

DESCRIPTION

IN-LOOP FILTER-BASED IMAGE ENCODING/DECODING METHOD AND APPARATUS

TECHNICAL FIELD

[0001] The present invention relates to an image encoding/decoding method and apparatus.

BACKGROUND ART

[0002] Recently, demand for high-resolution and high-quality images such as High Definition (HD) images and Ultra High Definition (UHD) images is increasing in various application fields, and accordingly, high-efficiency image compression techniques are being discussed.

[0003] For image compression technology, various technologies such as inter prediction technology that predicts pixel value included in the current picture from a picture before or after the current picture using image compression technology, intra prediction technology that predicts pixel value included in the current picture by using pixel information in the current picture, an entropy encoding technology that allocates a short code to a value with a high frequency of appearance and a long code to a value with a low frequency of appearance, exist, and by using such the image compression technology, image data can be effectively

compressed and transmitted or stored.

DISCLOSURE

TECHNICAL PROBLEM

[0004] An object of the present invention is to provide a method and apparatus for dividing a picture into a predetermined division unit.

[0005] An object of the present invention is to provide a method and apparatus for adaptively performing filtering on a boundary of a division unit.

[0006] An object of the present invention is to provide a method and apparatus for applying an improved deblocking filter.

TECHNICAL SOLUTION

[0007] The video decoding method and apparatus according to the present invention may divide one picture into a plurality of division units, determine whether to perform filtering on a boundary of a current division unit based on a predetermined flag, and perform filtering on the boundary of the current division unit in response to the determination.

[0008] In the video decoding method and apparatus according to the present invention, the division unit may include at least one of a sub picture, a slice, or a tile.

[0009] In the video decoding method and apparatus according to the present invention, the flag may include at least one of a first flag indicating whether filtering is performed on a boundary of a division unit within the one picture or a second flag indicating whether filtering is performed on the boundary of the current division unit in the one picture.

[0010] In the video decoding method and apparatus according to the present invention, when the first flag is a first value, it may not be restricted so that filtering is not performed on the boundary of the division unit within the one picture, and when the first flag is a second value, the restriction on the boundary of the division unit within the picture may not be imposed.

[0011] In the video decoding method and apparatus according to the present invention, when the second flag is the first value, it may be restricted so that filtering is not performed on the boundary of the current division unit, and when the second flag is the second value, filtering may be allowed to be performed on the boundary of the current division unit.

[0012] In the video decoding method and apparatus according to the present invention, the second flag may be decoded only when it is not restricted so that filtering is not performed on the boundary of the division unit within the one picture according to the first flag.

[0013] In the video decoding method and apparatus according to

the present invention, whether to perform filtering on the boundary of the current division unit may be determined by further considering a third flag indicating whether filtering is performed on a boundary of a neighboring division unit adjacent to the current block unit.

[0014] In the video decoding method and apparatus according to the present invention, a position of the neighboring division unit may be determined based on whether the boundary of the current division unit is a vertical boundary or a horizontal boundary.

[0015] In the video decoding method and apparatus according to the present invention, the step of performing the filtering may comprise, specifying a block boundary for deblocking filtering, deriving a decision value for the block boundary, determining a filter type for the deblocking filtering based on the decision value, and performing filtering on the block boundary based on the filter type.

[0016] The video encoding method and apparatus according to the present invention may divide one picture into a plurality of division units, determine whether to perform filtering on a boundary of a current division unit, and perform filtering on the boundary of the current division unit in response to the determination.

[0017] In the video encoding method and apparatus according to the present invention, the division unit may include at least one

of a sub picture, a slice, or a tile.

[0018] In the video encoding method and apparatus according to the present invention, the step of determining whether to perform filtering on the boundary of the current division unit may comprise, encoding at least one of a first flag indicating whether filtering is performed on a boundary of a division unit within the one picture or a second flag indicating whether filtering is performed on the boundary of the current division unit in the one picture.

[0019] In the video encoding method and apparatus according to the present invention, when it is determined that filtering is restricted not to be performed on the boundary of the division unit within the one picture, the first flag may be encoded as a first value, and when it is determined that the restriction is not imposed on the boundary of the division unit within the picture, the first flag may be encoded as a second value.

[0020] In the video encoding method and apparatus according to the present invention, when it is determined that filtering is restricted not to be performed on the boundary of the current division unit, the second flag may be encoded as the first value, and when it is determined that filtering is allowed to be performed on the boundary of the current division unit, the second flag may be encoded as the second value.

[0021] In the video encoding method and apparatus according to the present invention, the second flag may be encoded only when it

is not restricted so that filtering is not performed on the boundary of the division unit within the one picture.

[0022] In the video encoding method and apparatus according to the present invention, whether to perform filtering on the boundary of the current division unit may be determined by further considering a third flag indicating whether filtering is performed on a boundary of a neighboring division unit adjacent to the current block unit.

[0023] In the video encoding method and apparatus according to the present invention, a position of the neighboring division unit may be determined based on whether the boundary of the current division unit is a vertical boundary or a horizontal boundary.

[0024] In the video encoding method and apparatus according to the present invention, wherein the step of performing the filtering comprises, specifying a block boundary for deblocking filtering, deriving a decision value for the block boundary, determining a filter type for the deblocking filtering based on the decision value, and performing filtering on the block boundary based on the filter type.

ADVANTAGEOUS EFFECTS

[0025] According to the present invention, encoding/decoding efficiency can be improved by dividing one picture into various division units.

[0026] According to the present invention, encoding/decoding efficiency can be improved by adaptively performing filtering on the boundary of the division unit.

[0027] According to the present invention, image quality can be improved by applying an improved deblocking filter to a reconstructed image.

DESCRIPTION OF DRAWINGS

[0028] FIG. 1 is a block diagram showing an image encoding apparatus according to an embodiment of the present invention.

[0029] FIG. 2 is a block diagram showing an image decoding apparatus according to an embodiment of the present invention.

[0030] FIG. 3 to 6 illustrate a method of dividing one picture into one or more division units according to the present disclosure.

[0031] FIG. 7 illustrates a method of performing filtering based on a predetermined flag according to the present disclosure.

[0032] FIG. 8 to 15 illustrate a method of determining whether filtering is performed on a boundary of a division unit based on one or more flags according to the present disclosure.

[0033] FIG. 16 illustrates a method of applying a deblocking filter according to the present disclosure.

BEST MODE FOR INVENTION

[0034] An image decoding method and apparatus of the present invention may divide one picture into a plurality of division units, determine whether to perform filtering on a boundary of a current division unit based on a predetermined flag, and perform filtering on the boundary of the current division unit in response to the determination.

[0035] In the image decoding method and apparatus of the present invention, the division unit may include at least one of a sub picture, a slice, or a tile.

[0036] In the image decoding method and apparatus of the present invention, the flag may include at least one of a first flag indicating whether filtering is performed on a boundary of a division unit within the one picture or a second flag indicating whether filtering is performed on the boundary of the current division unit in the one picture.

[0037] In the image decoding method and apparatus of the present invention, when the first flag is a first value, it may be restricted so that filtering is not performed on the boundary of the division unit within the one picture, and when the first flag is a second value, the restriction on the boundary of the division unit within the picture may not be imposed.

[0038] In the image decoding method and apparatus of the present invention, when the second flag is the first value, it may be

restricted so that filtering is not performed on the boundary of the current division unit, and when the second flag is the second value, filtering may be performed on the boundary of the current division unit.

[0039] In the image decoding method and apparatus of the present invention, the second flag may be decoded only when it is not restricted so that filtering is not performed on the boundary of the division unit within the one picture according to the first flag.

[0040] In the image decoding method and apparatus of the present invention, whether to perform filtering on the boundary of the current division unit may be determined by further considering a third flag indicating whether filtering is performed on a boundary of a neighboring division unit adjacent to the current block unit.

[0041] In the image decoding method and apparatus of the present invention, a position of the neighboring division unit may be determined based on whether the boundary of the current division unit is a vertical boundary or a horizontal boundary.

[0042] In the image decoding method and apparatus of the present invention, performing the filtering may comprise specifying a block boundary for deblocking filtering, deriving a decision value for the block boundary, determining a filter type for the deblocking filtering based on the decision value, and performing the filtering on the block boundary based on the filter type.

[0043] An image encoding method and apparatus of the present invention may divide one picture into a plurality of division units, determine whether to perform filtering on a boundary of a current division unit, and perform filtering on the boundary of the current division unit in response to the determination.

[0044] In the image encoding method and apparatus of the present invention, the division unit may include at least one of a sub picture, a slice, or a tile.

[0045] In the image encoding method and apparatus of the present invention, determining whether to perform filtering on the boundary of the current division unit may comprise encoding at least one of a first flag indicating whether filtering is performed on a boundary of a division unit within the one picture or a second flag indicating whether filtering is performed on the boundary of the current division unit in the one picture.

[0046] In the image encoding method and apparatus of the present invention, when it is determined that filtering is restricted not to be performed on the boundary of the division unit within the one picture, the first flag may be encoded as a first value, and when it is determined that the restriction is not imposed on the boundary of the division unit within the picture, the first flag may be encoded as a second value,

[0047] In the image encoding method and apparatus of the present invention, when it is determined that filtering is restricted not

to be performed on the boundary of the current division unit, the second flag may be encoded as the first value, and when it is determined that filtering is allowed to be performed on the boundary of the current division unit, the second flag may be encoded as the second value.

[0048] In the image encoding method and apparatus of the present invention, the second flag may be encoded only when it is not restricted so that filtering is not performed on the boundary of the division unit within the one picture.

[0049] In the image encoding method and apparatus of the present invention, whether to perform filtering on the boundary of the current division unit may be determined by further considering a third flag indicating whether filtering is performed on a boundary of a neighboring division unit adjacent to the current block unit.

