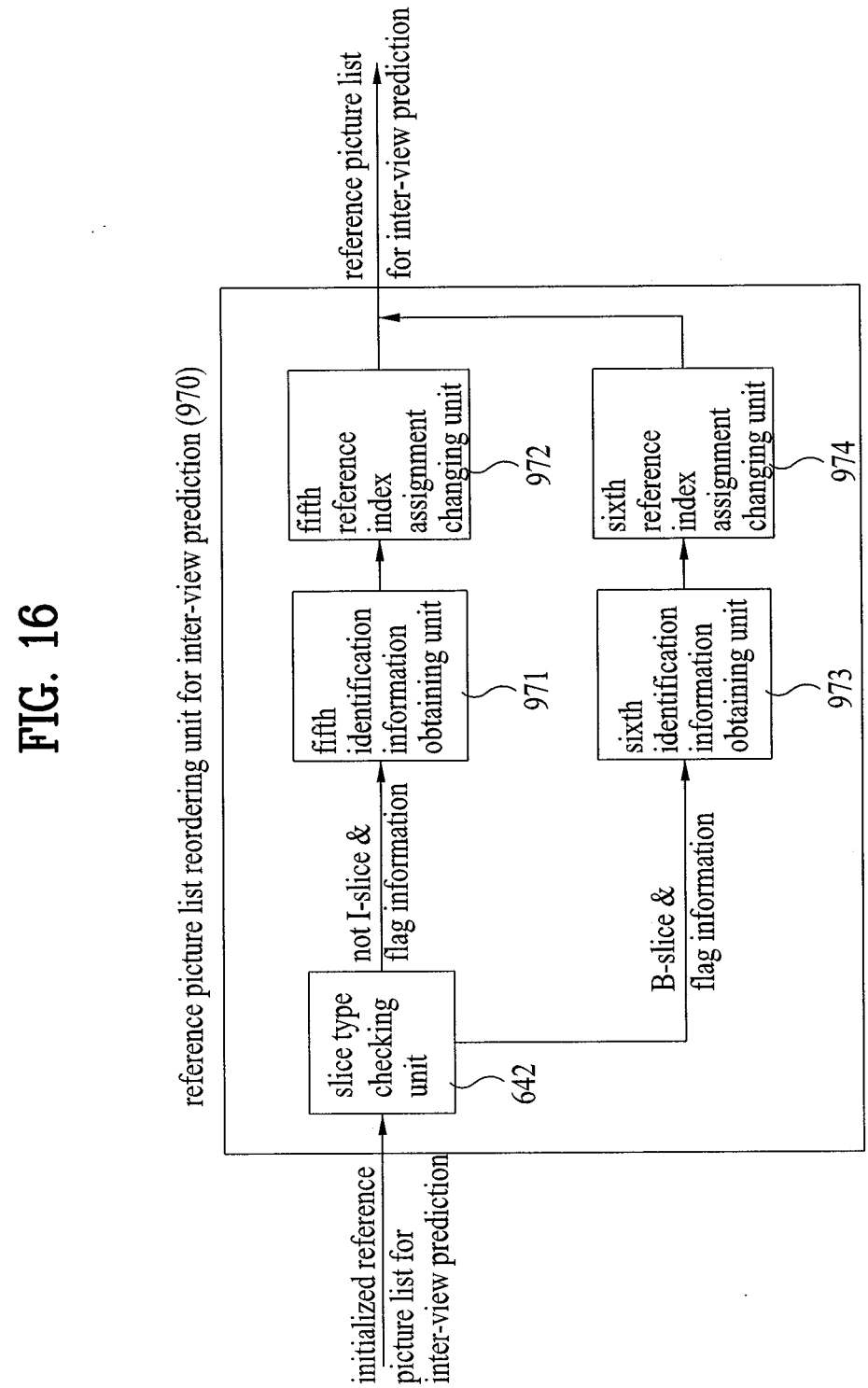
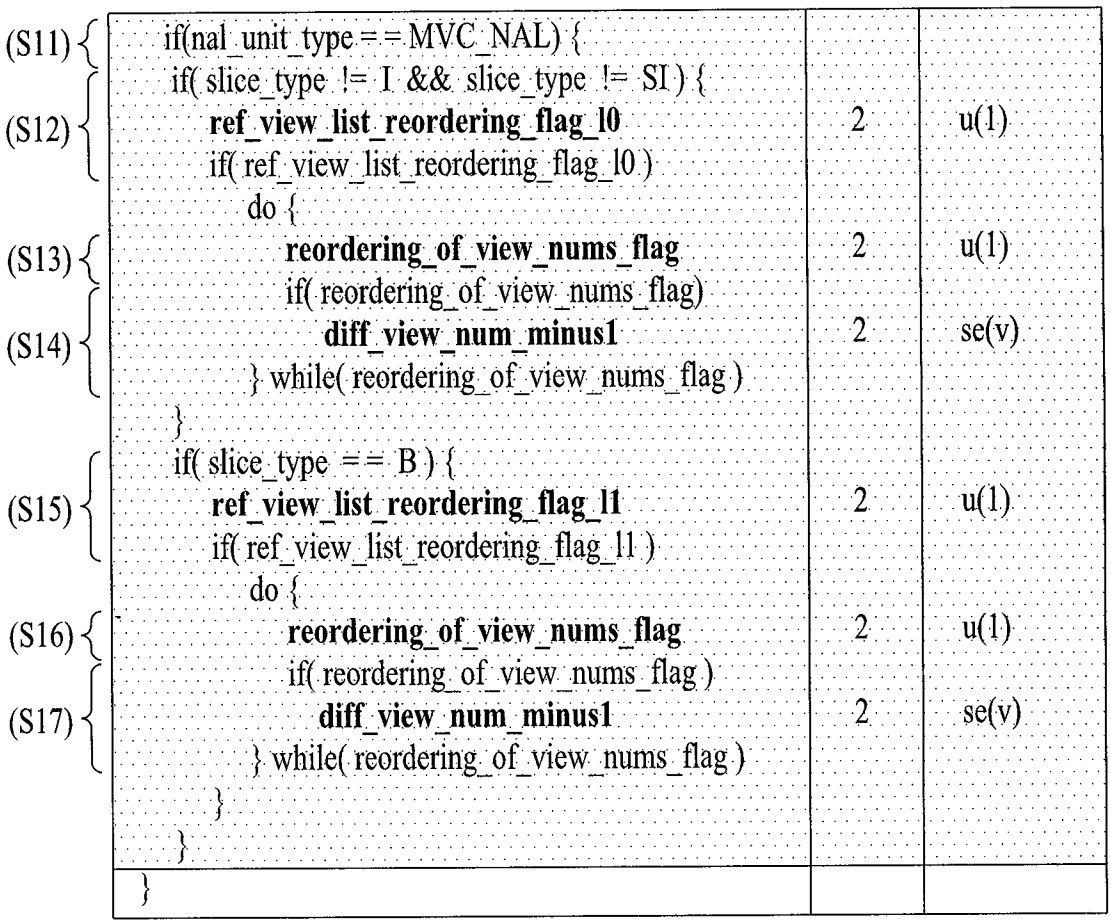

reference picture list reordering unit for inter-view prediction (970)
initialized reference picture list for inter-view prediction
slice type checking unit
642
not I-slice & flag information
B-slice & flag information
fifth identification information obtaining unit
971
sixth identification information obtaining unit
973
fifth reference index assignment changing unit
972
sixth reference index assignment changing unit
974
reference picture list for inter-view prediction

