

DESCRIPTION
IMAGE PROCESSING APPARATUS AND METHOD

TECHNICAL FIELD

5 [0001]

The present disclosure relates to an image processing apparatus and a method, and more particularly, to an image processing apparatus and a method capable of reducing the amount of codes required for controlling a sample adaptive offset filter.

10

BACKGROUND ART

[0002]

In recent years, image information is treated as digital information, and at that occasion, an apparatus is becoming widely prevalent that compresses and encodes an image by employing a coding method for performing compression by orthogonal transformation and motion compensation such as discrete cosine transform using redundancy unique to image information for the purpose of highly efficient transmission and accumulation of information. Examples of the coding methods include MPEG (Moving Picture Experts Group), H.264 and MPEG-4 Part 10 (Advanced Video Coding, hereinafter referred to as H.264/AVC), and the like.

15

20

25

[0003]

Currently, for the purpose of further improving the encoding efficiency as compared with the H.264/AVC, a coding method called HEVC (High Efficiency Video Coding) is being standardized by JCTVC (Joint Collaboration Team-Video Coding) which is a joint standards organization of

30

the ITU-T and the ISO/IEC (for example, see Non-Patent Document 1).

[0004]

In the current draft of HEVC, sample adaptive offset (SAO) filter is employed. In the sample adaptive offset filter, control signals between components of Y, Cb, Cr are all transmitted independently.

[0005]

In contrast, Non-Patent Document 2 suggests transmission upon replacing an ON/OFF control signal between components with a single syntax and transmitting the type of the filter and coefficients independently between components.

15 CITATION LIST

NON-PATENT DOCUMENT

[0006]

Non-Patent Document 1: Benjamin Bross, Woo-Jin Han, Jens-Rainer Ohm, Gary J. Sullivan, Thomas Wiegand, "High efficiency video coding (HEVC) text specification draft 7", JCTVC-I1003 ver3, 2012.6.1

Non-Patent Document 2: Woo-Shik Kim, Do-Kyoung Kwon, In Suk Chong, Marta Karczewicz, "LCU SAO Enable Flag Coding", JCTVC-I0193, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 9th Meeting: Geneva, CH, 27 April-7 May, 2012

SUMMARY OF THE INVENTION

PROBLEMS TO BE SOLVED BY THE INVENTION

30 [0007]

However, in the above suggestion, the type

information is transmitted independently between the components of Y, Cb, Cr, and therefore, there is a high amount of codes required for controlling the sample adaptive offset filter.

5 [0008]

The present disclosure is made in view of such circumstances, and the amount of codes required for controlling the sample adaptive offset filter can be reduced.

10

SOLUTIONS TO PROBLEMS

[0009]

An image processing apparatus according to a first aspect of the present disclosure includes a decoding unit
15 configured to perform decoding processing on a coded stream to generate an image, and a sample adaptive offset filter unit configured to apply sample adaptive offset filter to the image generated by the decoding unit in accordance with a type of a sample adaptive offset filter
20 which is common to components of the image.

[0010]

The type of the sample adaptive offset filter is common to a brightness component and a color difference component of the image.

25 [0011]

The type of the sample adaptive offset filter is common to color difference components of the image.

[0012]

The type of the offset filter is common to a first
30 color difference component and a second color difference component.

[0013]

When the type of the sample adaptive offset filter corresponding to the first color difference component is a band offset, the type of the sample adaptive offset
5 filter corresponding to the second color difference component is a band offset.

[0014]

When the type of the sample adaptive offset filter corresponding to the first color difference component is
10 an edge offset, the type of the sample adaptive offset filter corresponding to the second color difference component is an edge offset.

[0015]

When the type of the sample adaptive offset filter
15 is an edge offset, a pattern rule of the edge offset is common between color difference components of the image.

[0016]

When the type of the offset filter corresponding to the first color difference component is one-dimensional
20 pattern of an edge offset, the type of the offset filter corresponding to the second color difference component is one-dimensional pattern of the edge offset.

[0017]

When the type of the offset filter corresponding to
25 the first color difference component is two-dimensional pattern of the edge offset, the type of the offset filter corresponding to the second color difference component is two-dimensional pattern of the edge offset.

[0018]

30 When the type of the offset filter corresponding to the first color difference component is a type for not

applying an offset, the type of the offset filter corresponding to the second color difference component is a type for not applying an offset.

[0019]

5 A color space of the image is in a Y/Cb/Cr format.

[0020]

The image processing apparatus further includes a reception unit configured to receive the coded stream and type information indicating the type of the sample

10 adaptive offset filter common between components of the image, and the decoding unit can use the type information received by the reception unit to decode the coded stream received by the reception unit.

[0021]

15 The image processing apparatus further includes a deblock filter unit configured to apply deblock filter to the image generated by the decoding unit, and the sample adaptive offset filter unit can apply sample adaptive offset filter on the image to which the deblock filter is

20 applied by the deblock filter unit.

[0022]

The reception unit can receive a merge flag indicating a same offset as an adjacent coding unit adjacent to a current coding unit, and the sample

25 adaptive offset filter unit can use the merge flag received by the reception unit to apply sample adaptive offset filter to the image generated by the decoding unit.

[0023]

An image processing method according to a first

30 aspect of the present disclosure includes performing decoding processing on a coded stream to generate an

image, and applying sample adaptive offset filter to the generated image in accordance with a type of a sample adaptive offset filter which is common to components of the image.

5 [0024]

In the first aspect of the present disclosure, the coded stream is decoded, and the image is generated. Then, the sample adaptive offset filter is applied to the generated image in accordance with the type of the sample
10 adaptive offset filter which is common between the components of the image.

[0025]

It should be noted that the image processing apparatus explained above may be an independent apparatus,
15 or may be an internal block constituting an image coding device or an image decoding device.

EFFECTS OF THE INVENTION

[0026]

20 According to a first aspect of the present disclosure, an image can be decoded. In particular, the amount of codes can be reduced.

BRIEF DESCRIPTION OF DRAWINGS

25 [0027]

Fig. 1 is a block diagram illustrating an example of main configuration of an image coding device.

Fig. 2 is a flowchart for explaining an example of flow of encoding processing.

30 Fig. 3 is a block diagram illustrating a main example of configuration of an image decoding device.

Fig. 4 is a flowchart for explaining an example of flow of decoding processing.

Fig. 5 is a figure for explaining band offset.

Fig. 6 is a figure for explaining edge offset.

5 Fig. 7 is a figure illustrating a rule list table of the edge offset.

Fig. 8 is a flowchart for explaining conventional SAO control information transmission processing.

10 Fig. 9 is a block diagram illustrating an example of configuration of a sample adaptive offset filter to which the present disclosure is applied.

Fig. 10 is a flowchart for explaining sample adaptive offset filter processing.

15 Fig. 11 is a flowchart for explaining SAO control information setting processing.

Fig. 12 is a flowchart for explaining SAO control information transmission processing.

20 Fig. 13 is a block diagram illustrating another example of configuration of a sample adaptive offset filter to which the present disclosure is applied.

Fig. 14 is a flowchart for explaining sample adaptive offset filter processing.

Fig. 15 is a flowchart for explaining SAO control information transmission processing.

25 Fig. 16 is a block diagram illustrating an example of configuration of a sample adaptive offset filter to which the present disclosure is applied.

Fig. 17 is a flowchart for explaining sample adaptive offset filter processing.

30 Fig. 18 is a flowchart for explaining SAO control information reception processing.

Fig. 19 is a block diagram illustrating another example of configuration of a sample adaptive offset filter to which the present disclosure is applied.

Fig. 20 is a flowchart for explaining sample
5 adaptive offset filter processing.

Fig. 21 is a figure illustrating an example of a multi-viewpoint image encoding method.