[0050] In the image encoding method and apparatus of the present invention, a position of the neighboring division unit may be determined based on whether the boundary of the current division unit is a vertical boundary or a horizontal boundary.

[0051] In the image encoding method and apparatus of the present invention, performing the filtering may comprise specifying a block boundary for deblocking filtering, deriving a decision value for the block boundary, determining a filter type for the deblocking filtering based on the decision value, and performing the filtering on the block boundary based on the filter type.

MODE FOR INVENTION

[0052] In the present invention, various modifications may be made and various embodiments may be provided, and specific embodiments will be illustrated in the drawings and described in detail in the detailed description. However, this is not intended to limit the present invention to a specific embodiment, it should be understood to include all changes, equivalents, and substitutes included in the idea and scope of the present invention. In describing each drawing, similar reference numerals have been used for similar elements.

[0053] Terms such as first and second may be used to describe various components, but the components should not be limited by the terms. These terms are used only for the purpose of distinguishing one component from another component. For example, without departing from the scope of the present invention, a first component may be referred to as a second component, and similarly, a second component may be referred to as a first component. The term and/or includes a combination of a plurality of related listed items or any of a plurality of related listed items.

[0054] When a component is referred to as being "connected" or "connected" to another component, it may be directly connected or connected to the another component, but it should be understood that other component may exist in the middle. On the other hand, when a component is referred to as being "directly connected" or

"directly connected" to another component, it should be understood that there is no other component in the middle.

[0055] The terms used in the present application are only used to describe specific embodiments, and are not intended to limit the present invention. Singular expressions include plural expressions unless the context clearly indicates otherwise. In the present application, terms such as "comprise" or "have" are intended to designate the presence of features, numbers, steps, actions, components, parts, or combinations thereof described in the specification, but one or more other features. It should be understood that the presence or addition of elements or numbers, steps, actions, components, parts, or combinations thereof, does not preclude in advance.

[0056] Hereinafter, preferred embodiments of the present invention will be described in more detail with reference to the accompanying drawings. Hereinafter, the same reference numerals are used for the same elements in the drawings, and duplicate descriptions for the same elements are omitted.

[0057]

[0058] FIG. 1 is a block diagram showing an image encoding apparatus according to an embodiment of the present invention.

[0059] Referring to FIG. 1, the image encoding apparatus 100 may include a picture division unit 110, a prediction unit 120, 125, a transform unit 130, a quantization unit 135, a rearrangement

unit 160, and an entropy encoding unit 165, an inverse quantization unit 140, an inverse transform unit 145, a filter unit 150, and a memory 155.

[0060] Each of the components shown in FIG. 1 is independently shown to represent different characteristic functions in an image encoding apparatus, and does not mean that each component is formed of separate hardware or a single software component. That is, each constituent part is listed and included as a respective constituent part for convenience of explanation, and at least two constituent parts of each constituent part are combined to form one constituent part, or one constituent part may be divided into a plurality of constituent parts to perform a function. Integrated embodiments and separate embodiments of the components are also included in the scope of the present invention unless departing from the essence of the present invention.

[0061] In addition, some of the components may not be essential components that perform essential functions in the present invention, but may be optional components only for improving performance. The present invention may be implemented by including only components essential to implement the essence of the present invention excluding components used for performance improvement, and a structure including only essential components excluding optional components used for performance improvement may be also included in the scope of the present invention.

[0062] The picture division unit 110 may divide the input picture into at least one processing unit. In this case, the processing unit may be a prediction unit (PU), a transform unit (TU), or a coding unit (CU). The picture division unit 110 may encode the picture by dividing into a combination of a plurality of coding units, prediction units, and transform units, and selecting the combination of a coding unit, a prediction unit, and a transformation unit based on a predetermined criterion (for example, a cost function).

[0063] For example, one picture may be divided into a plurality of coding units. In order to split the coding units in a picture, a recursive tree structure such as a quad tree structure may be used, and a coding unit that is split into other coding unit based on one image or the largest coding unit as a root may be divided with as many child nodes as the number of divided coding units. Coding units that are no longer split according to certain restrictions become leaf nodes. That is, when it is assumed that only square splitting is possible for one coding unit, one coding unit may be split into up to four different coding units.

[0064] Hereinafter, in an embodiment of the present invention, a coding unit may be used as a meaning of a unit that performs encoding, or may be used as a meaning of a unit that performs decoding.

[0065] The prediction unit may be obtained by dividing one coding

unit into at least one square or non-square shape of the same size. One coding unit may be divided such that one prediction unit of prediction units has a different shape and/or size from another prediction unit.

[0066] When a prediction unit that performs intra prediction based on a coding unit is not a minimum coding unit, intra prediction may be performed without dividing into a plurality of $N \times N$ prediction units.

[0067] The prediction units 120 and 125 may include an inter prediction unit 120 that performs inter prediction, and an intra prediction unit 125 that performs intra prediction. Whether to use inter prediction or intra prediction for a prediction unit may be determined, and specific information (e.g., intra prediction mode, motion vector, reference picture, etc.) according to each prediction method may be determined. In this case, a processing unit in which prediction is performed may be different from a processing unit in which a prediction method and specific content are determined. For example, a prediction method and a prediction mode are determined in a prediction unit, and prediction may be performed in a transformation unit. The residual value (residual block) between the generated prediction block and the original block may be input to the transform unit 130. In addition, prediction mode information, motion vector information, and the like used for prediction may be encoded by the entropy encoding

unit 165 together with the residual value and transmitted to the decoder. In the case of using a specific encoding mode, it is possible to encode an original block as it is and transmit it to a decoder without generating a prediction block through the prediction units 120 and 125.

[0068] The inter prediction unit 120 may predict a prediction unit based on information of at least one of a previous picture or a subsequent picture of the current picture, and in some cases, predict a prediction unit based on information of some regions, which encoding has been completed, in the current picture. The inter prediction unit 120 may include a reference picture interpolation unit, a motion prediction unit, and a motion compensation unit.

[0069] The reference picture interpolation unit may receive reference picture information from the memory 155 and generate pixel information of an integer pixel or less in the reference picture. In the case of a luma pixel, a DCT-based 8-tap interpolation filter (DCT-based interpolation filter) having different filter coefficients may be used to generate pixel information of an integer pixel or less in units of a $1/4$ pixels. In case of a chroma signal, a DCT-based 4-tap interpolation filter (DCT-based interpolation filter) having different filter coefficients may be used to generate pixel information of an integer pixel or less in units of $1/8$ pixels.

[0070] The motion prediction unit may perform motion prediction based on the reference picture interpolated by the reference picture interpolation unit. As a method for calculating the motion vector, various methods such as Full Search-based Block Matching Algorithm (FBMA), Three Step Search (TSS), and New Three-Step Search Algorithm (NTS) may be used. The motion vector may have a motion vector value in units of $1/2$ or $1/4$ pixels based on the interpolated pixels. The motion prediction unit may predict a current prediction unit by differently using a motion prediction method. Various methods such as a skip method, a merge method, an AMVP (Advanced Motion Vector Prediction) method, and an intra block copy method may be used as the motion prediction method.

[0071] The intra prediction unit 125 may generate a prediction unit based on reference pixel information around a current block, which is pixel information in a current picture. When the neighboring block of the current prediction unit is a block that performs inter prediction and the reference pixel is a pixel that performs inter prediction, the reference pixel included in the block that performs inter prediction may be used by replacing it with reference pixel information of a block that performs intra prediction around it. That is, when the reference pixel is not available, the unavailable reference pixel information may be used by replacing with at least one reference pixel among the available reference pixels.

[0072] In intra prediction, the prediction mode may have a directional prediction mode in which reference pixel information is used according to a prediction direction and a non-directional mode in which directional information is not used when prediction is performed. A mode for predicting luma information and a mode for predicting chroma information may be different, and intra prediction mode information or predicted luma signal information used to predict luma information may be used to predict chroma information.

[0073] When the size of the prediction unit and the size of the transformation unit are the same when performing intra prediction, intra prediction for the prediction unit may be performed based on a pixel on the left, a pixel on the above left, and a pixel on the above of the prediction unit. However, when the size of the prediction unit and the size of the transformation unit are different when performing intra prediction, intra prediction may be performed using a reference pixel based on the transformation unit. Also, intra prediction using $N \times N$ splitting may be used for only the minimum coding unit.

[0074] The intra prediction method may generate a prediction block after applying an AIS (Adaptive Intra Smoothing) filter to a reference pixel according to a prediction mode. The types of AIS filters applied to the reference pixels may be different. In order to perform the intra prediction method, the intra prediction mode

of the current prediction unit may be predicted from the intra prediction mode of the prediction unit existing around the current prediction unit. When predicting the prediction mode of the current prediction unit using the mode information predicted from the neighboring prediction unit, if the intra prediction mode of the current prediction unit and the neighboring prediction unit are the same, information indicating that the prediction mode of the current prediction unit and the neighboring prediction units are the same may be transmitted using predetermined flag information, and if the prediction modes of the current prediction unit and the neighboring prediction units are different, entropy encoding may be performed to encode prediction mode information of the current block.

[0075] In addition, a residual block including residual information that is a difference value between the prediction unit that performs prediction based on the prediction units generated by the prediction units 120 and 125 and the original block of the prediction unit may be generated. The generated residual block may be input to the transform unit 130.

[0076] The transform unit 130 may transform a residual block including residual information between a prediction unit generated by the prediction units 120 and 125 and the original block by using the transform method such as DCT (Discrete Cosine Transform), DST (Discrete Sine Transform), and KLT. Whether DCT, DST, or KLT is

applied to transform the residual block may be determined based on intra prediction mode information of a prediction unit used to generate the residual block.