FIG. 17

	ref_pic_list_reordering( ) {	C	Descriptor
	if(slice_type != I && slice_type != SI) {		
(S1)	**ref_pic_list_reordering_flag_l0**	2	u(1)
	if(ref_pic_list_reordering_flag_l0)		
	do {		
(S2)	**reordering_of_pic_nums_idc**	2	ue(v)
	if(reordering_of_pic_nums_idc == 0 ||		
(S3)	reordering_of_pic_nums_idc == 1)		
	abs_diff_pic_num_minus1	2	ue(v)
	else if(reordering_of_pic_nums_idc == 2)		
(S4)	**long_term_pic_num**	2	ue(v)
(S5)	} while(reordering_of_pic_nums_idc != 3)		
	}		
	if(slice_type == B) {		
(S6)	**ref_pic_list_reordering_flag_l1**	2	u(1)
	if(ref_pic_list_reordering_flag_l1)		
	do {		
(S7)	**reordering_of_pic_nums_idc**	2	ue(v)
	if(reordering_of_pic_nums_idc == 0 ||		
(S8)	reordering_of_pic_nums_idc == 1)		
	abs_diff_pic_num_minus1	2	ue(v)
	else if(reordering_of_pic_nums_idc == 2)		
(S9)	**long_term_pic_num**	2	ue(v)
(S10)	} while(reordering_of_pic_nums_idc != 3)		
	}		

FIG. 18



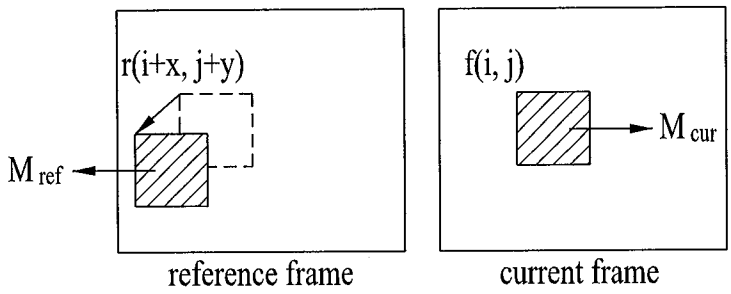
(S11)	if(nal_unit_type == MVC_NAL) {		
	if(slice_type != I && slice_type != SI) {		
(S12)	**ref_view_list_reordering_flag_l0**	2	u(1)
	if(ref_view_list_reordering_flag_l0)		
	do {		
(S13)	**reordering_of_view_nums_flag**	2	u(1)
	if(reordering_of_view_nums_flag)		
(S14)	**diff_view_num_minus1**	2	se(v)
	} while(reordering_of_view_nums_flag)		
	}		
	if(slice_type == B) {		
(S15)	**ref_view_list_reordering_flag_l1**	2	u(1)
	if(ref_view_list_reordering_flag_l1)		
	do {		
(S16)	**reordering_of_view_nums_flag**	2	u(1)
	if(reordering_of_view_nums_flag)		
(S17)	**diff_view_num_minus1**	2	se(v)
	} while(reordering_of_view_nums_flag)		
	}		
	}		
	}		

FIG. 19

	slice_header( ) {	C	Descriptor
	…		
	ref_pic_list_reordering( )		
(S21)	if(nal_unit_type = = MVC_NAL)		
	ref_view_list_reordering( )		
	…		
	}		

	ref_view_list_reordering( ) {	C	Descriptor
	if(slice_type != I && slice_type != SI) {		
(S22)	**ref_view_list_reordering_flag_l0**	2	u(1)
	if(ref_view_list_reordering_flag_l0)		
	do {		
(S23)	**reordering_of_view_nums_flag**	2	u(1)
	if(reordering_of_view_nums_flag)		
(S24)	**diff_view_num_minus1**	2	se(v)
	} while(reordering_of_view_nums_flag)		
	}		
	if(slice_type = = B) {		
(S25)	**ref_view_list_reordering_flag_l1**	2	u(1)
	if(ref_view_list_reordering_flag_l1)		
	do {		
(S26)	**reordering_of_view_nums_flag**	2	u(1)
	if(reordering_of_view_nums_flag)		
(S27)	**diff_view_num_minus1**	2	se(v)
	} while(reordering_of_view_nums_flag)		
	}		
	}		

FIG. 20



$IC_offset = M_{cur} - M_{ref}$

FIG. 21

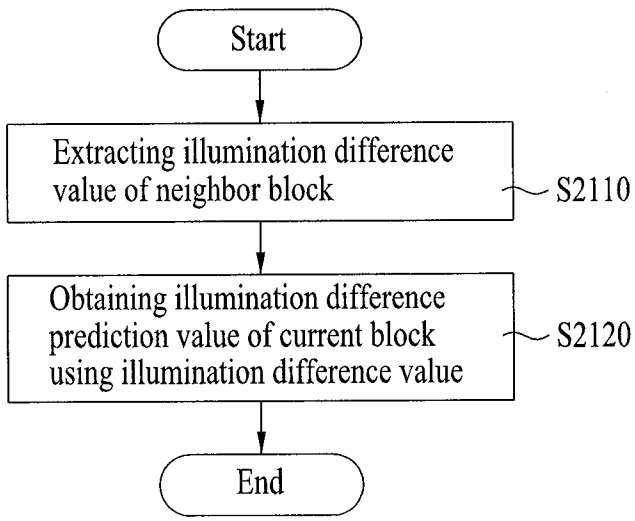


FIG. 22

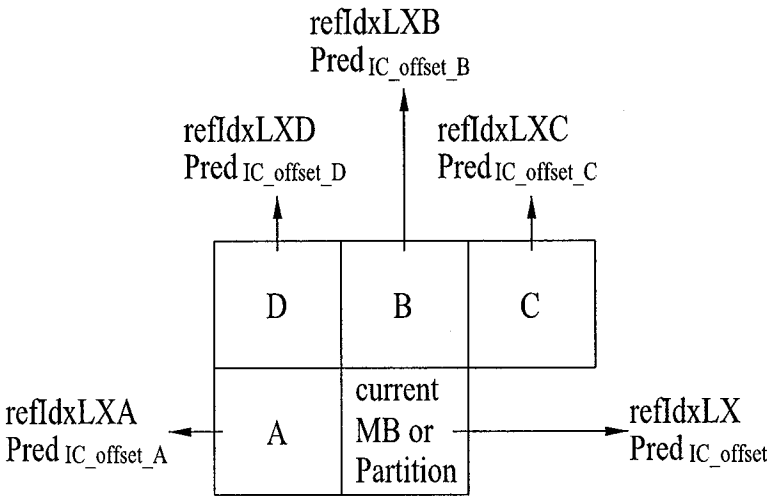

refIdxLXB
Pred IC_offset_B
refIdxLXD
Pred IC_offset_D
refIdxLXC
Pred IC_offset_C
D
B
C
refIdxLXA
Pred IC_offset_A
A
current
MB or
Partition
refIdxLX
Pred IC_offset