Fig. 22 is a figure illustrating a main example of configuration of a multi-viewpoint image coding device to
10 which the present disclosure is applied.

Fig. 23 is a figure illustrating a main example of configuration of a multi-viewpoint image decoding device to which the present disclosure is applied.

Fig. 24 is a figure illustrating an example of a
15 hierarchical image encoding method.

Fig. 25 is a figure illustrating a main example of configuration of a hierarchical image coding device to which the present disclosure is applied.

Fig. 26 is a figure illustrating an example of main
20 configuration of a hierarchical image decoding device to which the present technique is applied.

Fig. 27 is a block diagram illustrating an example of main configuration of a computer.

Fig. 28 is a block diagram illustrating an example
25 of schematic configuration of a television device.

Fig. 29 is a block diagram illustrating an example of schematic configuration of a cellular phone.

Fig. 30 is a block diagram illustrating an example of schematic configuration of a recording/reproducing
30 device.

Fig. 31 is a block diagram illustrating an example

of schematic configuration of an image-capturing device.

MODE FOR CARRYING OUT THE INVENTION

[0028]

5 Hereinafter, modes for carrying out the present disclosure (hereinafter referred to as embodiments) will be explained. It should be noted that the explanation will be made in the following order.

1. Apparatus and overview of operation
- 10 2. Explanation about conventional method
3. First embodiment (image processing apparatus (encoding side))
4. Second embodiment (image processing apparatus (decoding side))
- 15 5. Third embodiment (multi-viewpoint image encoding/multi-viewpoint image decoding device)
6. Fourth embodiment (hierarchical image encoding/hierarchical image decoding device)
7. Fifth embodiment (computer)
- 20 8. Example of application

[0029]

<1. Apparatus and overview of operation>

[Example of configuration of image coding device]

Fig. 1 illustrates a configuration of an embodiment
25 of an image coding device serving as an image processing apparatus to which the present disclosure is applied.

[0030]

An image coding device 11 as shown in Fig. 1 uses prediction processing to encode image data. In this case,
30 an example of an encoding method includes a HEVC (High Efficiency Video Coding) method.

[0031]

In this HEVC method, a coding unit (CU) is defined. The CU is also referred to as a Coding Tree Block (CTB), and is a partial area of an image in unit of picture, which is a counterpart of the macro block in an H.264/AVC method. In the latter, the size is fixed to 16 by 16 pixels, but in the former, the size is not fixed, and in each sequence, the size is designated in image compression information.

10 [0032]

For example, in Sequence Parameter Set (SPS) included in the coded data which are to be output, the maximum size of the CU (LCU (Largest Coding Unit)) and the minimum size thereof ((SCU (Smallest Coding Unit))).

15 [0033]

In each LCU, split-flag is 1 as long as the size is not less than the size of SCU, and accordingly, it is possible to divide a CU into CUs of a smaller size. When the value of split_flag is "1", a CU of which size is 2N by 2N is divided into CUs of which size is N by N, which is a hierarchy in one level below.

[0034]

Further, the CU is divided into Prediction Units (PUs), which are areas serving as target areas of intra- or inter-prediction (partial areas of an image in unit of picture), and divided into Transform Units (TUs) which are areas serving as target areas of orthogonal transformation (partial areas of an image in unit of picture). Currently, in the HEVC method, not only 4 by 4 and 8 by 8 but also 16 by 16 and 32 by 32 orthogonal transformation can be used.

[0035]

In the example of Fig. 1, the image coding device 11 includes an A/D (Analog/Digital) conversion unit 21, a screen sorting buffer 22, a calculation unit 23, an
5 orthogonal transformation unit 24, a quantization unit 25, a lossless coding unit 26, and accumulation buffer 27. The image coding device 11 also includes an inverse-quantization unit 28, an inverse-orthogonal transformation unit 29, a calculation unit 30, a deblock
10 filter 31, a frame memory 32, a selection unit 33, an intra-prediction unit 34, a motion prediction/compensation unit 35, a prediction image selection unit 36, and a rate control unit 37.

[0036]

15 Further, the image coding device 11 includes a sample adaptive offset filter 41 and an adaptive loop filter 42 provided between the deblock filter 31 and the frame memory 32.

[0037]

20 The A/D conversion unit 21 performs A/D conversion on received image data, and provides converted image data to the screen sorting buffer 22 to store the image data therein.

[0038]

25 The screen sorting buffer 22 sorts images of frames in the stored display order into the order of frames for coding in accordance with a GOP (Group of Picture) structure. The screen sorting buffer 22 provides the images of which frame order has been sorted to the
30 calculation unit 23. The screen sorting buffer 22 also provides the images of which frame order has been sorted

to the intra-prediction unit 34 and the motion prediction/compensation unit 35.

[0039]

The calculation unit 23 subtracts a prediction
5 image, which is provided from the intra-prediction unit
34 or the motion prediction/compensation unit 35 via the
prediction image selection unit 36, from an image read
from the screen sorting buffer 22, and provides
difference information thereof to the orthogonal
10 transformation unit 24.

[0040]

For example, in a case of an intra-coded image, the
calculation unit 23 subtracts a prediction image, which
is provided from the intra-prediction unit 34, from an
15 image read from the screen sorting buffer 22. For
example, in a case of an inter-coded image, the
calculation unit 23 subtracts a prediction image, which
is provided from the motion prediction/compensation unit
35, from an image read from the screen sorting buffer 22.

20 [0041]

The orthogonal transformation unit 24 applies
orthogonal transformation such as discrete cosine
transform and Karhunen-Loeve transform on difference
information provided from the calculation unit 23. The
25 orthogonal transformation unit 24 provides transformation
coefficients to the quantization unit 25.

[0042]

The quantization unit 25 quantizes the
transformation coefficients which are output from the
30 orthogonal transformation unit 24. The quantization unit
25 provides the quantized conversion coefficients to the

lossless coding unit 26.

[0043]

The lossless coding unit 26 applies lossless coding such as variable length encoding or arithmetic encoding to the quantized transformation coefficients.

[0044]

The lossless coding unit 26 obtains parameters such as information indicating the intra-prediction mode from the intra-prediction unit 34, and obtains parameters such as information indicating the inter-prediction mode and motion vector information from the motion prediction/compensation unit 35.

[0045]

The lossless coding unit 26 encodes the quantized transformation coefficients, and the obtained parameters (syntax elements), and the like, and makes them a part of header information of coded data (multiplexing). The lossless coding unit 26 provides the coded data obtained through the encoding process to the accumulation buffer 27 and stores the coded data therein.

[0046]

For example, the lossless coding unit 26 performs lossless coding processing such as variable length encoding or arithmetic encoding. An example of the variable length encoding includes CAVLC (Context-Adaptive Variable Length Coding). An example of the arithmetic encoding includes CABAC (Context-Adaptive Binary Arithmetic Coding).

[0047]

The accumulation buffer 27 temporarily holds coded stream (data) provided by the lossless coding unit 26.

With predetermined timing, the accumulation buffer 27 outputs the coded data held therein as an encoded image to, for example, a recording device, and a transmission path, not shown, provided in a later stage. More specifically, the accumulation buffer 27 is also a transmission unit which transmits a coded stream.

[0048]

The transformation coefficients quantized by the quantization unit 25 is also provided to the inverse-quantization unit 28. The inverse-quantization unit 28 dequantizes the quantized transformation coefficients according to a method corresponding to the quantization by the quantization unit 25. The inverse-quantization unit 28 provides the obtained transformation coefficients to the inverse-orthogonal transformation unit 29.

[0049]

The inverse-orthogonal transformation unit 29 performs inverse-orthogonal transformation on the provided transformation coefficients according to a method corresponding to the orthogonal transformation processing by the orthogonal transformation unit 24. The output obtained from the inverse-orthogonal transformation (restored difference information) is provided to the calculation unit 30.