[0077] The quantization unit 135 may quantize values transformed to the frequency domain by the transform unit 130. The quantization coefficient may vary depending on the block or the importance of the image. The value calculated by the quantization unit 135 may be provided to the inverse quantization unit 140 and the rearrangement unit 160.

[0078] The rearrangement unit 160 may perform the rearrangement of the coefficient value for the quantized residual value.

[0079] The rearrangement unit 160 may change coefficients of 2-dimensional block form into 1-dimensional vector form through a coefficient scanning method. For example, the rearrangement unit 160 may change into a 1-dimensional vector form by scanning from a DC coefficient to a coefficient in a high frequency region according to a Zig-Zag Scan method. Depending on the size of the transform unit and the intra prediction mode, a vertical scan of scanning coefficients of two-dimensional block form in a column direction and a horizontal scan of scanning coefficients of two-dimensional block form in a row direction may be used instead of a zig-zag scan. That is, depending on the size of the transform unit and the intra prediction mode, it may be determined which one of a zigzag scan, a vertical scan, and a horizontal scan is used.

[0080] The entropy encoding unit 165 may perform entropy-encoding based on values calculated by the rearrangement unit 160. Various encoding methods, such as exponential Golomb, CAVLC (Context-Adaptive Variable Length Coding), and CABAC (Context-Adaptive Binary Arithmetic Coding), may be used for entropy-encoding.

[0081] The entropy encoder 165 may encode various information, such as residual value coefficient information, block type information, prediction mode information, division unit information, prediction unit information, transmission unit information, motion vector information, reference frame information, block interpolation information, filtering information, etc.

[0082] The entropy encoder 165 may entropy-encode a coefficient value of a coding unit input from the reordering unit 160.

[0083] The inverse quantization unit 140 and the inverse transform unit 145 inverse-quantize the values quantized by the quantization unit 135 and inverse-transform the values transformed by the transform unit 130. The reconstructed block may be generated by combining the residual value generated by the inverse quantization unit 140 and the inverse transform unit 145 with the prediction unit predicted through the motion estimation unit, the motion compensation unit, and the intra prediction unit included in the prediction units 120 and 125.

[0084] The filter unit 150 may include at least one of a

deblocking filter, an offset correction unit, and an adaptive loop filter (ALF).

[0085] The deblocking filter may remove block distortion caused by boundary between blocks in the reconstructed picture. In order to determine whether to perform deblocking, it may be determined whether to apply the deblocking filter to the current block based on the pixels included in several columns or rows included in the block. When applying a deblocking filter to a block, a strong filter or a weak filter may be applied according to the required deblocking filtering strength. In addition, in applying the deblocking filter, horizontal filtering and vertical filtering may be processed in parallel when performing vertical filtering and horizontal filtering.

[0086] The offset correction unit may correct an offset from the original image in units of pixels for the deblocking-filtered image. In order to perform offset correction for a specific picture, after classifying the pixels included in the image into a certain number of regions and determining the region to which the offset is applied, a method of applying the offset to the region offset or a method of applying the offset by considering edge information of each pixel may be used.

[0087] ALF (Adaptive Loop Filtering) may be performed based on a value obtained by comparing a filtered reconstructed image with an original image. After classifying the pixels included in the image

into a predetermined group, one filter to be applied to the group may be determined to perform filtering differently for each group. Information related to whether to apply ALF may be transmitted for each coding unit (CU) of a luma signal, and a shape and filter coefficient of an ALF filter to be applied may vary according to each block. In addition, the same type (fixed type) ALF filter may be applied regardless of the characteristics of the block to be applied.

[0088] The memory 155 may store the reconstructed block or picture output from the filter unit 150, and the stored reconstructed block or picture may be provided to the prediction units 120 and 125 when performing inter prediction.

[0089]

[0090] FIG. 2 is a block diagram showing an image decoding apparatus according to an embodiment of the present invention.

[0091] Referring to FIG. 2, the image decoder 200 may include an entropy decoding unit 210, a rearrangement unit 215, an inverse quantization unit 220, an inverse transform unit 225, a prediction unit 230, 235, and a filter unit 240, a memory 245.

[0092] When an image bitstream is input from the image encoder, the input bitstream may be decoded in a procedure opposite to that of the image encoder.

[0093] The entropy decoding unit 210 may perform entropy-decoding in a procedure opposite to that performed by entropy-encoding in

the entropy encoding unit of the image encoder. For example, various methods corresponding to the method performed in the image encoder such as Exponential Golomb (CAVLC), Context-Adaptive Variable Length Coding (CAVLC), and Context-Adaptive Binary Arithmetic Coding (CABAC) may be applied.

[0094] The entropy decoding unit 210 may decode information related to intra prediction and inter prediction performed by the encoder.

[0095] The rearrangement unit 215 may perform rearrangement of the bitstream entropy-decoded by the entropy decoding unit 210 based on a rearrangement method of the encoding unit. The coefficients of a 1-dimensional vector form may be rearranged into coefficients of a 2-dimensional block form again. The rearrangement unit 215 may perform reordering through a method of receiving information related to coefficient scanning performed by the encoder and performing reverse scanning based on the scanning order performed by the corresponding encoder.

[0096] The inverse quantization unit 220 may perform inverse quantization based on the quantization parameter provided by the encoder and the coefficients of the rearranged block.

[0097] The inverse transform unit 225 may perform inverse transform, that is, inverse DCT, inverse DST, and inverse KLT, corresponding to transforms performed by the transform unit, that is, DCT, DST, and KLT for the quantization results performed by

the image encoder. The inverse transform may be performed based on the transmission unit determined by the image encoder. In the inverse transform unit 225 of the image decoder, a transform method (for example, DCT, DST, KLT) may be selectively performed according to a plurality of information such as a prediction method, a size of a current block, and a prediction direction.

[0098] The prediction units 230 and 235 may generate a prediction block based on prediction block generation related information provided by the entropy decoding unit 210 and previously decoded block or picture information provided by the memory 245.

[0099] As described above, when a size of the prediction unit and a size of the transform unit are the same in performing intra prediction in the same manner as in the image encoder, the intra prediction of the prediction unit may be performed based on pixels located on the left, the top-left and the top of the prediction unit. However, when the size of the prediction unit and the size of the transform unit are different in performing intra prediction, the intra prediction may be performed using a reference pixel based on the transform unit. In addition, the intra prediction using NxN division may be used only for the minimum coding unit.

[0100] The prediction units 230 and 235 may include a prediction unit determination unit, an inter prediction unit, and an intra prediction unit. The prediction unit determination unit may receive various information from the entropy decoding unit 210

such as prediction unit information, prediction mode information of an intra prediction method, and motion prediction related information of an inter prediction method, classify the prediction unit from the current coding unit, and determine whether the prediction unit performs inter prediction or intra prediction. The inter prediction unit 230 may perform inter prediction for a current prediction unit based on information included in at least one of a previous picture or a subsequent picture of the current picture including the current prediction unit, by using information required for inter prediction of the current prediction unit provided by the image encoder. Alternatively, inter prediction may be performed based on information on a partial region previously-reconstructed in the current picture including the current prediction unit.

[0101] In order to perform inter prediction, a motion prediction method of a prediction unit included in a coding unit may be determined among a skip mode, a merge mode, an AMVP mode, and an intra block copy mode.

[0102] The intra prediction unit 235 may generate a prediction block based on pixel information in the current picture. When the prediction unit is a prediction unit that has performed intra prediction, intra prediction may be performed based on intra prediction mode information of a prediction unit provided by an image encoder. The intra prediction unit 235 may include an

adaptive intra smoothing (AIS) filter, a reference pixel interpolation unit, and a DC filter. The AIS filter is a part that performs filtering on the reference pixel of the current block and may be applied by determining whether to apply the filter according to the prediction mode of the current prediction unit. AIS filtering may be performed on a reference pixel of a current block by using prediction mode and AIS filter information of a prediction unit provided by an image encoder. When the prediction mode of the current block is a mode that does not perform AIS filtering, the AIS filter may not be applied.

[0103] When the prediction mode of the prediction unit is the prediction unit that performs intra prediction based on the pixel value obtained by interpolating the reference pixel, the reference pixel interpolation unit may interpolate the reference pixel to generate a reference pixel of an integer pixel or less. When the prediction mode of the current prediction unit is a prediction mode in which a prediction block is generated without interpolating a reference pixel, the reference pixel may not be interpolated. The DC filter may generate a prediction block through filtering when the prediction mode of the current block is the DC mode.

[0104] The reconstructed block or picture may be provided to the filter unit 240. The filter unit 240 may include a deblocking filter, an offset correction unit, and an ALF.

[0105] Information about whether a deblocking filter is applied

to a corresponding block or picture and information about whether a strong filter is applied or a weak filter is applied in applying the deblocking filter may be provided from a video encoder. In the deblocking filter of the video decoder, information related to the deblocking filter provided by the video encoder may be provided, and the video decoder may perform deblocking filtering on the corresponding block.

[0106] The offset correction unit may perform offset correction on the reconstructed image based on a type of offset correction and offset value information applied to the image during encoding.

[0107] ALF may be applied to a coding unit based on information on whether to apply ALF, ALF coefficient information, and the like, provided by an encoder. This ALF information may be provided from a specific parameter set.

[0108] The memory 245 may store the reconstructed picture or block so that it can be used as a reference picture or a reference block, and may also provide the reconstructed picture to an output unit.

[0109] As described above, in an embodiment of the present invention, for convenience of description, a coding unit is used as a coding unit, but it may be a unit that performs not only encoding but also decoding.

[0110]

[0111] FIG. 3 to 6 illustrate a method of dividing one picture into one or more division units according to the present disclosure.