FIG. 23

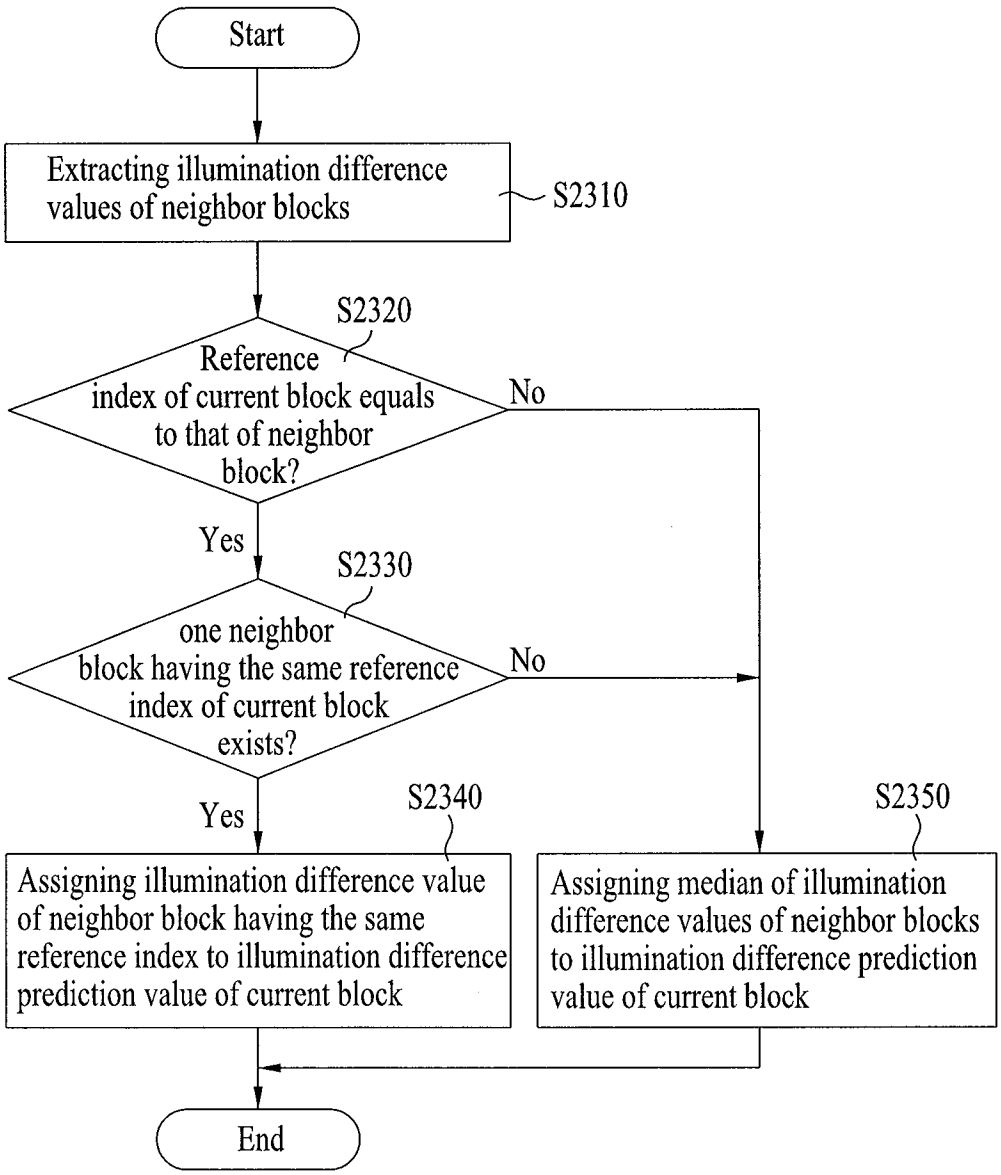

Start
Extracting illumination difference values of neighbor blocks
S2310
S2320
Reference index of current block equals to that of neighbor block?
No
Yes
S2330
one neighbor block having the same reference index of current block exists?
No
Yes
S2340
Assigning illumination difference value of neighbor block having the same reference index to illumination difference prediction value of current block
S2350
Assigning median of illumination difference values of neighbor blocks to illumination difference prediction value of current block
End

FIG. 24

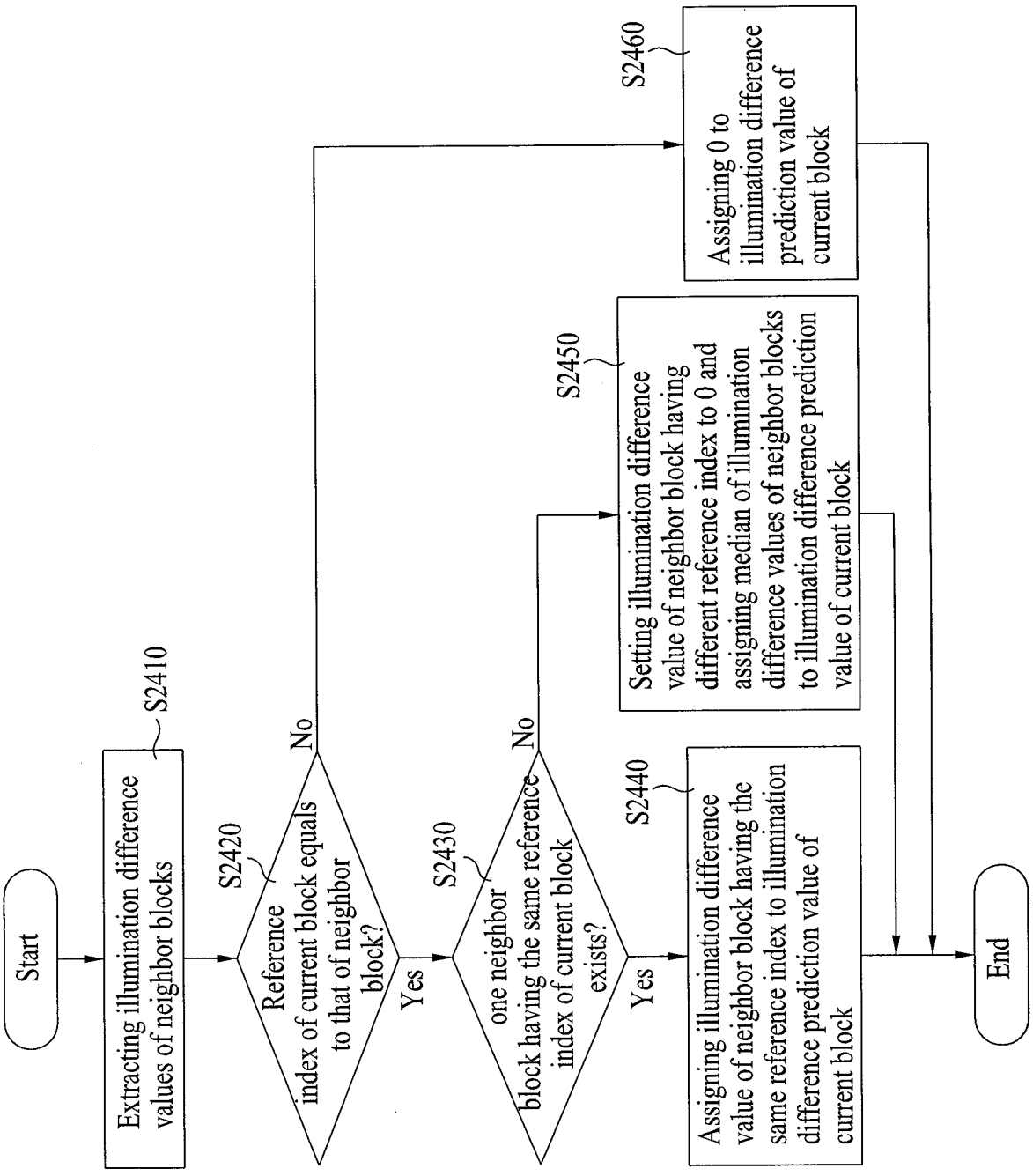

Start
Extracting illumination difference values of neighbor blocks
S2410
Reference index of current block equals to that of neighbor block?
S2420
No
Yes
one neighbor block having the same reference index of current block exists?
S2430
No
Yes
Assigning illumination difference value of neighbor block having the same reference index to illumination difference prediction value of current block
S2440
Setting illumination difference value of neighbor block having different reference index to 0 and assigning median of illumination difference values of neighbor blocks to illumination difference prediction value of current block
S2450
Assigning 0 to illumination difference prediction value of current block
S2460
End