[0050]

The calculation unit 30 adds a prediction image, which is provided from the intra-prediction unit 34 or the motion prediction/compensation unit 35 via the prediction image selection unit 36, to restored difference information which is an inverse-orthogonal transformation result provided from the inverse-

orthogonal transformation unit 29, thus obtaining a locally decoded image (decoded image).

[0051]

For example, when the difference information
5 corresponds to an intra-encoded image, the calculation unit 30 adds the prediction image provided by the intra-prediction unit 34 to the difference information. For example, when the difference information corresponds to an inter-encoded image, the calculation unit 30 adds the
10 prediction image provided by the motion prediction/compensation unit 35 to the difference information.

[0052]

The decoded image which is the result of the
15 addition is provided to the deblock filter 31 and the frame memory 32.

[0053]

The deblock filter 31 performs, as necessary, deblock filter processing on the decoded image thus
20 removing block distortion in the decoded image. The deblock filter 31 provides the filter processing result to the sample adaptive offset filter 41.

[0054]

The sample adaptive offset filter 41 performs
25 sample adaptive offset (SAO) processing for mainly removing ringing from an image filtered by the deblock filter 31. The sample adaptive offset filter 41 provides the filtered image to the adaptive loop filter 42.

[0055]

30 There are totally nine types of offset filters, which include two types of band offsets, six types of

edge offsets, and one without any offset. The sample adaptive offset filter 41 sets information indicating the ON/OFF state of each component for each slice, a type of an offset filter for each LCU which is the maximum coding unit, and an offset (value). Among them, the type of the filter is set commonly to the components of Y, Cb, Cr. The sample adaptive offset filter 41 uses the type and the offset which have been set to apply filter processing to the image filtered by the deblock filter 31. In the explanation below, a color space will be used as an example, in which the brightness signal is denoted as Y, and the color difference signals are denoted as Cb, Cr. More specifically, they are simply denoted as Y, Cb, Cr, but Y represents a brightness component, Cb represents a Cb color difference component, and Cr represents a Cr color difference component.

[0056]

The sample adaptive offset filter 41 provides information indicating the type, which has been set, to the lossless coding unit 26 so that it is encoded. In the sample adaptive offset filter 41, the offset explained above is a coefficient of a filter. As necessary, the offset is also referred to as a coefficient. It should be noted that the details of the sample adaptive offset filter 41 will be explained in Fig. 9 later.

[0057]

For example, the adaptive loop filter 42 performs adaptive loop filter (ALF) processing on each LCU which is the maximum coding unit. The adaptive loop filter 42 uses, for example, two-dimensional Wiener Filter as a

filter. It is to be understood that a filter other than the Wiener Filter may be used.

[0058]

5 The adaptive loop filter 42 uses the filter coefficient to perform filter processing, for each LCU, on the image filtered by the sample adaptive offset filter 41, and provides the filter processing result to the frame memory 32.

[0059]

10 Although detailed description is omitted in this specification, the image coding device 11 uses the filter coefficient calculated by the adaptive loop filter 42 for each LCU so that the residual from the original image given by the screen sorting buffer 12 becomes the
15 smallest. The calculated filter coefficient is encoded by the lossless coding unit 26, and is transmitted to the image decoding device 51 of Fig. 3 explained later. In this specification, for example, processing is performed for each LUC, but the processing unit of the adaptive
20 loop filter 42 is not limited thereto.

[0060]

 With predetermined timing, the frame memory 32 outputs the accumulated reference image via the selection unit 33 to the intra-prediction unit 34 or the motion
25 prediction/compensation unit 35.

[0061]

 For example, in a case of an intra-coded image, the frame memory 32 provides the reference image via the selection unit 33 to the intra-prediction unit 34. For
30 example, in a case of an inter-coded image, the frame memory 32 provides the reference image via the selection

unit 33 to the motion prediction/compensation unit 35.

[0062]

When the reference image provided from the frame memory 32 is an intra-coded image, the selection unit 33 provides the reference image to the intra-prediction unit 34. When the reference image provided from the frame memory 32 is an inter-coded image, the selection unit 33 provides the reference image to the motion prediction/compensation unit 35.

10 [0063]

The intra-prediction unit 34 uses pixel values within the screen to perform intra-prediction (prediction within a screen) to generate a prediction image. The intra-prediction unit 34 performs intra-prediction in multiple modes (intra-prediction modes).

[0064]

The intra-prediction unit 34 generates prediction images with all the intra-prediction modes, and evaluates each prediction image, thus selecting the optimum mode. When the optimum intra-prediction mode is selected, the intra-prediction unit 34 provides the prediction image generated with the optimum mode via the prediction image selection unit 36 to the calculation unit 23 and the calculation unit 30.

25 [0065]

As described above, intra-prediction unit 34 provides a parameter such as intra-prediction mode information and the like indicating the employed intra-prediction mode to the lossless coding unit 26 as necessary.

[0066]

The motion prediction/compensation unit 35 uses an input image provided from the screen sorting buffer 22 and a reference image provided from the frame memory 32 via the selection unit 33 to perform the motion prediction on the image which is to be inter encoded. The motion prediction/compensation unit 35 performs the motion compensation processing in accordance with the motion vectors detected by the motion prediction, thus generating a prediction image (inter-prediction image information).

[0067]

The motion prediction/compensation unit 35 performs the inter-prediction with all the candidate inter-prediction modes to generate the prediction images. The motion prediction/compensation unit 35 provides the generated prediction image via the prediction image selection unit 36 to the calculation unit 23 and the calculation unit 30.

[0068]

The motion prediction/compensation unit 35 provides the inter-prediction mode information indicating the employed inter-prediction mode and a parameter such as motion vector information indicating the calculated motion vectors to the lossless coding unit 26.

[0069]

In a case of an intra-coded image, the prediction image selection unit 36 provides the output of the intra-prediction unit 34 to the calculation unit 23 and the calculation unit 30, and in a case of an inter-coded image, the prediction image selection unit 36 provides the output of the motion prediction/compensation unit 35

to the calculation unit 23 and the calculation unit 30.
[0070]

The rate control unit 37 controls the rate of the quantization operation of the quantization unit 25 so as
5 not to cause overflow or underflow, on the basis of a compressed image accumulated in the accumulation buffer 27.

[0071]

[Operation of image coding device]

10 A flow of encoding processing executed by the image coding device 11 as explained above will be explained with reference to Fig. 2.

[0072]

In step S11, the A/D conversion unit 21 performs
15 A/D conversion on a received image. In step S12, the screen sorting buffer 22 stores images that have been subjected to the A/D conversion, and sorts them from the order in which pictures are displayed into the order in which they are encoded.

20 [0073]

When a processing target image provided from the screen sorting buffer 22 is an image of a block which is intra processed, a decoded image which is referred to is read from the frame memory 32, and is provided via the
25 selection unit 33 to the intra-prediction unit 34.

[0074]

On the basis of these images, in step S13, the intra-prediction unit 34 performs intra-prediction on the pixels of the processing target block in all the
30 candidate intra-prediction modes. It should be noted that a pixel not filtered by the deblock filter 31 is

used as a decoded pixel which is referred to.

[0075]

With this processing, the intra-prediction is performed in all the candidate intra-prediction modes, and the cost function values are calculated for all the candidate intra-prediction modes. Then, the optimum intra-prediction mode is selected on the basis of the calculated cost function values, and the prediction image generated with the intra-prediction in the optimum intra-prediction mode and the cost function value thereof are provided to the prediction image selection unit 36.

[0076]

When the processing target image provided from the screen sorting buffer 22 is an image which is to be inter-processed, the image which is referred to is read from the frame memory 32 and is provided via the selection unit 33 to the motion prediction/compensation unit 35. On the basis of these images, in step S14, the motion prediction/compensation unit 35 performs the motion prediction/compensation processing.