[0112] One picture may be divided into division units pre-defined in the encoding/decoding apparatus. Here, the pre-defined division unit may include at least one of a sub picture, a slice, a tile, or a brick.

[0113] Specifically, one picture may be divided into one or more tile rows or one or more tile columns. In this case, the tile may mean a group of blocks covering a predetermined rectangular area in the picture. Here, the block may mean a coding tree block (largest coding block). Coding tree blocks belonging to one tile may be scanned based on a raster scan order.

[0114] Tiles may be divided into one or more bricks. Bricks may be composed of blocks in rows or columns of tiles. That is, the division into bricks may be performed only in either a vertical direction or a horizontal direction. However, the present invention is not limited thereto, and one tile may be divided into a plurality of bricks based on one or more vertical lines and one or more horizontal lines. Brick may be a sub-concept of a tile, and may be called a sub-tile.

[0115] One slice may include one or more tiles. Alternatively, one slice may include one or more bricks. Alternatively, one slice may be defined as one or more coding tree block rows (CTU rows) in a tile. Alternatively, one slice may be defined as one or more coding tree block columns (CTU columns) within a tile. That is, one tile may be set as one slice, and one tile may be composed of

a plurality of slices. When one tile is divided into a plurality of slices, the division may be limited to be performed only in the horizontal direction. In this case, the vertical boundary of the slice may coincide with the vertical boundary of the tile, but the horizontal boundary of the slice is not same with the horizontal boundary of the tile, but rather may coincide with the horizontal boundary of the coding tree block in the tile. On the other hand, the division may be limited to be performed only in the vertical direction.

[0116] The encoding/decoding apparatus may define a plurality of splitting modes for a slice. For example, the segmentation mode may include at least one of a raster-scan mode and a rectangular slice mode. In the case of the raster-scan mode, one slice may include a series of tiles (or blocks, bricks) according to the raster scan order. In the case of the rectangular slice mode, one slice may include a plurality of tiles forming a rectangular area, or may include one or more rows (or columns) of coding tree blocks in one tile forming a rectangular area.

[0117] The information on the split mode of the slice may be explicitly encoded by an encoding device and signaled to a decoding device, or may be implicitly determined by an encoding/decoding device. For example, a flag indicating whether it is a rectangular slice mode may be signaled. When the flag is a first value, a raster-scan mode may be used, and when the flag is a second value,

a rectangular slice mode may be used. The flag may be signaled at at least one level of a video parameter set (VPS), a sequence parameter set (SPS), a picture parameter set (PPS), or a picture header (PH).

[0118] As described above, one slice may be configured in a rectangular unit including one or more blocks, bricks, or tiles, and the location and size information of the slice may be expressed in the corresponding unit.

[0119] The sub picture may include one or more slices. Here, the slice may cover a rectangular area within one picture. That is, the boundary of the sub picture may always coincide with the slice boundary, and the vertical boundary of the sub picture may always coincide with the vertical boundary of the tile. All coding tree blocks (CTUs) belonging to one sub picture may belong to the same tile. All coding tree blocks belonging to one tile may belong to the same sub picture.

[0120] In the present invention, a picture may be composed of one or more sub pictures, a sub picture may be composed of one or more slices, tiles, or bricks, a slice may be composed of one or more tiles or bricks, and a tile may be composed of one or more bricks. It is described on the assumption that it can be configured, but is not limited thereto. That is, as described above, one tile may be composed of one or more slices.

[0121] The division unit may be composed of an integer number of

blocks, but is not limited thereto and may be composed of a decimal number instead of an integer number. That is, when it is not composed of an integer number of blocks, at least one division unit may be composed of sub-blocks. FIG. 3 illustrates an example of slice division according to a raster-scan mode. Referring to FIG. 3, it may be seen that it is composed of 18 x 12 blocks (each column and row), 12 tiles, and 3 slices. Here, the slice may be regarded as an example of a group of blocks or tiles according to a predetermined scan (raster scan).

[0122] FIG. 4 illustrates an example of slice division according to a rectangular slice mode. Referring to FIG. 4, it may be seen that it is composed of 18 x 12 blocks, 24 tiles, and 9 slices. Here, 24 tiles may be represented by 6 tile columns and 4 tile rows.

[0123] FIG. 5 illustrates an example in which one picture is divided into a plurality of tiles and rectangular slices. Referring to FIG. 5, one picture may be composed of 11 bricks, 4 tiles (2 tile columns and 2 tile rows), and 4 slices.

[0124] FIG. 6 illustrates an example of dividing one picture into a plurality of sub pictures. Referring to FIG. 6, one picture may consist of 18 tiles. Here, 4x4 blocks on the left (i.e., 16 CTUs) may constitute one tile, and 2x4 blocks on the right (i.e., 8 CTUs) may constitute one tile. Also, like a tile on the left, one tile may be one sub picture (or slice), and like a tile on the right,

one tile may be composed of two sub pictures (or slices).

[0125] Table 1 shows examples of division or configuration information on a sub picture according to the present disclosure, and encoding control information (e.g., information on whether to apply an in-loop filter to a boundary, etc.).

[0126] 【 Table 1】

subpics_present_flag
if(subpics_present_flag) {
max_subpics_minus1
subpic_grid_col_width_minus1
subpic_grid_row_height_minus1
for(i = 0; i < NumSubPicGridRows; i++)
for(j = 0; j < NumSubPicGridCols; j++)
subpic_grid_idx[i][j]
for(i = 0; i <= NumSubPics; i++) {
subpic_treated_as_pic_flag[i]
loop_filter_across_subpic_enabled_flag[i]
}
}

[0127] In Table 1, subpics_present_flag means whether a sub picture is supported, and when the sub picture is supported (when 1), information on the number of sub pictures (max_subpic_minus1) or information width or height (subpic_grid_col_width_minus1, subpic_grid_row_height_minus1) of the sub picture may be generated. At this time, length information such as width and height may be expressed as it is (e.g., in units of 1 pixel) or may be expressed as multiple, exponent, etc. of a predetermined unit/constant (e.g., integers such as 2, 4, 8, 16, 32, 64, 128, etc., or maximum encoding

unit, minimum coding unit, maximum transform unit, minimum transform unit, etc.).

[0128] Here, based on the width and height of the picture, and the width and height (equal length) of each sub picture, how many sub pictures exist in the picture in units of columns and rows (NumSubPicGridRows, NumSubPicGridCols) may be derived. In addition, how many sub pictures are in the picture (NumSubPics) may be derived.

[0129] The above example assumes that the width or height of each sub picture is uniform, but it may also be possible when at least one of the width or height of each sub picture is not uniform. Therefore, a flag specifying whether all sub pictures constituting one picture have the same size may be used.

[0130] When all sub pictures do not have the same size according to the flag, position information of each sub picture and size information of each sub picture may be encoded/decoded. On the other hand, when all sub pictures have the same size according to the flag, size information may be encoded/decoded only for the first sub picture.

[0131] Information on how many sub pictures exist in a column or row unit in a picture may be generated, and information on the width or height of each sub picture may be individually generated.

[0132] After sub pictures are partitioned in units of columns and rows, an index of each sub picture may be allocated. In the

implicit case, indexes may be allocated based on a predetermined scan order (raster scan, etc.) (e.g., 0, 1, 2, etc. indexes are allocated to the left->right of the first sub picture row), but explicitly index information each sub picture (subpic_grid_idx) may be generated.

[0133] In the case of a sub picture, it may be determined whether to set as a picture (subpic_treated_as_pic_flag) during a decoding process excluding an in-loop filtering operation. This may operate in relation to whether a sub picture can be considered as one independent picture (when the flag is 1) in processing such as a reference picture list for inter prediction.

[0134] It is possible to determine whether to set all sub pictures within a picture as pictures (one flag is applied in common), or it is possible to determine whether to individually set them as pictures (a plurality of flags are applied individually). Here, a plurality of individual flags may be encoded/decoded for each sub picture only when a restriction that all sub pictures are treated as pictures according to one flag applied in common is not imposed.

[0135] In addition, loop_fiter_across_subpic_enabled_flag[i] may determine whether to perform the filtering operation across the boundary of the i-th sub picture. If it is 1, it is performed across, if it is 0, it is not performed.

[0136] Here, the part related to loop_fitler_across_subpic_enabled_flag may be applied in the

same/similar manner not only in the case of the syntax element but also in the example of processing the boundary of various division units described later.

[0137]

[0138] The area located at the boundary of the picture cannot be referred or filtered because there is no data outside the picture. That is, the area located inside a picture may be referred to or may be filtered.

[0139] On the other hand, even inside a picture, it can be divided into in units of such as sub pictures, slices, tiles, bricks, etc., and in the case of some division units, whether to refer to regions adjacent to both sides of the boundary of different division units, whether to apply filtering, and the like may be adaptively determined.

[0140] Here, whether to cross-reference both regions adjacent to the boundary may be determined by explicit information or may be implicitly determined.

[0141] Here, whether to perform boundary filtering (e.g., an in-loop filter or an inside loop filter, De-blocking filter, SAO, ALF, etc.) may be determined by explicit information or implicitly determined.

[0142] For example, in the case of some units A, cross-reference for encoding/decoding between division units may be possible, and filtering may be performed on a boundary of the division unit. For

example, in the case of some units B, cross-referencing for encoding/decoding between division units may be prohibited, and filtering cannot be performed on the boundary of division units. For example, in the case of some units C, cross-reference for encoding/decoding between division units may be prohibited, and filtering may be performed on a boundary of the division unit. For example, in the case of some units D, cross-reference for encoding/decoding between division units may be possible, and whether to perform filtering at a boundary of the division unit may be determined based on predetermined flag information. For example, in the case of some units (E), whether to cross-reference for encoding/decoding between division units may be determined based on predetermined flag information, and whether to perform filtering on the boundary of division units may be determined based on predetermined flag information.