FIG. 25

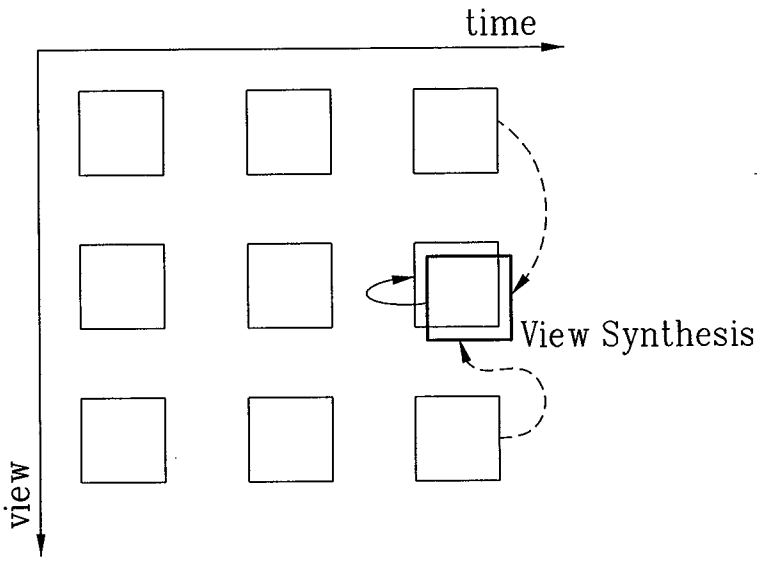

time
view
View Synthesis

FIG. 26

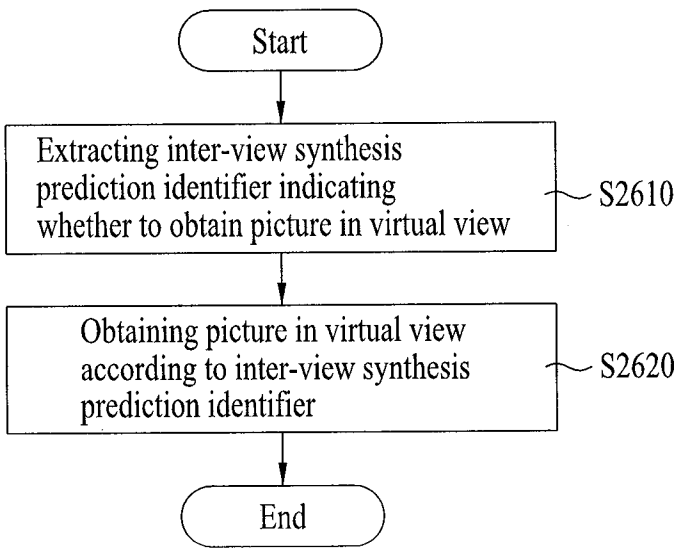

Start
Extracting inter-view synthesis prediction identifier indicating whether to obtain picture in virtual view
S2610
Obtaining picture in virtual view according to inter-view synthesis prediction identifier
S2620
End

FIG. 27

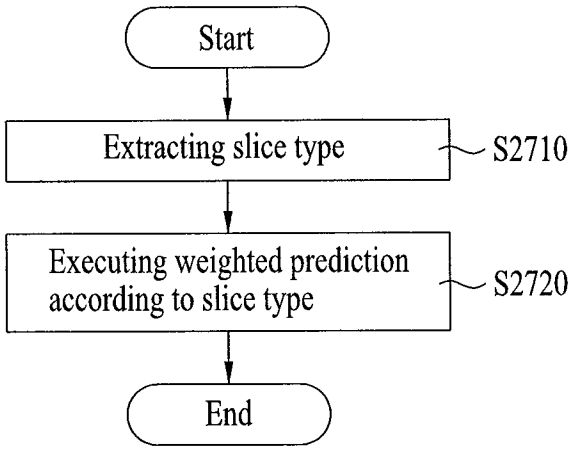


FIG. 28

	Slice types	allowed collective macroblock types
2810	VP (View_P)	I, P, VP
2820	VB (View_B)	I, P, B, VP, VB
2830	Mixed	I, P, B, VP, VB, Mixed

FIG. 29

```
pred_weight_table( )
{
    if( slice_type != VP || slice_type != VB)
    {
        luma_log2_weight_denom
        chroma_log2_weight_denom
        for( i = 0; i <= num_ref_idx_l0_active_minus1; i++ ) {
            luma_weight_l0_flag
            if( luma_weight_l0_flag ) {
                luma_weight_l0[i]
                luma_offset_l0[i]
            }
            chroma_weight_l0_flag
            if( chroma_weight_l0_flag )
                for( j =0; j < 2; j++ ) {
                    chroma_weight_l0[i] [j]
                    chroma_offset_l0[i] [j]
            }
    }                                                          ~2910

    if(slice_type==B|| slice_type==Mixed)
    {
     for( i = 0; i <= num_ref_idx_l1_active_minus1; i++ )
     {
        luma_weight_l1_flag
        if( luma_weight_l1_flag ) {
            luma_weight_l1[i]
            luma_offset_l1[i]
        }
        chroma_weight_l1_flag
        if( chroma_weight_l1_flag )
            for( j = 0; j < 2; j++ ) {
                chroma_weight_l1[i] [j]
                chroma_offset_l1[i] [j]
            }
        }
    }                                                          ~2920
}
```

FIG. 30

```
if(slice_type != P || slice_type != B)
{
    luma_log2_view_weight_denom
    chroma_log2_view_weight_denom
    for( i = 0; i <= num_ref_idx_View10_active_minus1; i++ ) {
        luma_weight_View10_flag
        if( luma_weight_View10_flag ) {
            luma_weight_View10[i]
            luma_offset_View10 luma_offset_View10[i]
    }
    chroma_weight_View10_flag
    if( chroma_weight_View10_flag )
        for( j =0; j < 2; j++ ) {
            chroma_weight_View10[i] [j]
            chroma_offset_View10[i] [j]
        }
    }
    if(slice_type==VB||slice_type==Mixed)
    {
      for( i = 0; i <= num_ref_idx_View11_active_minus1; i++ )
      {
        luma_weight_View11_flag
        if( luma_weight_View11_flag ) {
            luma_weight_View11[i]
            luma_offset_View11[i]
        }
        chroma_weight_View11_flag
        if( chroma_weight_View11_flag )
            for( j = 0; j < 2; j++ ) {
                chroma_offset_View11 chroma_weight_View11[i] [j]
                chroma_offset_View11[i] [j]
            }
        }
      }
    }
}
```