[0077]

With this processing, the motion prediction processing is performed in all the candidate inter-prediction modes, and the cost function values are calculated for all the candidate inter-prediction modes, and the optimum inter-prediction mode is determined on the basis of the calculated cost function value. Then, the prediction image generated in the optimum inter-prediction mode and the cost function value thereof are provided to the prediction image selection unit 36.

[0078]

In step S15, the prediction image selection unit 36 determines, as the optimum prediction mode, one of the optimum intra-prediction mode and the optimum inter-prediction mode on the basis of the cost function values which are output from the intra-prediction unit 34 and the motion prediction/compensation unit 35. Then, the prediction image selection unit 36 selects the prediction image in the determined optimum prediction mode, and provides the prediction image to the calculation units 23, 30. The prediction image is used for calculation in steps S16, S21 explained later.

[0079]

The selection information of the prediction image is provided to the intra-prediction unit 34 or the motion prediction/compensation unit 35. When the prediction image in the optimum intra-prediction mode is selected, the intra-prediction unit 34 provides the information indicating the optimum intra-prediction mode (more specifically, a parameter of intra-prediction) to the lossless coding unit 26.

[0080]

When the prediction image in the optimum inter-prediction mode is selected, the motion prediction/compensation unit 35 outputs information indicating the optimum inter-prediction mode and information according to the optimum inter-prediction mode (more specifically, a parameter of motion prediction) to the lossless coding unit 26. The information according to the optimum inter-prediction mode includes motion vector information, reference frame information, and the like.

[0081]

In step S16, the calculation unit 23 calculates a difference between the images sorted by the processing in step S12 and the prediction image selected by the processing in step S15. When the inter-prediction is performed, the prediction image is provided from the motion prediction/compensation unit 35 via the prediction image selection unit 36 to the calculation unit 23. When the intra-prediction is performed, the prediction image is provided from the intra-prediction unit 34 via the prediction image selection unit 36 to the calculation unit 23.

[0082]

The amount of data of the difference data is reduced as compared with the original image data. Therefore, the amount of data can be compressed more greatly as compared with a case where an image is encoded as it is.

[0083]

In step S17, the orthogonal transformation unit 24 performs orthogonal transformation on difference information provided from the calculation unit 23. More specifically, orthogonal transformation such as discrete cosine transform and Karhunen-Loeve transformation and like is performed and, transformation coefficients are output.

[0084]

In step S18, the quantization unit 25 quantizes the transformation coefficients. In this quantization, the rate is controlled as explained in the processing of step S28 described later.

[0085]

As described above, the quantized difference information is locally decoded as follows. More specifically, in step S19, the inverse-quantization unit
5 28 dequantizes the transformation coefficient quantized by the quantization unit 25 according to the characteristics corresponding to the characteristics of the quantization unit 25. In step S20, the inverse-orthogonal transformation unit 29 performs inverse-
10 orthogonal transformation on the transformation coefficients dequantized by the inverse-quantization unit 28 according to the characteristics corresponding to the characteristics of the orthogonal transformation unit 24.

[0086]

15 In step S21, the calculation unit 30 adds the prediction image received via the prediction image selection unit 36 to difference information locally decoded, and generates a locally decoded (i.e., locally decoded) image (image corresponding to input to the
20 calculation unit 23).

[0087]

In step S22, the deblock filter 31 performs the deblock filter processing on the image which is output from the calculation unit 30, thus removing block
25 distortion. The filtered image provided by the deblock filter 31 is output to the sample adaptive offset filter 41.

[0088]

In step S23, the sample adaptive offset filter 41
30 performs sample adaptive offset filter processing. With this processing, the type of the offset filter which is

common to the components of Y, Cb, Cr is set for each LCU which is the maximum coding unit, and further, the offset is set for each component. Further, by using them, the filter processing is applied to the image filtered by the deblock filter 31. It should be noted that the details of the sample adaptive offset filter processing will be explained later with reference to Fig. 10.

[0089]

Each piece of information obtained when the type information common to the components and the offset for each component are provided to the lossless coding unit 26 (which will be hereinafter collectively referred to as adaptive offset parameters) are encoded in step S26 explained later.

[0090]

In step S24, the adaptive loop filter 42 applies adaptive loop filter processing to the image filtered by the sample adaptive offset filter 41. For example, on the image filtered by the sample adaptive offset filter 41, the filter processing is applied to an image for each LCU using the filter coefficients, and the filter processing result is provided to the frame memory 32.

[0091]

As described above, the processing units of the sample adaptive offset filter 41 and the adaptive loop filter 42 are caused to be the same, so that the processing can be done efficiently.

[0092]

In step S25, the frame memory 32 stores filtered images. It should be noted that the frame memory 32 also receives images, which are not filtered by the deblock

filter 31, the sample adaptive offset filter 41, and the adaptive loop filter 42, from the calculation unit 30, and stores the images.

[0093]

5 On the other hand, the transformation coefficients quantized in step S18 explained above are also provided to the lossless coding unit 26. In step S26, the lossless coding unit 26 encodes the quantized transformation coefficients which are output from the quantization unit 25 and the provided parameters. More
10 specifically, the difference image is encoded by lossless encoding such as variable length encoding, arithmetic encoding, and the like to be compressed. In this case, the parameters of the sample adaptive offset filter are
15 also encoded.

[0094]

 In step S27, the accumulation buffer 27 accumulates the encoded difference image (which is a coded stream) as a compressed image. The compressed images accumulated in
20 the accumulation buffer 27 are read as necessary, and transmitted to the decoding side via the transmission path.

[0095]

 In step S28, the rate control unit 37 controls the
25 rate of the quantization operation of the quantization unit 25 so as not to cause overflow or underflow, on the basis of the compressed image accumulated in the accumulation buffer 27.

[0096]

30 When the processing of step S28 is finished, the encoding processing is terminated.

[0097]

[Example of configuration of image decoding device]

Fig. 3 illustrates a configuration of an embodiment of an image decoding device serving as an image processing apparatus to which the present disclosure is applied. The image decoding device 51 as shown in Fig. 3 is a decoding device corresponding to the image coding device 11 of Fig. 1.

[0098]

10 The coded stream (data) encoded by the image coding device 11 is transmitted via a predetermined transmission path to the image decoding device 51 corresponding to the image coding device 11, and is decoded by the image decoding device 51.

15 [0099]

As shown in Fig. 3, the image decoding device 51 includes an accumulation buffer 61, a lossless decoding unit 62, an inverse-quantization unit 63, an inverse-orthogonal transformation unit 64, a calculation unit 65, 20 a deblock filter 66, a screen sorting buffer 67, and a D/A conversion unit 68. The image decoding device 51 includes a frame memory 69, a selection unit 70, an intra-prediction unit 71, a motion prediction/compensation unit 72, and a selection unit 73.

25 [0100]

Further, the image decoding device 51 includes a deblock filter 66 and a sample adaptive offset filter 81 and adaptive loop filter 82 between the screen sorting buffer 67 and the frame memory 69.

30 [0101]

The accumulation buffer 61 is also a reception unit

which receives coded data transmitted. The accumulation buffer 61 receives and accumulates coded data transmitted. The coded data are encoded by the image coding device 11. The lossless decoding unit 62 decodes coded data, which
5 are read from the accumulation buffer 61 with predetermined timing, in accordance with the method corresponding to the encoding method of the lossless coding unit 26 of Fig. 1.