[0143] The A to E division units may correspond to at least one of the above-described sub picture, slice, tile, or brick. For example, all of the division units A to E may be sub pictures, tiles, or slices. Alternatively, some of the A to E division units may be sub pictures, and the rest may be slices or tiles. Alternatively, some of the A to E division units may be slices and the rest may be tiles.

[0144]

[0145] FIG. 7 illustrates a method of performing filtering based

on a predetermined flag according to the present disclosure.

[0146] Referring to FIG. 7, one picture may be divided into a plurality of division units (S700).

[0147] The division unit may be at least one of the aforementioned sub picture, slice, tile, and brick. For example, one picture may be divided into a plurality of sub pictures. Of course, one picture may be additionally divided into a plurality of slices and/or tiles in addition to the sub picture. Since the division unit is the same as described above, a detailed description will be omitted here.

[0148] Referring to FIG. 7, it may be determined whether to perform filtering on a boundary of a division unit based on a predetermined flag (S710).

[0149] For convenience of explanation, it is assumed that the division unit in the present disclosure is a sub picture. However, the present disclosure is not limited thereto, and the present disclosure may be applied same/similarly to the boundary of a slice, tile, or brick. In addition, filtering in the present disclosure may mean in-loop filtering applied to a reconstructed picture, and a filter for the in-loop filtering may include at least one of a deblocking filter (DF), a sample adaptive offset (SAO), or an adaptive loop filter (ALF).

[0150] The division unit may be supported for purposes such as parallel processing and partial decoding.

[0151] Therefore, when the encoding/decoding of each division unit is finished, whether to perform filtering at a boundary between division units or filtering at a boundary of division unit (e.g., in-loop filter) may be determined implicitly or explicitly.

[0152] Specifically, a flag indicating whether to perform filtering on the boundary of the division unit may be used. Here, the flag may be implicitly determined in a higher unit including a plurality of division units, or may be explicitly encoded/decoded. The higher unit may mean a picture or may mean a unit composed of only some of the division units constituting the picture. Alternatively, the flag may be implicitly determined for each division unit, or may be explicitly encoded/decoded. Alternatively, the flag may be implicitly determined for each boundary of the division unit, or may be explicitly encoded/decoded. This will be described in detail with reference to FIGS. 8 to 15.

[0153] Referring to FIG. 7, filtering may be performed on a boundary of a division unit in response to the determination in S710 (S720).

[0154] Specifically, at least one of a deblocking filter, a sample adaptive offset, or an adaptive loop filter may be applied to the boundary of the division unit. The above-described filters may be sequentially applied according to a predetermined priority. For example, after the deblocking filter is applied, a sample adaptive offset may be applied. After the sample adaptive offset is applied,

the adaptive loop filter may be applied. A method of applying the deblocking filter to the boundary of the division unit will be described in detail with reference to FIG. 16.

[0155]

[0156] FIG. 8 to 15 illustrate a method of determining whether filtering is performed on a boundary of a division unit based on one or more flags according to the present disclosure.

[0157] A flag for determining whether to perform filtering at the boundary of the division unit may be supported for each type of division unit. For example, at least one of a flag indicating whether filtering is performed on the boundary of a sub picture (loop_filter_across_subpic_enabled_flag, hereinafter referred to as a first flag), a flag indicating whether filtering is performed on the boundary of a slice (loop_filter_across_slices_enabled_flag, hereinafter referred to as a second flag), a flag indicating whether filtering is performed on the boundary of the tile (loop_filter_across_tiles_enabled_flag, hereinafter referred to as a third flag), or a flag indicating whether filtering is performed on the boundary of the brick (loop_filter_across_bricks_enabled_flag, hereinafter referred to as a fourth flag) may be supported.

[0158] Alternatively, the encoding/decoding apparatus may support only some of the aforementioned flags. For example, {the first flag, the second flag, the third flag}, {the first flag, the second

flag, the fourth flag}, {the second flag, the third flag, the fourth flag}, {the first flag, the second flag}, {the first flag, the third flag}, {the first flag, the fourth flag}, {the second flag, the third flag}, {the second flag, the fourth flag}, {the third flag, the fourth flag}, {the first flag}, {the second flag}, {the third flag}, or {the fourth flag} may be supported.

[0159] Also, all of the first to fourth flags described above may be explicitly supported, or, some of the first to fourth flags may be explicitly supported, and the others may be implicitly supported. For example, one of the first to fourth flags may be explicitly supported, and the other may be implicitly determined based on the explicitly supported flag.

[0160] In an embodiment to be described later, for convenience of description, the first to fourth flags will be referred to as `loop_filter_across_enabled_flag`. In addition, it is assumed that the flag is supported when the corresponding division unit is supported.

[0161] FIG. 8 illustrates an example in which a flag (`loop_filter_across_enabled_flag`) is supported for one picture including a plurality of division units.

[0162] Referring to Table 2, when the flag (`loop_filter_across_enabled_flag`) is a first value, filtering is restricted not to be performed at the boundary of a division unit in a picture, and when the flag is a second value, the restriction

is not imposed on the boundary of the division unit. That is, when the flag is the second value, filtering on the boundary of the division unit within the picture may be performed or may not be performed.

[0163] In other words, when the flag is the first value, it may mean that the boundary of the division unit is treated the same as the boundary of the picture, and when the flag is the second value, it may mean that the limitation that the boundary of the division unit is treated the same as the boundary of the picture is not imposed.

[0164] 【 Table 2】

loop_filter_across_enabled_flag

[0165] In FIG. 8, A to F denote a division unit, the presence of an arrow as shown in the left drawing may mean that filtering between the boundaries of the division unit can be performed, and as shown in the right drawing, the absence of an arrow may mean that filtering between the boundaries of is restricted not to be performed. For convenience of explanation, it is assumed that each division unit has a rectangular shape.

[0166] The above embodiment refers to a case where it is determined whether to collectively perform filtering on a division unit in a picture, regardless of a vertical relationship between division units.

[0167] FIG. 9 illustrates an example in which the flag is individually supported in a higher unit of a division unit. That is, based on the flag of the higher unit, it may be determined whether to perform filtering on the boundary of the division unit existing in the higher unit.

[0168] Referring to Table 3, a flag (loop_filter_across_enabled_flag) may be encoded/decoded for each higher unit. loop_filter_across_enabled_flag[i] may indicate whether filtering is performed on the boundary of the division unit within the i-th higher unit. For example, when the flag is a first value, filtering is restricted not to be performed on the boundary of the division unit within the higher unit, and when the flag is a second value, the restriction is not imposed on the boundary of the division unit within the higher unit. That is, when the flag is the second value, filtering may or may not be performed on the boundary of the division unit within the higher unit.

[0169] Alternatively, when the flag is a first value, filtering may not be performed on the boundary of the division unit within the higher unit, and when the flag is the second value, filtering may be performed on the boundary of the division unit within the higher unit. This may mean that filtering may be performed on at least one boundary of the division units within the higher unit, or filtering may be performed on the boundary of all division units

belonging to the higher unit.

[0170] In this embodiment, one picture may be composed of a plurality of higher units, and each higher unit may be composed of a plurality of division units. For example, when the higher unit is a sub picture, the division unit may be a tile, a slice, or a brick. Alternatively, the higher unit may be defined as a group of sub pictures having a smaller size than a picture, and in this case, the division unit may be a sub picture, tile, slice, or brick.

[0171] 【 Table 3】

for(i=0; i<Num_Units; i++)
loop_filter_across_enabled_flag[i]

[0172] The above embodiment refers to a case in which a flag for determining whether to perform filtering on a boundary of a division unit is supported in a higher unit defined as a group of a predetermined division unit.

[0173] Referring to FIG. 9, one picture may be composed of two higher units (i.e., a first higher unit composed of A to C and a second higher unit composed of D to F).

[0174] In the first higher unit, it is the case in which the flag indicating whether to perform filtering is 0, and in the second higher unit, it is the case in which the flag indicating whether to perform filtering is 1. The boundary between the first higher

unit and the division unit belonging to the second higher unit may or may not be filtered by a flag that determines whether to perform filtering of the higher unit.

[0175] FIG. 10 illustrates a case in which a flag (loop_filter_across_enabled_flag) is supported for each division unit constituting one picture.

[0176] This embodiment, unlike the embodiment of FIG. 9, is an example of determining whether filtering is performed on the boundary of each division unit. Thus, even if the syntax elements are the same as in Table 2, their meanings may be different.

[0177] Referring to Table 4, a flag (loop_filter_across_enabled_flag) may be encoded/decoded for each division unit. loop_filter_across_enabled_flag[i] may indicate whether filtering is performed on the boundary of the i-th division unit in the picture.

[0178] For example, when the flag is a first value, filtering is restricted so that no filtering is performed on the boundary of the i-th division unit in the picture, and when the flag is a second value, filtering may be performed on the boundary of the i-th division unit in the picture. That is, when the flag is the second value, filtering may or may not be performed on the boundary of the division unit within the picture.

[0179] Meanwhile, a flag for each division unit (loop_filter_across_enabled_flag[i]) may be selectively

encoded/decoded based on a flag (loop_filter_across_enabled_flag) for one picture. The flag for one picture is the same as described in the embodiment of FIG. 8, and a detailed description will be omitted.

[0180] For example, when filtering is restricted so that no filtering is performed on a boundary of a division unit within a picture according to a flag for one picture, the flag for each division unit is not encoded/decoded. According to the flag for the one picture, the flag for each of the division units may be encoded/decoded only when the restriction is not imposed on the boundary of the division unit.

[0181] Meanwhile, the flag for one picture and the flag for each of the division units may be encoded/decoded at the same level. Here, the same level may be any one of a video parameter set, a sequence parameter set, or a picture parameter set.