2930

2940

FIG. 31

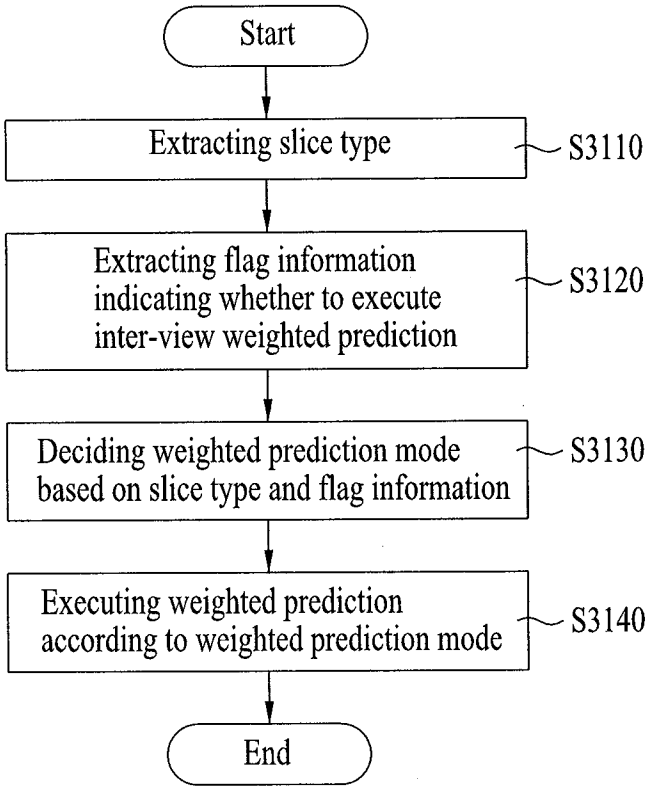

Start
Extracting slice type
S3110
Extracting flag information indicating whether to execute inter-view weighted prediction
S3120
Deciding weighted prediction mode based on slice type and flag information
S3130
Executing weighted prediction according to weighted prediction mode
S3140
End

FIG. 32

VP slice	view_weigthed_pred_flag == 0 : not used view_weighted_pred_flag == 1 : explicit mode
VB slice	view_weighted_bipred_idc == 0 : not used view_weighted_bipred_idc == 1 : explicit mode view_weighted_bipred_idc == 2 : implicit mode

FIG. 33

```
If (( weighted_pred_flag && (slice_type == P || slice_type == SP )) ||
                    ( weighted_bipred_idc == 1 && slice_type == B ) ||
( view_weighted_pred_flag && (slice_type == VP )) ||
                    ( view_weighted_bipred_idc == 1 && slice_type == VB ))
```

FIG. 34

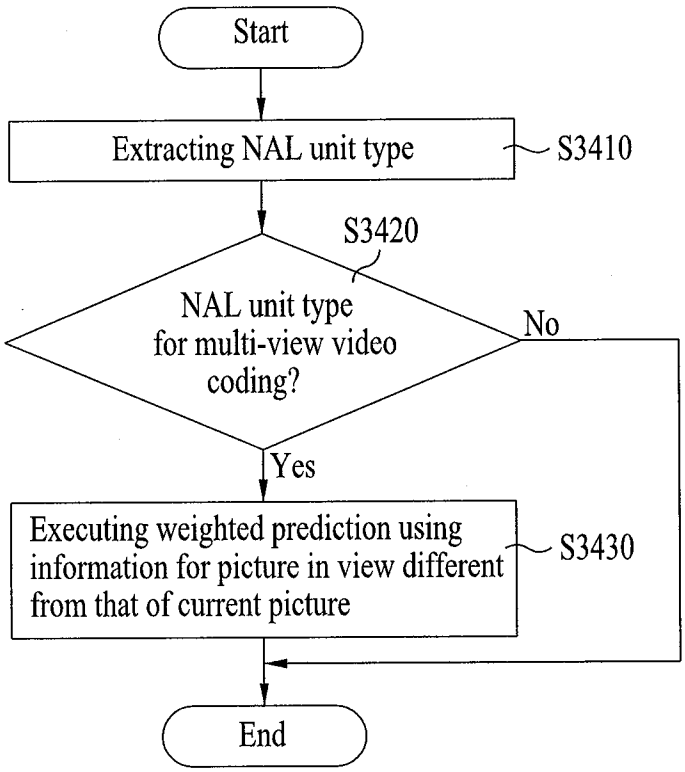

Start
Extracting NAL unit type
S3410
S3420
NAL unit type for multi-view video coding?
No
Yes
Executing weighted prediction using information for picture in view different from that of current picture
S3430
End

FIG. 35

```
pred_weight_table(?) {
  luma_log2_weight_denom
  chroma_log2_weight_denom
  for( i = 0; i <= num_ref_idx_l0_active_minus1; i++ ) {
    luma_weight_l0_flag
    if( luma_weight_l0_flag ) {
      luma_weight_l0[i]
      luma_offset_l0[i]
    }
    chroma_weight_l0_flag
    if( chroma_weight_l0_flag )
      for( j =0; j < 2; j++ ) {
        chroma_weight_l0[i] [j]
        chroma_offset_l0[i] [j]
      }
  }
```

3510

```
  if ( nal_unit_for_MVC) {
    luma_view_log2_weight_denom
    chroma_view_log2_weight_denom
    for( i = 0; i < num_multiview_refs_for_list0; i++ ) {
      luma_view_weight_l0_flag
      if( luma_view_weight_l0_flag ) {
        luma_view_weight_l0[i]
        luma_view_offset_l0[i]
      }
      chroma_view_weight_l0_flag
      if( chroma_view_weight_l0_flag )
        for( j =0; j < 2; j++ ) {
          chroma_view_weight_l0[i] [j]
          chroma_view_offset_l0[i] [j]
        }
    }
  }
```

3520

FIG. 36

```
if( slice_type == B ) {
  for( i = 0; i <= num_ref_idx_l1_active_minus1; i++ ) {
    luma_weight_l1_flag
    if( luma_weight_l1_flag ) {
      luma_weight_l1[i]
      luma_offset_l1[i]
    }
    chroma_weight_l1_flag
    if( chroma_weight_l1_flag )
      for( j = 0; j < 2; j++ ) {
        chroma_weight_l1[i] [j]
        chroma_offset_l1[i] [j]
      }
  }
```

3530

```
if (nal_unit_for_MVC) {
  for( i = 0; i < num_multiview_refs_for_list1; i++ ) {
    luma_view_weight_l1_flag
    if( luma_weight_l1_flag ) {
      luma_view_weight_l1[i]
      luma_view_offset_l1[i]
    }
    chroma_view_weight_l1_flag
    if( chroma_view_weight_l1_flag )
      for( j = 0; j < 2; j++ ) {
        chroma_view_weight_l1[i] [j]
        chroma_view_offset_l1[i] [j]
      }
    }
  }
}
```