[0102]

10 The lossless decoding unit 62 provides a parameter such as information indicating the intra-prediction mode in which decoding is performed to the intra-prediction unit 71, and provides information indicating the inter-prediction mode and a parameter such as motion vector
15 information to the motion prediction/compensation unit 72. The lossless decoding unit 62 provides the decoded adaptive offset parameters (such as information indicating ON/OFF states of the components, information indicating the type common to the components for each LCU,
20 the offset of each component, and the like) to the sample adaptive offset filter 81.

[0103]

The inverse-quantization unit 63 dequantizes the coefficient data obtained in the decoding process
25 performed by the lossless decoding unit 62 (quantized coefficients) in accordance with the method corresponding to the quantization method of the quantization unit 25 of the Fig. 1. More specifically, the inverse-quantization unit 63 uses the quantization parameters provided by the
30 image coding device 11 to dequantize the quantization coefficients in accordance with the same method as that

of the inverse-quantization unit 28 of Fig. 1.

[0104]

The inverse-quantization unit 63 provides the inverse-quantized coefficient data, i.e., the orthogonal transformation coefficients, to the inverse-orthogonal transformation unit 64. The inverse-orthogonal transformation unit 64 performs inverse-orthogonal transformation on the orthogonal transformation coefficients according to the method corresponding to the orthogonal transformation method of the orthogonal transformation unit 24 of Fig. 1, and obtains decoded residual data corresponding to residual data before the orthogonal transformation is performed by the image coding device 11.

15 [0105]

The obtained decoded residual data obtained from the inverse-orthogonal transformation is provided to the calculation unit 65. The calculation unit 65 receives a prediction image from the intra-prediction unit 71 or the motion prediction/compensation unit 72 via the selection unit 73.

[0106]

The calculation unit 65 adds the decoded residual data and the prediction image, and obtains decoded image data corresponding to image data before the prediction image is subtracted by the calculation unit 23 of the image coding device 11. The calculation unit 65 provides the decoded image data to the deblock filter 66.

[0107]

30 The deblock filter 66 performs the deblock filter processing on the decoded image as necessary to remove

block distortion from the decoded image. The deblock filter 66 provides the filter processing result to the sample adaptive offset filter 81.

[0108]

5 The sample adaptive offset filter 81 performs offset filter (SAO) processing, mainly for removing ringing, on the image filtered by the deblock filter 66.

[0109]

10 The sample adaptive offset filter 81 uses the adaptive offset parameter provided from the lossless decoding unit 62 to apply the filter processing on the image filtered by the deblock filter 66 for each LCU which is the maximum coding unit. The sample adaptive offset filter 81 provides the filtered image to the
15 adaptive loop filter 82.

[0110]

 The details of the sample adaptive offset filter 81 will be explained in Fig. 16, but the sample adaptive offset filter 81 receives the information indicating the
20 ON/OFF state of each component for each slice, and the type (type) and the offset (value) of the offset filter for each LCU which is the maximum coding unit. Among them, the type of the filter is set commonly to the components of Y, Cb, Cr. The sample adaptive offset
25 filter 81 uses the received type and offset to apply the filter processing to the image filtered by the deblock filter 66.

[0111]

 The adaptive loop filter 82 is configured basically
30 in the same manner as the adaptive loop filter 42 of the image coding device 11 of Fig. 1, and performs the

adaptive loop filter processing on each LCU which is the maximum coding unit. For the image filtered by the sample adaptive offset filter 81, the adaptive loop filter 82 uses the filter coefficients to perform the filter processing on each LCU, and provides the filter processing result to the frame memory 69 and the screen sorting buffer 67.

[0112]

Although the detailed description is omitted in this specification, the filter coefficients are calculated for each LCU by the adaptive loop filter 42 of the image coding device 11 in the image decoding device 51, and those encoded and received are decoded by the lossless decoding unit 62 to be used.

[0113]

The screen sorting buffer 67 sorts images. More specifically, the order of frames sorted for the order of encoding by the screen sorting buffer 22 of Fig. 1 is sorted into the original order for display. The D/A conversion unit 68 performs D/A conversion on an image provided from the screen sorting buffer 67, outputs the image to a display, not shown, and causes the display to show the image.

[0114]

The output of the adaptive loop filter 82 is further provided to the frame memory 69.

[0115]

The frame memory 69, the selection unit 70, the intra-prediction unit 71, the motion prediction/compensation unit 72, and the selection unit 73 respectively correspond to the frame memory 32, the

selection unit 33, the intra-prediction unit 34, the motion prediction/compensation unit 35, and the prediction image selection unit 36 of the image coding device 11.

5 [0116]

The selection unit 70 reads inter-processed image and the reference image from the frame memory 69, and provides the images to the motion prediction/compensation unit 72. The selection unit 70 reads the image used for
10 the intra-prediction from the frame memory 69, and provides the image to the intra-prediction unit 71.

[0117]

As necessary, the lossless decoding unit 62 provides the intra-prediction unit 71 with, such as
15 information indicating intra-prediction mode obtained by decoding the header information. On the basis of the information, the intra-prediction unit 71 generates a prediction image from the reference image obtained from the frame memory 69, and provides the generated
20 prediction image to the selection unit 73.

[0118]

The motion prediction/compensation unit 72 receives information obtained by decoding the header information (the prediction mode information, the motion vector
25 information, the reference frame information, the flags, various kinds of parameters, and the like) from the lossless decoding unit 62.

[0119]

On the basis of the information provided from the
30 lossless decoding unit 62, the motion prediction/compensation unit 72 generates the prediction

image from the reference image obtained from the frame memory 69, and provides the generated prediction image to the selection unit 73.

[0120]

5 The selection unit 73 selects the prediction image generated by the motion prediction/compensation unit 72 or the intra-prediction unit 71, and provides the selected prediction image to the calculation unit 65.

[0121]

10 [Operation of image decoding device]

 An example of flow of decoding processing executed by the image decoding device 51 as explained above will be explained with reference to Fig. 4.

[0122]

15 When the decoding processing is started, the accumulation buffer 61 receives and accumulates the coded stream (data), which has been transmitted, in step S51. In step S52, the lossless decoding unit 62 decodes the coded data provided from the accumulation buffer 61.

20 Ipicture, Ppicture, and Bpicture encoded by the lossless coding unit 26 of Fig. 1 are decoded.

[0123]

 Before decoding of the picture, information about parameters such as the motion vector information, the

25 reference frame information, and the prediction mode information (the intra-prediction mode or the inter-prediction mode) is also decoded.

[0124]

 When the prediction mode information is intra-

30 prediction mode information, the prediction mode information is provided to the intra-prediction unit 71.

When the prediction mode information is inter-prediction mode information, the motion vector information corresponding to the prediction mode information and the like are provided to the motion prediction/compensation unit 72. The adaptive offset parameter is also decoded, and is provided to the sample adaptive offset filter 81. [0125]

In step S53, the intra-prediction unit 71 or the motion prediction/compensation unit 72 respectively performs prediction image generation processing in association with the prediction mode information provided from the lossless decoding unit 62. [0126]

More specifically, when the lossless decoding unit 62 provides the intra-prediction mode information, the intra-prediction unit 71 generates the intra-prediction image in the intra-prediction mode. When the lossless decoding unit 62 provides the inter-prediction mode information, the motion prediction/compensation unit 72 performs the motion prediction/compensation processing in the inter-prediction mode, and generates the inter-prediction image. [0127]

With this processing, the prediction image (intra-prediction image) generated by the intra-prediction unit 71 or the prediction image (inter-prediction image) generated by the motion prediction/compensation unit 72 is provided to the selection unit 73. [0128]

In step S54, the selection unit 73 selects the prediction image. More specifically, the prediction

image generated by the intra-prediction unit 71 or the prediction image generated by the motion prediction/compensation unit 72 is provided. Therefore, the provided prediction image is selected, and provided
5 to the calculation unit 65, and is added to the output of the inverse-orthogonal transformation unit 64 in step S57 explained later.