[0182] 【 Table 4】

for(i = 0; i < Num_Units; i ++)
loop_filter_across_enabled_flag[i]

[0183] Referring to FIG. 10, based on a flag for each division unit, filtering may be performed on a boundary of a corresponding division unit as shown in the left drawing, and filtering may not be performed on the boundary of a corresponding division unit as shown in the right drawing.

[0184] FIG. 11 illustrates a method of determining whether to perform filtering according to a boundary position of a division unit.

[0185] This embodiment relates to the embodiment of FIG. 9 or the embodiment of FIG. 10 described above. When it is determined that filtering is performed on the boundary of each division unit, predetermined direction information to which filtering is applied may be encoded/decoded.

[0186] Referring to Table 5, information on whether filtering is performed on boundary at least one of left, right, top, or bottom directions may be encoded/decoded. When a boundary of a specific direction among the boundaries of a division unit coincides with a boundary of a picture, encoding/decoding of information about the corresponding direction may be omitted.

[0187] In this embodiment, a flag for determining whether to perform filtering on a boundary in a specific direction is used, and a flag for determining whether to perform filtering in unit of a bundle in some directions (e.g., left+right, top+bottom, left+right+top, etc.) may be used.

[0188] 【 Table 5】

for(i = 0; i < Num_Units; i ++)
{
loop_filter_across_enabled_flag [i]
if(loop_filter_across_enabled_flag [i])
{
loop_filter_left_boundary_flag [i]
loop_filter_right_boundary_flag [i]
loop_filter_top_boundary_flag [i]
loop_filter_bottom_boundary_flag [i]
}
}

[0189] Referring to FIG. 11, the left drawing shows a case in which filtering is performed on the boundary in the omnidirectional direction of the division unit (X). The central figure shows a case in which filtering is performed only on the left and right boundary of the division unit (x), and the right figure shows a case where filtering is not performed on the boundary of the division unit (X).

[0190] FIG. 12 and 13 illustrate a method of determining whether to perform filtering on a boundary of a current division unit based on a flag for a neighboring division unit.

[0191] This embodiment may be related to the embodiment of FIG. 10 described above. That is, whether to perform filtering on the boundary of the current division unit may be determined by further considering a flag for the neighboring division unit in addition to the flag for the current division unit. When the boundary of the current division unit is a vertical boundary, the neighboring

division unit may mean a division unit adjacent to the left or right of the current division unit. When the boundary of the current division unit is a horizontal boundary, the neighboring division unit may mean a division unit adjacent to the top or bottom of the current division unit.

[0192] Referring to FIG. 12, it is assumed that filtering is performed on the boundary in the division units of X and Y. Since it is determined that filtering is performed on the boundary where X and Y contact each other, filtering may be performed on the right boundary of the division unit X (i.e., the left boundary of the division unit Y).

[0193] Referring to FIG. 13, this is a case where it is determined that filtering is not performed for only one of the division units X and Y. That is, since the flag (loop_filter_across_enabled_flag) for the division unit X is 1, filtering may be performed on the boundary of the division unit X. On the other hand, since the flag (loop_filter_across_enabled_flag) for the division unit Y is 0, filtering is not performed on the boundary of the division unit Y.

[0194] One of the reasons for filtering between division units is to reduce deterioration between division units caused by individual encoding/decoding between division units.

[0195] Thus, as in the above embodiment, filtering may be performed on the boundary of one of the adjacent division regions and filtering may not be performed on the boundary of the other

division region.

[0196] Alternatively, if applying filtering only to the boundary of one of the division regions may not be effective in removing image quality deterioration, it may be determined not to perform filtering on the boundary.

[0197] For example, when values of the flag for the division unit X and the flag for the division unit Y are different from each other, filtering may be performed on the boundary between the division units X and Y, or it may be determined that filtering is allowed.

[0198] For example, it is assumed that the left boundary of the current block coincides with the left boundary of the current division unit to which the current block belongs. In this case, even if the flag for the current division unit is 0, if the flag for the left division unit adjacent to the current division unit is 1, filtering may be performed on the left boundary of the current block.

[0199] Similarly, it is assumed that the top boundary of the current block coincides with the top boundary of the current division unit to which the current block belongs. In this case, even if the flag for the current division unit is 0, if the flag for the top division unit adjacent to the current division unit is 1, filtering may be performed on the top boundary of the current block.

[0200] Alternatively, when the values of the flag for the division unit X and the flag for the division unit Y are different from each other, it may be determined that filtering is not performed or filtering is not allowed at the boundary between the division units X and Y.

[0201] For example, it is assumed that the left boundary of the current block coincides with the left boundary of the current division unit to which the current block belongs. In this case, even if the flag for the current division unit is 1, if the flag for the left division unit adjacent to the current division unit is 0, filtering may not be performed on the left boundary of the current block.

[0202] Similarly, it is assumed that the top boundary of the current block coincides with the top boundary of the current division unit to which the current block belongs. In this case, even if the flag for the current division unit is 1, if the flag for the top division unit adjacent to the current division unit is 0, filtering may not be performed on the top boundary of the current block.

[0203] FIG. 14 illustrates a case in which information indicating whether to perform filtering is generated for each division unit boundary.

[0204] Referring to FIG. 14, whether to perform filtering for each division boundary line that divides or partitions division

units A to F may be performed. If filtering is performed on the C0 boundary, filtering may be performed on the A and B boundaries, and on the D and E boundary, otherwise, filtering may not be performed on the boundary.

[0205] The number or index of the C0, C1, R0, etc. may be derived by the division information or the partition information of the division unit. Alternatively, information for explicitly allocating information on the division unit boundary and an index may be generated. As shown in the syntax elements of Table 6 below, information for determining whether to perform filtering may be generated at each division unit boundary (in this example, each column or row) based on Num_units_rows and Num_unit_cols.

[0206] 【 Table 6】

for(i = 0; i < Num_Unit_Rows; i ++)
{
loop_filter_across_row [i]
}
for(i = 0; i < Num_Unit_Cols; i ++)
{
loop_filter_across_col [i]
}

[0207] FIG. 15 illustrates another example in which information indicating whether to perform filtering is generated for each division unit boundary.

[0208] Referring to FIG. 15, whether to perform filtering for each division boundary line that divides or partitions division

units A to F may be performed. The difference from the embodiment of FIG. 14 is that related information is generated for each divisional unit boundary, not for one column or row across the picture.

[0209] If filtering is performed on the L0 boundary, filtering may be performed on the A and B boundary, otherwise, filtering may not be performed on the boundary.

[0210] The number or index of the C0, C1, R0, etc. may be derived by the division information or the division information of the division unit. Alternatively, information for explicitly allocating information on the division unit boundary and an index may be generated. As in the syntax element of Table 7, information for determining whether to perform filtering at each division unit boundary may be generated based on Num_unit_boundary.

[0211] 【 Table 7】

for(i = 0; i < Num_Unit_boundary; i ++)
{
loop_filter_across [i]
}

[0212] FIG. 16 illustrates a method of applying a deblocking filter according to the present disclosure.

[0213] Referring to FIG. 16, a block boundary for deblocking filtering (hereinafter, referred to as an edge) among block boundaries of a reconstructed picture may be specified (S1600).

[0214] The reconstructed picture may be partitioned into a predetermined NxM sample grid. The NxM sample grid may mean a unit in which deblocking filtering is performed. Here, N and M may be 4, 8, 16 or more integers. Each pixel grid may be defined for each component type. For example, when the component type is a luminance component, N and M may be set to 4, and when the component type is a chrominance difference component, N and M may be set to 8. Regardless of the component type, a fixed-size NxM pixel grid may be used.

[0215] The edge is a block boundary positioned on an NxM sample grid, and may include at least one of a boundary of a transform block, a boundary of a prediction block, or a boundary of a sub-block.

[0216] Referring to FIG. 16, a decision value for the specified edge may be derived (S1610).

[0217] In this embodiment, it is assumed that the edge type is a vertical edge, and a 4x4 sample grid is applied. Based on the edge, the left block and the right block will be referred to as P blocks and Q blocks, respectively. The P block and the Q block are pre-reconstructed blocks, the Q block refers to a region in which deblocking filtering is currently performed, and the P block may refer to a block spatially adjacent to the Q block.

[0218] First, the decision value may be derived using a variable dSam for inducing the decision value. The variable dSam may be

derived for at least one of a first pixel line or a fourth pixel line of the P block and the Q block. Hereinafter, dSam for the first pixel line (row) of the P block and Q block is referred to as dSam0, and dSam for the fourth pixel line (row) is referred to as dSam3.

[0219] When at least one of the following conditions is satisfied, dSam0 may be set to 1, otherwise, dSam0 may be set to 0.

[0220] 【 Table 8】

condition	
1	$dqp < \text{first threshold}$
2	$(sp + sq) < \text{second threshold}$
3	$spq < \text{third threshold}$

[0221] In Table 8, dpq may be derived based on at least one of a first pixel value linearity d1 of the first pixel line of the P block or a second pixel value linearity d2 of the first pixel line of the Q block. Here, the first pixel value linearity d1 may be derived using i pixels p belonging to the first pixel line of the P block. The i may be 3, 4, 5, 6, 7 or more. The i pixels p may be continuous pixels adjacent to each other, or may be non-contiguous pixels separated by a predetermined interval. In this case, the pixel p may be i pixels closest to the edge among the pixels of the first pixel line. Similarly, the second pixel value linearity

d2 can be derived using j pixels q belonging to the first pixel line of the Q block. The j may be 3, 4, 5, 6, 7 or more. The j is set to the same value as the i , but is not limited thereto, and may be a value different from the i . The j pixels q may be contiguous pixels adjacent to each other, or may be non-contiguous pixels separated by a predetermined interval. In this case, the pixel q may be j pixels closest to the edge among the pixels of the first pixel line.