3540

FIG. 37

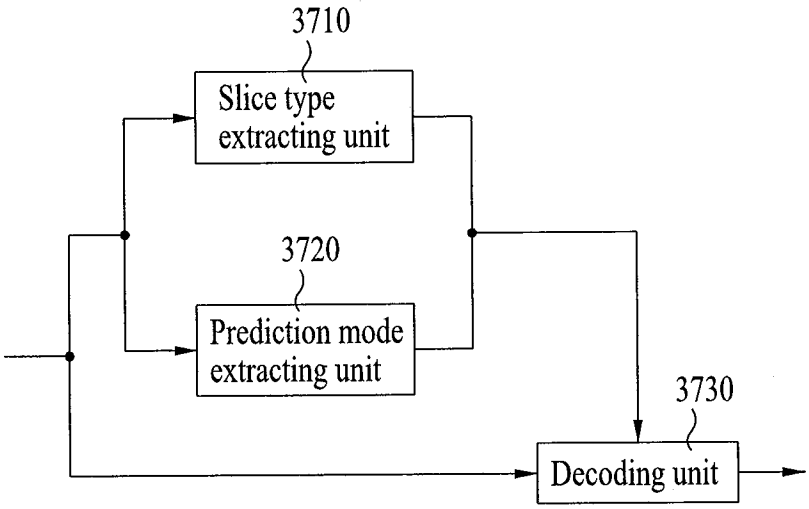

3710
Slice type extracting unit
3720
Prediction mode extracting unit
3730
Decoding unit

FIG. 38

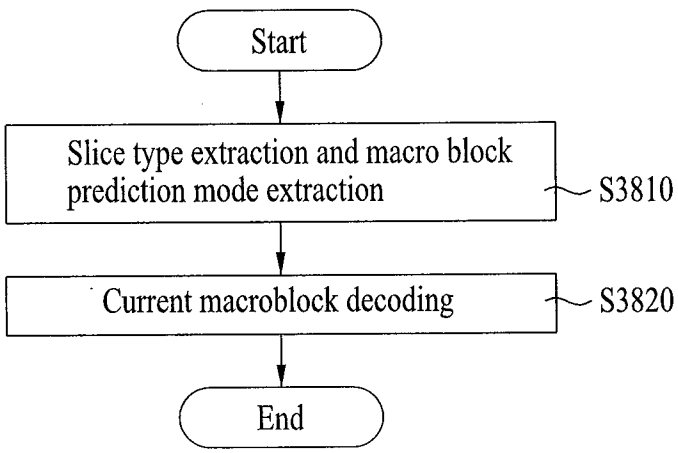

Start
Slice type extraction and macro block prediction mode extraction
S3810
Current macroblock decoding
S3820
End

FIG. 39

mb_pred_mode	description
0	Only temporal prediction is used for the macroblock List0/List1 shall be used for temporal prediction
1	Only inter-view prediction is used for the macroblock ViewList0/ViewList1 shall be used for inter-view prediction
2	Both temporal and inter-view prediction is used for the macroblock List0/List1 and ViewList0/ViewList1 shall be used for both prediction

(a)

mb_pred_mode	description	mb_type
0	Only temporal prediction List0 / List1	dependent to slice_type
1	Only inter-view prediction ViewList0 / ViewList1	dependent to view_slice_type which is similar with a slice_type for a temporal prediction
2	Mixed prediction (temporal + inter-view) List0 / List1 / ViewList0 / ViewList1	Dependent to slice_type and view_slice_type

(b)

FIG. 40

slice_layer_in_mvc_extension_rbsp( ) {	Descriptor
slice_header_in_mvc_extension( )	
slice_data_in_mvc_extension( )	
....	

slice_header_in_mvc_extension( ) {	Descriptor
....	
slice_type	ue(v)
view_slice_type	ue(v)
....	

slice_data_in_mvc_extension( ) {	Descriptor
....	
macroblock_layer_in_mvc_extension( )	
....	

Macroblock_layer_in_mvc_extension( ) {	Descriptor
mb_pred_mode	u(2)
mb_type	ue(v)| ae(v)
if(mb_pred_mode == 2) {	
mb_type	ue(v)| ae(v)
}	
....	

(a)

FIG. 41

slice_layer_in_mvc_extension_rbsp( ) {	Descriptor
slice_header_in_mvc_extension( )	
slice_data_in_mvc_extension( )	
....	

slice_header_in_mvc_extension( ) {	Descriptor
....	
slice_type	ue(v)
....	

slice_data_in_mvc_extension( ) {	Descriptor
....	
macroblock_layer_in_mvc_extension( )	
....	

Macroblock_layer_in_mvc_extension( ) {	Descriptor
mb_pred_mode	u(2)
mb_type	ue(v)| ae(v)
if(mb_pred_mode == 2) {	
mb_type	ue(v)| ae(v)
}	
....	

(b)

FIG. 42

slice_type	Name of slice type
0	P
1	B
2	I
3	SP
4	SI
5	P
6	B
7	I
8	SP
9	SI
10	VP
11	VB
12	Mixed

(a)

slice_type	Name of slice type
0	VB
1	VP
2	Mixed

(b)

slice_type	Name of slice type
0	EB(B slice in scalable extension)
1	EP(P slice in scalable extension)
2	EI(I slice in scalable extension)
3	PR(Progressive refinement slice)
4	VB
5	VP
6	Mixed

(c)

FIG. 43

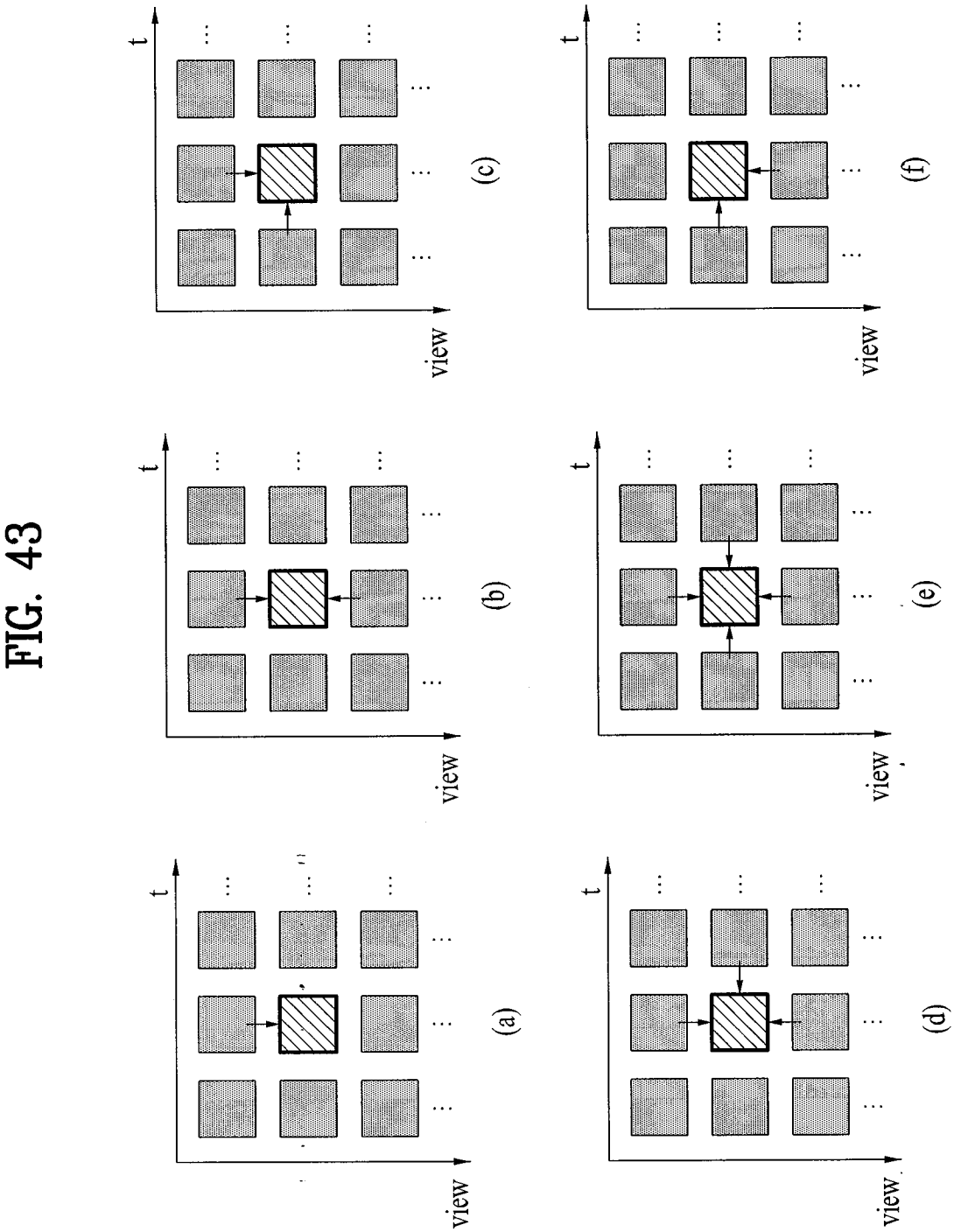
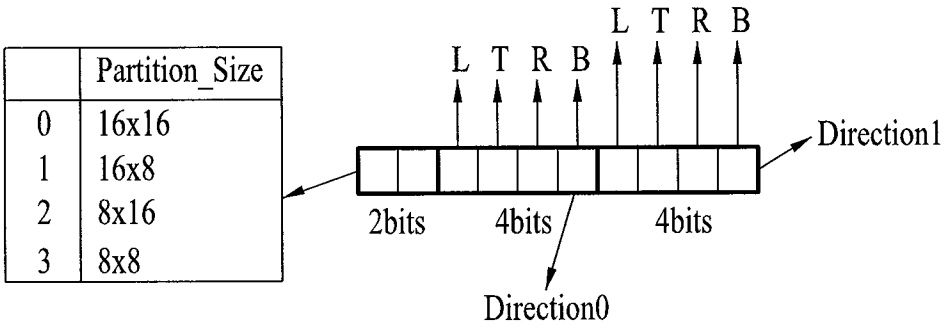