[0129]

In step S52 explained above, the transformation
10 coefficients decoded by the lossless decoding unit 62 are also provided to the inverse-quantization unit 63. In step S55, the inverse-quantization unit 63 dequantizes the transformation coefficients decoded by the lossless decoding unit 62 in accordance with the characteristics
15 corresponding to the characteristics of the quantization unit 25 of Fig. 1.

[0130]

In step S56, the inverse-orthogonal transformation unit 29 performs the inverse-orthogonal transformation on
20 the transformation coefficients, which are dequantized by the inverse-quantization unit 28, in accordance with the characteristics corresponding to the characteristics of the orthogonal transformation unit 24 of Fig. 1. Accordingly, the difference information corresponding to
25 the input of the orthogonal transformation unit 24 of Fig. 1 (the output of the calculation unit 23) is decoded.

[0131]

In step S57, the calculation unit 65 adds the difference information to the prediction image which is
30 selected in the processing of step S54 explained above and is input via the selection unit 73. Thus, the

original image is decoded.

[0132]

In step S58, the deblock filter 66 performs deblock filter processing on the image which is output by the calculation unit 65. Therefore, the block distortion is removed. The decoded image provided by the deblock filter 66 is output to the sample adaptive offset filter 81.

[0133]

10 In step S59, the sample adaptive offset filter 81 performs the sample adaptive offset filter processing. The sample adaptive offset filter 81 uses the adaptive offset parameter provided by the lossless decoding unit 62 to apply the filter processing onto the image filtered by the deblock filter 66. The sample adaptive offset filter 81 provides the filtered image to the adaptive loop filter 82.

[0134]

20 Although the details of the sample adaptive offset filter 81 will be explained later in Fig. 17, the sample adaptive offset filter 81 performs the sample adaptive offset filter processing. With this processing, the type of the offset filter which is common to the components of Y, Cb, Cr is set for each LCU which is the maximum coding unit, and further, the offset is set for each component. Further, by using them, the filter processing is applied to the image filtered by the deblock filter 66.

[0135]

30 In step S60, the adaptive loop filter 82 performs the adaptive loop filter processing on the image filtered by the sample adaptive offset filter 81. The adaptive

loop filter 82 uses the filter coefficients calculated for each LCU which is the maximum coding unit to perform the filter processing for each LCU on the input image, and provides the filter processing result to the screen sorting buffer 67 and the frame memory 69.

[0136]

In step S61, the frame memory 69 stores the filtered image.

[0137]

10 In step S62, the screen sorting buffer 67 sorts the images filtered by the adaptive loop filter 82. More specifically, the order of frames sorted for encoding by the screen sorting buffer 22 of the image coding device 11 is sorted into the original order for display.

15 [0138]

In step S63, the D/A conversion unit 68 performs the D/A conversion on the images provided by the screen sorting buffer 67. The images are output to a display, not shown, and the images are displayed.

20 [0139]

When the processing in step S63 is finished, the decoding processing is terminated.

[0140]

<Explanation about conventional method>

25 [Adaptive offset processing according to HEVC method]

Subsequently, the sample adaptive offset filter according to the HEVC method will be explained. In the HEVC method, the Sample Adaptive Offset method is employed.

30

[0141]

At the encoding side, the sample adaptive offset filter 41 is provided between the deblock filter (DB) 31 and the adaptive loop filter (ALF) 42. Likewise, at the decoding side, the sample adaptive offset filter 81 is provided between the deblock filter (DB) 66 and the adaptive loop filter (ALF) 82.

[0142]

The types of the adaptive offsets include two types of band offsets and six types of edge offsets, and alternatively, the offset may not be applied. It is possible to select which type of adaptive offset explained above is used to encode the image for each LCU.

[0143]

This selection information is encoded as PQAO Info. by the encoding unit (Entropy Coding), and the bit stream is generated, and the generated bit stream is transmitted to the decoding side. By using this method, the encoding efficiency can be improved.

[0144]

First, the details of the band offset will be explained with reference to Fig. 5.

[0145]

In the band offset, a single mark denotes one band = eight pixels in the example of Fig. 5. The brightness pixel value is divided into 32 bands, and each band independently has an offset value.

[0146]

More specifically, in the example of Fig. 5, sixteen central bands of 0 to 255 pixels (32 bands) are classified into a first group, and eight bands at both sides are classified into a second group.

[0147]

Then, the offset of any one of the first group and the second group is encoded and sent to the decoding side. In general, any given area is often either an area where it is clear as to white or black or an area where the color is ambiguous. All pixels are rarely in both of the first group and the second group. For this reason, when the offset of only one of them is sent, this can suppress increase in the amount of encoding caused by transmission of pixel values of not-included pixels in each LCU.

[0148]

When the input signal is given by broadcast, the brightness signal is limited to the range of 16, 235, and the color difference signal is limited to the range of 16, 240. At this occasion, the broadcast legal shown in the lower column of Fig. 5 is applied, and the offset values for the two bands at each of the both sides which are indicated with × marks are not transmitted.

[0149]

Subsequently, the details of the edge offset will be explained with reference to Fig. 6.

[0150]

At the edge offset, the pixel value in question and an adjacent pixel value adjacent to the pixel value are compared, and the offset value is transmitted to a category corresponding thereto.

[0151]

The edge offsets include four one-dimensional patterns indicated in A of Fig. 6 to D of Fig. 6, and two two-dimensional patterns indicated in E of Fig. 6 and F of Fig. 6, and the offsets are transmitted in the

categories indicated in Fig. 6.

[0152]

In A of Fig. 6, adjacent pixels are arranged at the right and the left of the pixel C in one dimensional
5 direction. More specifically, this indicates 1-D, 0-degree pattern forming zero degree with respect to the pattern A of Fig. 6. In B of Fig. 6, adjacent pixels are arranged above and below the pixel C in one dimensional
10 direction. More specifically, this indicates 1-D, 90-degrees pattern forming 90 degrees with respect to the pattern A of Fig. 6.

[0153]

In C of Fig. 6, adjacent pixels are arranged upper left and lower right of the pixel C in one dimensional
15 direction. More specifically, this indicates 1-D, 135-degrees pattern forming 135 degrees with respect to the pattern A of Fig. 6. In D of Fig. 6, adjacent pixels are arranged upper right and lower left of the pixel C in one dimensional direction. More specifically, this indicates
20 1-D, 135-degrees pattern forming 45 degrees with respect to the pattern A of Fig. 6.

[0154]

In E of Fig. 6, adjacent pixels are arranged above, below, at the left, and at the right of the pixel C in
25 two dimensional direction. More specifically, this indicates 2-D, cross pattern which crosses the pixel C. In F of Fig. 6, adjacent pixels are arranged upper right, lower left, upper left, and lower right of the pixel C in two dimensional direction. More specifically, this
30 indicates 2-D, diagonal pattern which diagonally crosses the pixel C.

[0155]

Reference symbol A of Fig. 7 shows a rule list table of one-dimensional patterns (Classification rule for 1-D patterns). The patterns of A of Fig. 6 to D of Fig. 6 are classified into five types of categories as shown in A of Fig. 7, and according to the category, the offset is calculated and sent to the decoding unit.

[0156]

When the pixel value of the pixel C is less than the pixel values of the two adjacent pixels, it is classified into category 1. When the pixel value of the pixel C is less than the pixel value of one of the adjacent pixels and matches the pixel value of the other of the adjacent pixels, it is classified into category 2. When the pixel value of the pixel C is more than the pixel value of one of the adjacent pixels and matches the pixel value of the other of the adjacent pixels, it is classified into category 3. When the pixel value of the pixel C is more than the pixel values of the two adjacent pixels, it is classified into category 4. If none of the above is applicable, it is classified into category 0.