[0222] For example, when three pixels p and three pixels q are used, the first pixel value linearity $d1$ and the second pixel value linearity $d2$ may be derived as in Equation 1 below.

[0223] 【 Equation 1】

$$\text{[0224]} \quad d1 = \text{Abs}(p_{2,0} - 2 * p_{1,0} + p_{0,0})$$

$$\text{[0225]} \quad d2 = \text{Abs}(q_{2,0} - 2 * q_{1,0} + q_{0,0})$$

[0226] Alternatively, when six pixels p and six pixels q are used, the first pixel value linearity $d1$ and the second pixel value linearity $d2$ may be derived as in Equation 2 below.

[0227] 【 Equation 2】

$$\text{[0228]} \quad d1 = (\text{Abs}(p_{2,0} - 2 * p_{1,0} + p_{0,0}) + \text{Abs}(p_{5,0} - 2 * p_{4,0} + p_{3,0}) + 1) \gg 1$$

$$\text{[0229]} \quad d2 = (\text{Abs}(q_{2,0} - 2 * q_{1,0} + q_{0,0}) + \text{Abs}(q_{5,0} - 2 * q_{4,0} + q_{3,0}) + 1) \gg 1$$

[0230] In Table 8, sp may denote a first pixel value gradient $v1$ of a first pixel line of the block P , and sq may denote a second pixel value gradient $v2$ of a first pixel line of the Q block. Here,

the first pixel value gradient v_1 may be derived using m pixels p belonging to the first pixel line of the P block. The m may be 2, 3, 4, 5, 6, 7 or more. The m pixels p may be contiguous pixels adjacent to each other, or may be non-contiguous pixels separated by a predetermined interval. Alternatively, some of the m pixels p may be contiguous pixels adjacent to each other, and the others may be non-contiguous pixels separated by a predetermined interval. Similarly, the second pixel value gradient v_2 may be derived using n pixels q belonging to the first pixel line of the Q block. The n may be 2, 3, 4, 5, 6, 7 or more. The n is set to the same value as m , but is not limited thereto, and may be a value different from m . The n pixels q may be contiguous pixels adjacent to each other, or may be non-contiguous pixels separated by a predetermined interval. Alternatively, some of the n pixels q may be contiguous pixels adjacent to each other, and the others may be non-contiguous pixels separated by a predetermined interval.

[0231] For example, when two pixels p and two pixels q are used, a first pixel value gradient v_1 and a second pixel value gradient v_2 may be derived as in Equation 3 below.

[0232] 【 Equation 3】

[0233] $v_1 = \text{Abs}(p_{3,0} - p_{0,0})$

[0234] $v_2 = \text{Abs}(q_{0,0} - q_{3,0})$

[0235] Alternatively, when six pixels p and six pixels q are used, the first pixel value gradient v_1 and the second pixel value

gradient v_2 may be derived as in Equation 4 below.

[0236] **[Equation 4]**

[0237] $v_1 = \text{Abs}(p_{3,0} - p_{0,0}) + \text{Abs}(p_{7,0} - p_{6,0} - p_{5,0} + p_{4,0})$

[0238] $v_2 = \text{Abs}(q_{0,0} - q_{3,0}) + \text{Abs}(q_{4,0} - q_{5,0} - q_{6,0} + q_{7,0})$

[0239] The spq in Table 8 may be derived from the difference between the pixel $p_{0,0}$ and the pixel $q_{0,0}$ adjacent to the edge.

[0240] The first and second thresholds of Table 8 may be derived based on a predetermined parameter QP . Here, the QP may be determined using at least one of a first quantization parameter of the P block, a second quantization parameter of the Q block, or an offset for inducing QP . The offset may be a value encoded and signaled by an encoding device. For example, QP may be derived by adding the offset to the average value of the first and second quantization parameters. The third threshold of Table 8 may be derived based on the above-described quantization parameter (QP) and block boundary strength (BS). Here, the BS may be variably determined in consideration of a prediction mode of a P/Q block, an inter prediction mode, a presence or absence of a non-zero transform coefficient, a difference in motion vectors, etc.

[0241] For example, when at least one prediction mode of the P block and the Q block is an intra mode, the BS may be set to 2. When at least one of the P blocks or the Q blocks is encoded in the combined prediction mode, the BS may be set to 2. When at least one of the P block or Q block includes a non-zero transform

coefficient, the BS may be set to 1. When the P block is coded in an inter prediction mode different from the Q block (e.g., when the P block is coded in the current picture reference mode and the Q block is coded in the merge mode or AMVP mode), the BS may be set to 1. When both the P block and the Q block are coded in the current picture reference mode, and the difference between their block vectors is greater than or equal to a predetermined threshold difference, the BS may be set to 1. Here, the threshold difference may be a fixed value (e.g., 4, 8, 16) pre-committed to the encoding/decoding device.

[0242] Since dSam3 is derived using one or more pixels belonging to the fourth pixel line through the same method as dSam0 described above, a detailed description will be omitted.

[0243] A decision value may be derived based on the derived dSam0 and dSam3. For example, when both dSam0 and dSam3 are 1, the decision value may be set to a first value (e.g. 3), otherwise, the decision value may be set to a second value (e.g., 1 or 2).

[0244] Referring to FIG. 16, a filter type of a deblocking filter may be determined based on the derived decision value (S1620).

[0245] In the encoding/decoding apparatus, a plurality of filter types having different filter lengths may be defined. As an example of the filter type, there may be a long filter having the longest filter length, a short filter having the shortest filter length, or one or more middle filters that are longer than the short filter

and shorter than the long filter. The number of filter types defined in the encoding/decoding apparatus may be 2, 3, 4 or more.

[0246] For example, when the decision value is the first value, the long filter may be used, and when the decision value is the second value, the short filter may be used. Alternatively, when the decision value is the first value, one of the long filter or the middle filter may be selectively used, and when the decision value is the second value, the short filter may be used. Alternatively, when the decision value is the first value, the long filter is used, and when the decision value is not the first value, either the short filter or the middle filter may be selectively used. In particular, when the decision value is 2, the middle filter may be used, and when the decision value is 1, the short filter may be used.

[0247] Referring to FIG. 16, filtering may be performed on an edge of a reconstructed picture based on a deblocking filter according to the determined filter type (S1630).

[0248] The deblocking filter may be applied to a plurality of pixels located in both directions based on an edge and located in the same pixel line. Here, a plurality of pixels to which the deblocking filter is applied is referred to as a filtering region, and the length (or number of pixels) of the filtering region may be different for each filter type. The length of the filtering region may be interpreted as having the same meaning as the filter

length of the aforementioned filter type. Alternatively, the length of the filtering region may mean a sum of the number of pixels to which the deblocking filter is applied in the P block and the number of pixels to which the deblocking filter is applied in the Q block.

[0249] In this embodiment, it is assumed that three filter types, that is, the long filter, the middle filter, and the short filter, are defined in an encoding/decoding apparatus, and a deblocking filtering method for each filter type will be described. However, the present disclosure is not limited thereto, and only the long filter and the middle filter may be defined, only the long filter and the short filter may be defined, or only the middle filter and the short filter may be defined.

[0250] 1. In case of long filter-based deblocking filtering

[0251] For convenience of explanation, it is assumed that the edge type is a vertical edge, and the currently filtered pixel (hereinafter, the current pixel q) belongs to the Q block unless otherwise stated. The filtered pixel f_q may be derived through a weighted average of a first reference value and a second reference value.

[0252] Here, the first reference value may be derived using all or part of the pixels in the filtering area to which the current pixel q belongs. Here, the length (or number of pixels) of the filtering region may be an integer of 8, 10, 12, 14 or more. Some

pixels in the filtering area may belong to the P block and the other pixels may belong to the Q block. For example, when the length of the filtering region is 10, 5 pixels may belong to the P block and 5 pixels may belong to the Q block. Alternatively, 3 pixels may belong to the P block and 7 pixels may belong to the Q block. Conversely, 7 pixels may belong to the P block and 3 pixels may belong to the Q block. In other words, the long filter-based deblocking filtering may be performed symmetrically or asymmetrically on the P block and the Q block.

[0253] Regardless of the location of the current pixel q , all pixels belonging to the same filtering area may share one and the same first reference value. That is, the same first reference value may be used regardless of whether the currently filtered pixel is located in the P block or the Q block. The same first reference value may be used regardless of the position of the currently filtered pixel in the P block or the Q block.

[0254] The second reference value may be derived using at least one of a pixel farthest from the edge (hereinafter, referred to as a first pixel) among pixels of the filtering area belonging to the Q block or neighboring pixels of the filtering area. The neighboring pixel may mean at least one pixel adjacent to the right direction of the filtering area. For example, the second reference value may be derived as an average value between one first pixel and one neighboring pixel. Alternatively, the second reference

value may be derived as an average value between two or more first pixels and two or more adjacent pixels adjacent to the right side of the filtering area.

[0255] For the weighted average, predetermined weights $f1$ and $f2$ may be applied to the first reference value and the second reference value, respectively. Specifically, the encoding/decoding apparatus may define a plurality of weight sets, and may set the weight $f1$ by selectively using any one of the plurality of weight sets. The selection may be performed in consideration of the length (or number of pixels) of the filtering region belonging to the Q block. For example, the encoding/decoding apparatus may define a weight set as shown in Table 9 below. Each weight set may consist of one or more weights corresponding to each location of the pixel to be filtered. Accordingly, from among a plurality of weights belonging to the selected weight set, a weight corresponding to the position of the current pixel q may be selected and applied to a current pixel q . The number of weights constituting the weight set may be the same as the length of the filtering region belonging to the Q block. A plurality of weights constituting one weight set may be sampled at a predetermined interval within a range of an integer greater than 0 and less than 64. Here, 64 is only an example, and may be larger or smaller than 64. The predetermined interval may be 9, 13, 17, 21, 25 or more. The interval may be variably determined according to the length L of the filtering

region included in the Q block. Alternatively, a fixed spacing may be used regardless of L.