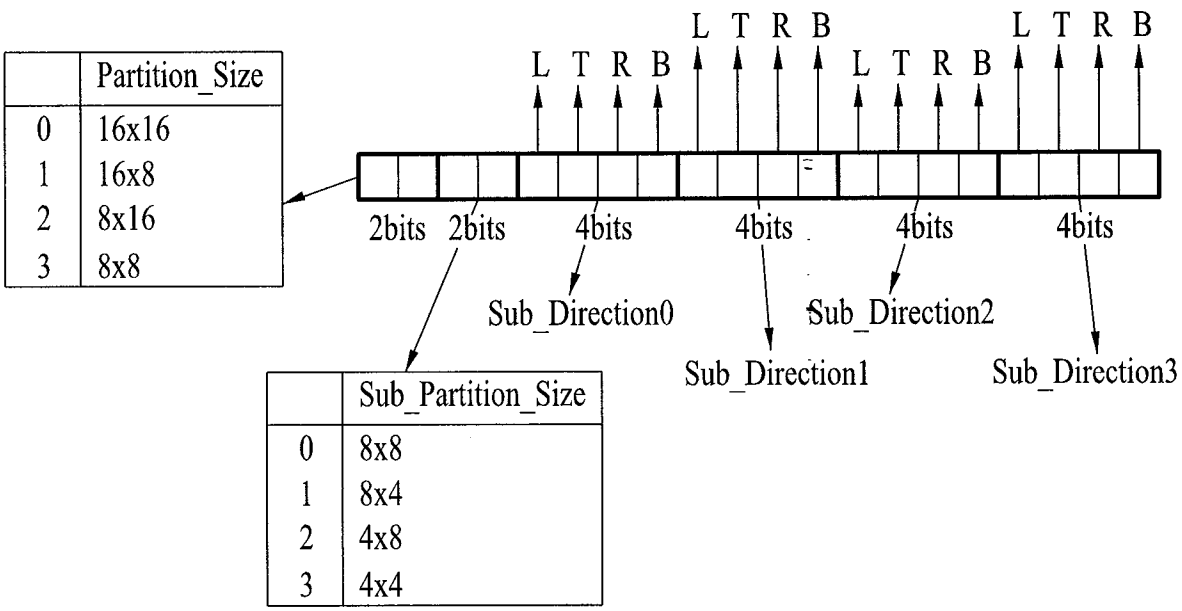
FIG. 44

Slice types	Description	allowed collective macroblock types
VP (View_P)	Each macroblock or macroblock partition is predicted from 'a picture in current view' or 'a picture in different view'	I, P, VP
VB (View_B)	Each macroblock or macroblock partition is predicted from 'a picture /pictures in current view' or 'a picture/ pictures in different view'	I, P, B, VP, VB
Mixed	Each macroblock or macroblock partition is predicted from a picture/pictures from current view or a picture/pictures in different view/views or 'a picture/pictures in current view' and 'a picture/pictures in different view/views'	I, P, B, VP, VB, Mixed

FIG. 45


Partition_Size
0 16x16
1 16x8
2 8x16
3 8x8
L T R B
L T R B
Direction1
2bits
4bits
4bits
Direction0

(a)


Partition_Size
0 16x16
1 16x8
2 8x16
3 8x8
L T R B
L T R B
L T R B
L T R B
2bits
2bits
4bits
4bits
4bits
4bits
Sub_Direction0
Sub_Direction1
Sub_Direction2
Sub_Direction3
Sub_Partition_Size
0 8x8
1 8x4
2 4x8
3 4x4

(b)

FIG. 46

L	1000
T	0100
R	0010
B	0001
LT	1100
LR	1010
LB	1001
TR	0110
TB	0101
RB	0011
LTR	1110
LTB	1101
LRB	1011
TRB	0111
LTRB	1111

(c)