[0157]

Reference symbol B of Fig. 7 indicates a rule list table of the two-dimensional patterns (Classification rule for 2-D patterns). The patterns of E of Fig. 6 and F of Fig. 6 are classified into seven types of categories as shown in B of Fig. 7, and according to the category, the offset is sent to the decoding side.

[0158]

When the pixel value of the pixel C is less than the pixel values of the four adjacent pixels, it is

classified into category 1. When the pixel value of the pixel C is less than the pixel values of the three adjacent pixels and matches the pixel value of the fourth adjacent pixel, it is classified into category 2. When
5 the pixel value of the pixel C is less than the pixel values of the three adjacent pixels and more than the pixel value of the fourth adjacent pixel, it is classified into category 3.

[0159]

10 When the pixel value of the pixel C is more than the pixel values of the three adjacent pixels and less than the pixel value of the fourth adjacent pixel, it is classified into category 4. When the pixel value of the pixel C is more than the pixel values of the three
15 adjacent pixels and matches the pixel value of the fourth adjacent pixel, it is classified into category 5. When the pixel value of the pixel C is more than the pixel values of the four adjacent pixels, it is classified into category 6. If none of the above is applicable, it is
20 classified into category 0.

[0160]

As described above, at the edge offset, the amount of calculation for one-dimensional pattern is lower because comparison can be done with only two adjacent
25 pixels. In a high efficiency encoding condition, the value of one bit offset is made into higher precision as compared with the low delay encoding condition, and sent to the decoding side.

[0161]

30 The adaptive offset processing explained above is processing that is performed for each LCU in the HEVC

method (Non-Patent Document 1). The adaptive offset processing in the HEVC method, the control signals between the components of Y, Cb, Cr are all independently transmitted.

5 [0162]

More specifically, in the HEVC method, the SAO control information such as the ON/OFF information, the filter type information, and the filter coefficient is transmitted for each LCU or all independently for each component.

[0163]

In contrast, Non-Patent Document 2 suggests the following technique: the ON/OFF control signal between components in the SAO control information explained above is replaced with a single syntax and transmitted.

[0164]

Subsequently, conventional SAO control information transmission processing for each LCU will be explained with reference to a flowchart of Fig. 8.

20 [0165]

In step S11, the component ID(cIdx) is zero. In this case, cIdx = 0 indicates Y (brightness signal), cIdx = 1 indicates Cb (color difference signal), and cIdx = 2 indicates Cr (color difference signal).

25 [0166]

In step S12, a determination is made as to whether $cIdx < 3$ holds or not. When $cIdx < 3$ is determined to hold in step S12, the processing in step S13 is subsequently performed. In step S13, a determination is made as to whether $slice_sample_adaptive_offset_flag[cIdx] = 1$ holds or not.

The flag `slice_sample_adaptive_offset_flag[cIdx]` is the information indicating the ON/OFF state of (flag) each component for each slice.

[0167]

5 When `slice_sample_adaptive_offset_flag[cIdx] = 1` is determined to hold (more specifically, the sample adaptive offset filter is performed on `cIdx`) in step S13, the processing in step S14 is subsequently performed. In step S14, the SAO control information for `cIdx (= 0)` is
10 transmitted. In Non-Patent Document 2, the filter type information, the filter coefficient (offset), and the like are transmitted as the SAO control information.

[0168]

 In step S13,
15 `slice_sample_adaptive_offset_flag[cIdx] = 0` holds (more specifically, the sample adaptive offset filter is determined not to perform on `cIdx`, the processing in step S14 is skipped, and the processing in step S15 is subsequently performed).

20 [0169]

 In step S15, `cIdx++` is performed (more specifically, `cIdx = 1`), the processing in step S12 is performed back again. In step S12, `cIdx < 3` is determined not to hold, the processing in step S16 is subsequently performed.

25 [0170]

 In step S16, a determination is made as to whether the current LCU which is currently being processed is the last LCU or not. When the current LCU which is currently being processed is determined not to be the last LCU in
30 step S16, the processing in step S17 is subsequently performed. In step S17, a subsequent LCU is adopted as

the current LCU, and the processing in step S11 is performed back again.

[0171]

When the current LCU which is currently being
5 processed is determined to be the last LCU in step S16, the SAO control information transmission processing for each LCU is terminated.

[0172]

As described above, in Non-Patent Document 2, the
10 SAO control information such as the filter type information and the filter coefficient is transmitted for each LCU or all independently for each component.

[0173]

However, in Non-Patent Documents 1, 2, the type
15 information is transmitted independently between the components of Y, Cb, Cr, and therefore, the amount of codes required for controlling the sample adaptive offset filter is high.

[0174]

20 In view of the above issues, the present technique is made such that the filter type is made common between the components (Y, Cb, Cr or Cb, Cr), and only the coefficients are independently transmitted as the SAO control information.

25 [0175]

<3. First embodiment>

[Example of configuration of sample adaptive offset filter]

Fig. 9 is a block diagram illustrating an example
30 of configuration of a sample adaptive offset filter and a lossless coding unit in the image coding device of Fig. 1.

[0176]

In the example of Fig. 9, the sample adaptive offset filter 41 includes an SAO flag setting unit 111, a type setting unit 112, an offset setting unit 113, an SAO control information setting unit 114, an offset buffer 115, a filter processing unit 116, and an image buffer 117.

[0177]

The lossless coding unit 26 includes at least a syntax write unit 121.

[0178]

The deblocked pixel values given by the deblock filter 31 are input into the SAO flag setting unit 111, the type setting unit 112, the offset setting unit 113, and the filter processing unit 116. The SAO flag setting unit 111 uses the deblocked pixel values to set an SAO flag (slice_sample_adaptive_offset_flag) indicating the ON/OFF state of the sample adaptive offset filter of each component in units of slices. The SAO flag setting unit 111 provides the SAO flag, which has been set, to the type setting unit 112 and syntax write unit 121.

[0179]

The type setting unit 112 uses the deblocked pixel values to set, in units of LCUs, the type of the filter which is common to the components of Y, Cb, Cr. For example, in units of LCUs, the cost function value is calculated, and the optimum type for the LCU of which cost function value is the least is set. At this occasion, the signals of all the components are used, but the used signals are not limited, and, for example, only the brightness signal may be used. The type setting unit

112 provides the information indicating the type to the offset setting unit 113 and the syntax write unit 121. The type setting unit 112 also provides the SAO flag of each component to the offset setting unit 113.

5 [0180]

The offset setting unit 113 uses the deblocked pixel values to set the offsets (coefficients) independent in each component of Y, Cb, Cr in units of LCUs. At this occasion, for example, the signals of each component (deblocked pixel values) are used to calculate the cost function value in units of LCUs, and the optimum offset for the LCU of which cost function value is the least is set.

[0181]

15 The offset setting unit 113 provides the filter processing unit 116 with the SAO flag and the offset of each component and the type common to the components. The offset setting unit 113 also provides the offset to the SAO control information setting unit 114.

20 [0182]

The SAO control information setting unit 114 writes the offset given by the offset setting unit 113 to the offset buffer 115 for each component. The SAO control information setting unit 114 refers to the information written to the offset buffer 115, and determines whether the offset given by the offset setting unit 113 matches the offset of a left LCU which is adjacent to the left of the current LCU which is currently being processed. When the offset given by the offset setting unit 113 is determined to match the offset of a left LCU which is adjacent to the left of the current LCU which is

currently being processed, the SAO control information setting unit 114 sets `sao_merge_left_flag`, and provides `sao_merge_left_flag`, which has been set, to the syntax write unit 121 as the SAO control information.