[0256] 【 Table 9】

Length of filtering area belonging to Q block (L)	Weight set
$L > 5$	{ 59, 50, 41, 32, 23, 14, 5 }
5	{ 58, 45, 32, 19, 6 }
$L < 5$	{ 53, 32, 11 }

[0257] Referring to Table 9, when the length (L) of the filtering area belonging to the Q block is greater than 5, {59, 50, 41, 32, 23, 14, 5} may be selected among three weight sets, and when L is 5, {58, 45, 32, 19, 6} may be selected, and when L is less than 5, {53, 32, 11} may be selected. However, Table 9 is only an example of a weight set, and the number of weight sets defined in the encoding/decoding apparatus may be 2, 4 or more.

[0258] Also, when L is 7 and the current pixel is a first pixel q_0 based on the edge, a weight 59 may be applied to the current pixel. When the current pixel is a second pixel q_1 based on the edge, a weight 50 may be applied to the current pixel, and when the current pixel is a seventh pixel q_6 based on the edge, a weight 5 may be applied to the current pixel.

[0259] A weight f_2 may be determined based on the pre-determined weight f_1 . For example, the weight f_2 may be determined as a value

obtained by subtracting the weight f_1 from a pre-defined constant. Here, the pre-defined constant is a fixed value pre-defined in the encoding/decoding apparatus, and may be 64. However, this is only an example, and an integer greater than or less than 64 may be used.

[0260] 2. In case of middle filter-based deblocking filtering

[0261] The filter length of the middle filter may be smaller than the filter length of the long filter. The length (or number of pixels) of the filtering region according to the middle filter may be smaller than the length of the filtering region according to the aforementioned long filter.

[0262] For example, the length of the filtering area according to the middle filter may be 6, 8 or more. Here, the length of the filtering region belonging to the P block may be the same as the length of the filtering region belonging to the Q block. However, the present invention is not limited thereto, and the length of the filtering region belonging to the P block may be longer or shorter than the length of the filtering region belonging to the Q block.

[0263] Specifically, a filtered pixel f_q may be derived using a current pixel q and at least one neighboring pixel adjacent to the current pixel q . Here, the neighboring pixel may include at least one of one or more pixels adjacent to the left of the current pixel q (hereinafter, left peripheral pixels) or one or more pixels

adjacent to the right of the current pixel q (hereinafter, right peripheral pixels).

[0264] For example, when the current pixel q is q_0 , two left neighboring pixels p_0 and p_1 and two right neighboring pixels q_1 and q_2 may be used. When the current pixel q is q_1 , two left neighboring pixels p_0 and q_0 and one right neighboring pixel q_2 may be used. When the current pixel q is q_2 , three left neighboring pixels p_0 , q_0 , and q_1 and one right neighboring pixel q_3 may be used.

[0265] 3. Short filter-based deblocking filtering

[0266] The filter length of the short filter may be smaller than that of the middle filter. The length (or number of pixels) of the filtering region according to the short filter may be smaller than the length of the filtering region according to the above-described middle filter. For example, the length of the filtering region according to the short filter may be 2, 4 or more.

[0267] Specifically, a filtered pixel f_q may be derived by adding or subtracting a predetermined first offset ($offset_1$) to a current pixel q . Here, the first offset may be determined based on a difference value between the pixels of the P block and the pixels of the Q block. For example, as shown in Equation 5 below, the first offset may be determined based on a difference value between the pixel p_0 and the pixel q_0 and a difference value between the pixel p_1 and the pixel q_1 . However, filtering for the current pixel

q may be performed only when the first offset is smaller than a predetermined threshold. Here, the threshold is derived based on the above-described quantization parameter (QP) and block boundary strength (BS), and a detailed description thereof will be omitted.

[0268] **[** Equation 5**]**

[0269] $\text{offset1} = (9 * (q0 - p0) - 3 * (q1 - p1) + 8) \gg 4$

[0270] Alternatively, the filtered pixel f_q may be derived by adding a predetermined second offset (offset2) to the current pixel q. Here, the second offset may be determined in consideration of at least one of a difference (or change amount) between the current pixel q and the neighboring pixels or the first offset. Here, the neighboring pixels may include at least one of a left pixel or a right pixel of the current pixel q. For example, the second offset may be determined as in Equation 6 below.

[0271] **[** Equation 6**]**

[0272] $\text{offset2} = (((q2 + q0 + 1) \gg 1) - q1 - \text{offset1}) \gg 1$

[0273] The above-described filtering method is not limited to being applied only to the deblocking filter, and may be applied similarly or similarly to an adaptive sample offset (SAO), an adaptive loop filter (ALF), etc., which are examples of an in-loop filter.

[0274]

[0275] Exemplary methods of the present disclosure are expressed

as a series of operations for clarity of explanation, but this is not intended to limit the order in which steps are performed, and each step may be performed simultaneously or in a different order if necessary. In order to implement the method according to the present disclosure, the exemplary steps may include additional steps, other steps may be included excluding some steps, or may include additional other steps excluding some steps.

[0276] Various embodiments of the present disclosure are not intended to list all possible combinations, but to describe representative aspects of the present disclosure, and matters described in the various embodiments may be applied independently or may be applied in combination of two or more.

[0277] In addition, various embodiments of the present disclosure may be implemented by hardware, firmware, software, or a combination thereof. For implementation by hardware, one or more ASICs (Application Specific Integrated Circuits), DSPs (Digital Signal Processors), DSPDs (Digital Signal Processing Devices), PLDs (Programmable Logic Devices), FPGAs (Field Programmable Gate Arrays), general purpose It may be implemented by a processor (general processor), a controller, a microcontroller, a microprocessor, etc.

[0278] The scope of the present disclosure includes software or machine-executable instructions (e.g., operating systems, applications, firmware, programs, etc.) that cause an operation

according to the method of various embodiments to be executed on a device or computer, and or a non-transitory computer-readable medium (non-transitory computer-readable medium) which stores such software or instructions etc., and is executable on a device or a computer.

Industrial Applicability

[0279] The present invention may be used to encode/decode a video signal.

CLAIMS

1. A method of decoding an image, comprising:
dividing one picture into a plurality of division units;
determining whether to perform filtering on a boundary of a current division unit based on a predetermined flag; and
performing filtering on the boundary of the current division unit in response to the determination,
wherein the division unit includes at least one of a sub picture, a slice, or a tile,
wherein the flag includes at least one of a first flag indicating whether filtering is performed on a boundary of a division unit within the one picture or a second flag indicating whether filtering is performed on the boundary of the current division unit in the one picture.

2. The method of claim 1, wherein when the first flag is a first value, it is restricted so that filtering is not performed on the boundary of the division unit within the one picture, and when the first flag is a second value, the restriction on the boundary of the division unit within the picture is not imposed,
wherein when the second flag is the first value, it is restricted so that filtering is not performed on the boundary of the current division unit, and when the second flag is the second value, filtering is allowed to be performed on the boundary of the

current division unit.

3. The method of claim 2, wherein the second flag is decoded only when it is not restricted so that filtering is not performed on the boundary of the division unit within the one picture according to the first flag.

4. The method of claim 1, wherein whether to perform filtering on the boundary of the current division unit is determined by further considering a third flag indicating whether filtering is performed on a boundary of a neighboring division unit adjacent to the current block unit.

5. The method of claim 4, wherein a position of the neighboring division unit is determined based on whether the boundary of the current division unit is a vertical boundary or a horizontal boundary.

6. The method of claim 1, wherein performing the filtering comprises:

specifying a block boundary for deblocking filtering;

deriving a decision value for the block boundary;

determining a filter type for the deblocking filtering based on the decision value; and

performing the filtering on the block boundary based on the filter type.

7. A method of encoding an image, comprising:
dividing one picture into a plurality of division units;
determining whether to perform filtering on a boundary of a current division unit; and
performing filtering on the boundary of the current division unit in response to the determination,
wherein the division unit includes at least one of a sub picture, a slice, or a tile,
wherein determining whether to perform filtering on the boundary of the current division unit comprises,
encoding at least one of a first flag indicating whether filtering is performed on a boundary of a division unit within the one picture or a second flag indicating whether filtering is performed on the boundary of the current division unit in the one picture.

8. The method of claim 7, wherein when it is determined that filtering is restricted not to be performed on the boundary of the division unit within the one picture, the first flag is encoded as a first value, and when it is determined that the restriction is not imposed on the boundary of the division unit within the picture,

the first flag is encoded as a second value,

wherein when it is determined that filtering is restricted not to be performed on the boundary of the current division unit, the second flag is encoded as the first value, and when it is determined that filtering is allowed to be performed on the boundary of the current division unit, the second flag is encoded as the second value.

9. The method of claim 8, wherein the second flag is encoded only when it is not restricted so that filtering is not performed on the boundary of the division unit within the one picture.

10. The method of claim 7, wherein whether to perform filtering on the boundary of the current division unit is determined by further considering a third flag indicating whether filtering is performed on a boundary of a neighboring division unit adjacent to the current block unit.

11. The method of claim 10, wherein a position of the neighboring division unit is determined based on whether the boundary of the current division unit is a vertical boundary or a horizontal boundary.

12. The method of claim 7, wherein performing the filtering

comprises:

- specifying a block boundary for deblocking filtering;
- deriving a decision value for the block boundary;
- determining a filter type for the deblocking filtering based on the decision value; and
- performing the filtering on the block boundary based on the filter type.