FIG. 47

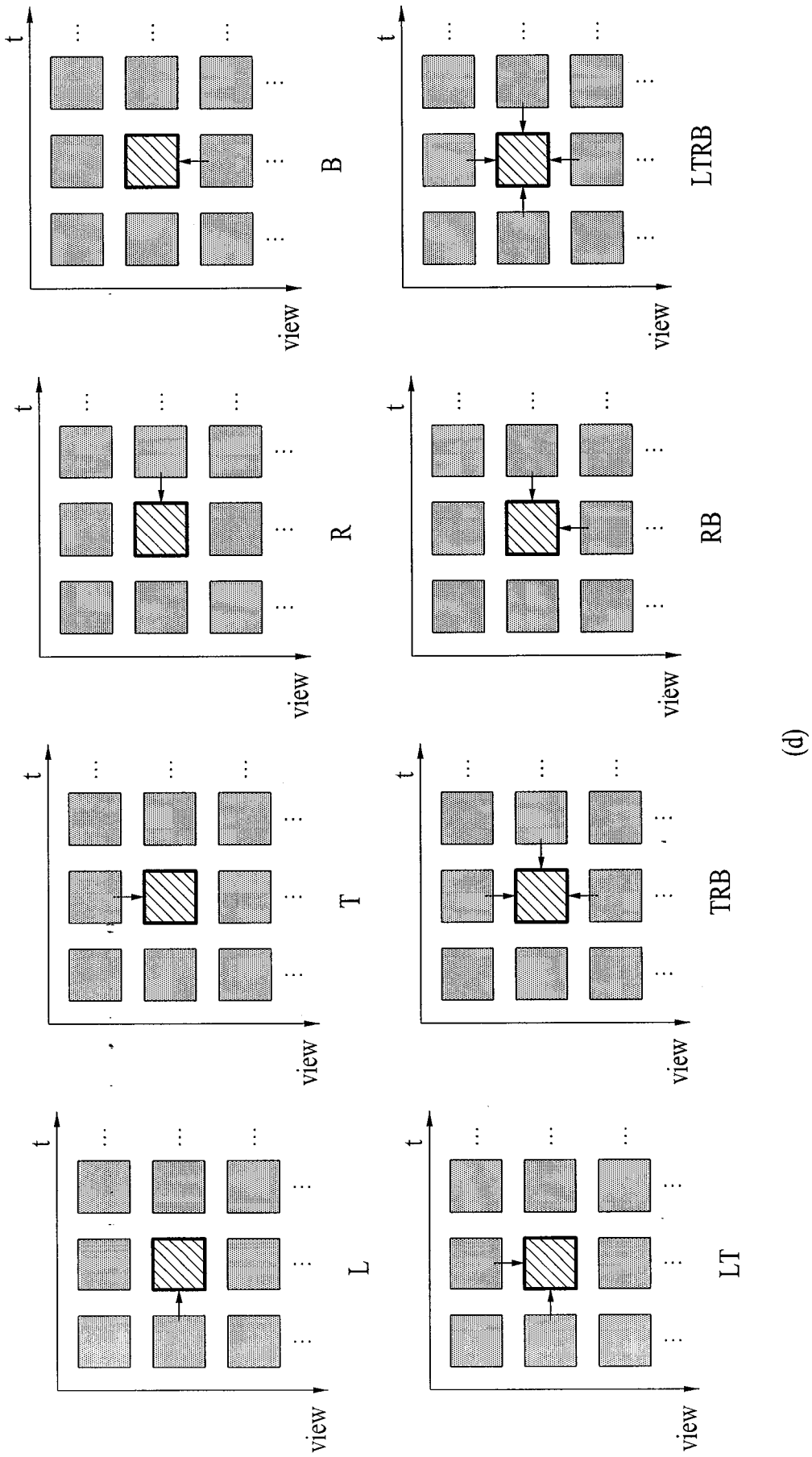


FIG. 48

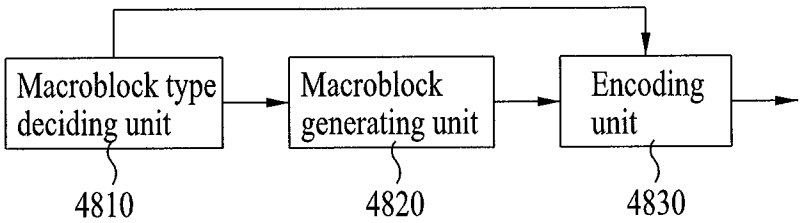
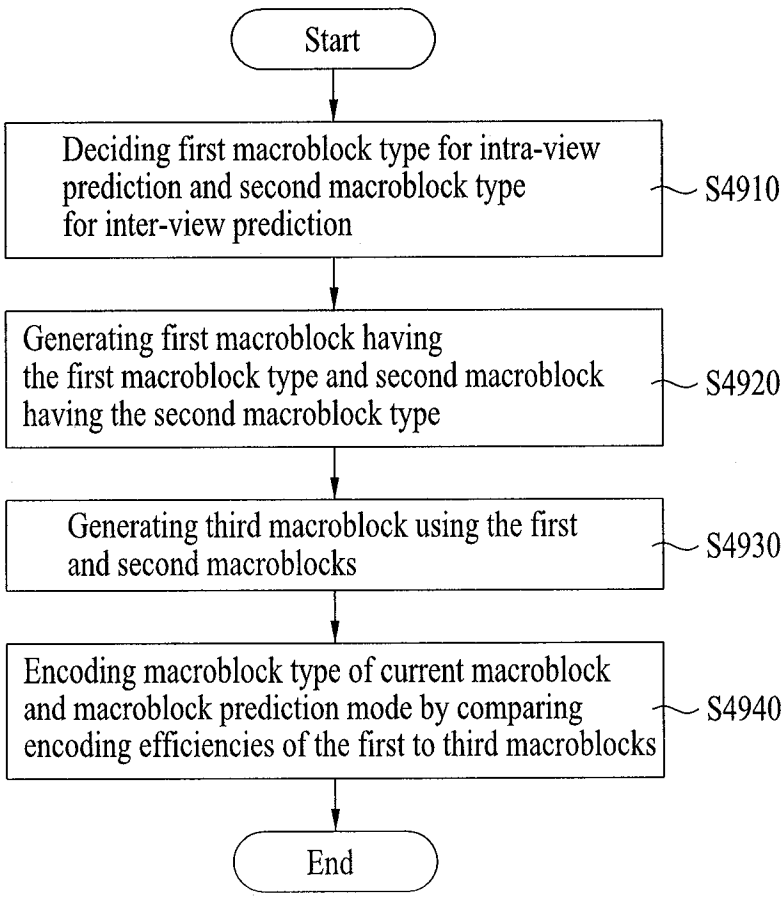

Macroblock type deciding unit
Macroblock generating unit
Encoding unit
4810
4820
4830

FIG. 49


Start
Deciding first macroblock type for intra-view prediction and second macroblock type for inter-view prediction
S4910
Generating first macroblock having the first macroblock type and second macroblock having the second macroblock type
S4920
Generating third macroblock using the first and second macroblocks
S4930
Encoding macroblock type of current macroblock and macroblock prediction mode by comparing encoding efficiencies of the first to third macroblocks
S4940
End

INTERNATIONAL SEARCH REPORT

International application No.
PCT/KR2007/001581

A. CLASSIFICATION OF SUBJECT MATTER

H04N 7/24(2006.01)i, H04N 7/32(2006.01)i, H04N 7/36(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 8 H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean Utility models and applications for Utility Models: IPC as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKIPASS(KIPO internal): "multi-view, encoding"

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2002/0146239 A1 (HIROSHI HAMASAKA, et al.) 10. Oct. 2002. See abstract, page 1, [0012] - page 1, [0015].	1-11
A	WO 2006/001653 A1 (LG ELECTRONICS CO., LTD.) 5. Jan. 2006. See abstract, page 1, [1] - page 7, [53].	1-11

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
06 AUGUST 2007 (06.08.2007)	**06 AUGUST 2007 (06.08.2007)**

Name and mailing address of the ISA/KR
Korean Intellectual Property Office
920 Dunsan-dong, Seo-gu, Daejeon 302-701,
Republic of Korea
Facsimile No. 82-42-472-7140

Authorized officer
CHO, Woo Yeon
Telephone No. 82-42-481-8524

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/KR2007/001581

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US20020146239A1	10.10.2002	BR200206500A	13.01.2004
		CA2434424AA	18.07.2002
		CN1489765A	14.04.2004
		CZ20031749A3	14.01.2004
		EP01363291A1	19.11.2003
		HU200303862AB	01.03.2004
		JP15018549	17.01.2003
		JP3820155B2	13.09.2006
		KR1020030069207	25.08.2003
		MXPA03006194A	11.12.2003
		RU2003125269A	20.02.2005
		RU2291501C2	10.01.2007
		US20020146239A1	10.10.2002
		WO2002056314A1	18.07.2002
WO2006001653A1	05.01.2006	EP01772022A1	11.04.2007
		KR2005122717A	29.12.2005