5 [0183]

Further, the SAO control information setting unit 114 determines whether the offset given by the offset setting unit 113 matches the offset of an upper LCU which is adjacent to the upper side of the current LCU. When
10 the offset given by the offset setting unit 113 is determined to match the offset of an upper LCU which is adjacent to the upper side of the current LCU, the SAO control information setting unit 114 sets `sao_merge_top_flag`, and provides `sao_merge_top_flag`,
15 which has been set, to the syntax write unit 121 as the SAO control information.

[0184]

When the offset given by the offset setting unit 113 matches neither the left LCU nor the upper LCU, the
20 SAO control information setting unit 114 provides only the offset to the syntax write unit 121 as the SAO control information. More specifically, `sao_merge_left_flag` is a flag indicating the same offset as the left LCU, and `sao_merge_top_flag` is a flag
25 indicating the same offset as the upper LCU. When these flags are set, the flag that has been set is sent in place of the offset. In this case, an example where the determination is made based on only the offset has been explained. Alternatively, the offset setting unit 113
30 may provide not only the type but also the offset, and the determination may be made based on the type and the

offset.

[0185]

The offset buffer 115 holds the offset of an LCU adjacent to the current LCU. The offset buffer 115 may
5 hold at least the offset of the left LCU and the upper LCU of the current LCU.

[0186]

The filter processing unit 116 performs sample adaptive offset filter processing, for each LCU, on the
10 deblocked pixel value of each component given by the deblock filter 31 on the basis of the SAO flag, the type information, and the offset information given by the offset setting unit 113. The filter processing unit 116 provides the pixel value having been subjected to the
15 offset processing to the image buffer 117.

[0187]

The image buffer 117 temporarily stores the pixel values which have been subjected to the offset processing by the filter processing unit 116, and provides the pixel
20 values to the adaptive loop filter 42 with predetermined timing.

[0188]

The syntax write unit 121 writes the SAO flag given by the SAO flag setting unit 111 to the header portion of
25 the coded stream for each slice. The syntax write unit 121 writes, for each LCU, the type common to the components given by the type setting unit 112 and the offset or the merge flag for each component given by the SAO control information setting unit 114 to the header
30 portion of the coded stream.

[0189]

[Sample adaptive offset filter processing of encoding side]

Subsequently, the sample adaptive offset filter processing that is performed by the sample adaptive offset filter 41 of Fig. 9 will be explained with reference to the flowchart of Fig. 10. It should be noted that the sample adaptive offset filter processing is processing in step S23 of Fig. 2.

[0190]

10 The deblocked pixel values given by the deblock filter 31 are input into the SAO flag setting unit 111, the type setting unit 112, the offset setting unit 113, and the filter processing unit 116. In step S111, the SAO flag setting unit 111 uses the deblocked pixel values to set the SAO flag (`slice_sample_adaptive_offset_flag`) indicating the ON/OFF state of the sample adaptive offset filter of each component in units of slices. The SAO flag setting unit 111 provides the SAO flag, which has been set, to the type setting unit 112 and the syntax write unit 121.

[0191]

In step S112, the type setting unit 112 sets component ID(`cIdx`) = 0. In this case, `cIdx` = 0 indicates Y (brightness signal), `cIdx` = 1 indicates Cb (color difference signal), and `cIdx` = 2 indicates Cr (color difference signal).

[0192]

In step S113, the type setting unit 112 determines whether `slice_sample_adaptive_offset_flag`[0 or 1 or 2] = 1 holds or not. More specifically, in step S113, the sample adaptive offset filter of at least one of Y, Cb,

Cr is to be applied or not is determined.

[0193]

When slice_sample_adaptive_offset_flag[0 or 1 or 2] = 1 is determined to hold, and more specifically, when
5 the sample adaptive offset filter of at least one of Y, Cb, Cr is determined to be applied in step S113, the processing in step S114 is subsequently performed.

[0194]

In step S114, the type setting unit 112 uses the
10 deblocked pixel values to set the type (sao_type) of the filter which is common to the components of Y, Cb, Cr in units of LCUs. The type setting unit 112 provides the information indicating the type to the syntax write unit 121, and thereafter, the processing in step S115 is
15 subsequently performed.

[0195]

In association with this, the syntax write unit 121 writes sao_type to the syntax, provides it to the accumulation buffer 27, and transmits it in step S143 of
20 Fig. 12 explained later.

[0196]

When slice_sample_adaptive_offset_flag[0 and 1 and 2] = 0 is determined to hold, and more specifically, when sample adaptive offset filters of all of Y, Cb, Cr are
25 determined not to be applied in step S112, the processing in step S114 is skipped and step S115 is subsequently performed.

[0197]

The type setting unit 112 determines whether cIdx < 3 holds or not in step S115. When cIdx < 3 is determined
30 to hold in step S115, the processing in step S116 is

subsequently performed.

[0198]

In step S116, the type setting unit 112 determines whether slice_sample_adaptive_offset_flag[cIdx] = 1 holds
5 or not.

[0199]

When slice_sample_adaptive_offset_flag[cIdx] = 1 is determined to hold (more specifically, the sample adaptive offset filter is performed for cIdx) in step
10 S116, the processing in step S117 is subsequently performed. At this occasion, the type setting unit 112 provides the type of the filter which is set in step S114 and the SAO flag (slice_sample_adaptive_offset_flag) of corresponding cIdx which is set in step S111 to the
15 offset setting unit 113.

[0200]

In step S117, the offset setting unit 113 and the SAO control information setting unit 114 set the SAO control information for cIdx (= 0 (more specifically, Y
20 brightness signal)). This SAO control information setting processing will be explained later with reference to Fig. 11. With this processing, the offset or the merge flag is set as the SAO control information, and the SAO control information which has been set is provided to
25 the syntax write unit 121. In this case, the offset which has been set is provided to the filter processing unit 116.

[0201]

In association with this, the syntax write unit 121
30 writes the SAO control information for cIdx to the syntax, provides the SAO control information to the accumulation

buffer 27, and transmits the SAO control information in step S146 of Fig. 12 explained later.

[0202]

After step S117, the processing in step S118 is subsequently performed. In step S116, slice_sample_adaptive_offset_flag[cIdx] = 0 is determined to hold (more specifically, the sample adaptive offset filter for cIdx is not performed), the processing in step S117 is skipped and step S118 is subsequently performed.

10 [0203]

In step S118, the filter processing unit 116 performs the adaptive filter processing on the deblocked pixels given by the deblock filter 31. At this occasion, the adaptive filter processing is performed on the basis of the type of the filter which has been set in step S114, the SAO flag corresponding to cIdx which has been set in step S111, and further the offset having been set in step S117.

[0204]

20 The filter processing unit 116 provides the pixel values having been subjected to the offset processing to the image buffer 117. The image buffer 117 temporarily stores the pixel values having been subjected to the offset processing by the filter processing unit 116, and provides the pixel values to the adaptive loop filter 42 with predetermined timing.

[0205]

In step S119, the type setting unit 112 performs cIdx++ (more specifically, cIdx = 1 (which means Cb color difference signal)), and the processing in step S115 is performed back again. When cIdx < 3 is determined not to

30

hold in step S115, the processing in step S120 is subsequently performed.

[0206]

In step S120, the type setting unit 112 determines
5 whether the current LCU which is currently being
processed is the last LCU or not. When the current LCU
which is currently being processed is determined not to
be the last LCU in step S120, the processing in step S121
is subsequently performed. In step S121, the current LCU
10 is changed to a subsequent LCU, and the processing in
step S112 is performed back again.

[0207]

When the current LCU which is currently being
processed is determined to be the last LCU in step S120,
15 the sample adaptive offset filter is terminated.

[0208]

[SAO control information setting processing]

Subsequently, the SAO control information setting
processing in step S117 of Fig. 10 will be explained with
20 reference to the flowchart of Fig. 11.

[0209]

In step S131, the offset setting unit 113 uses the
deblocked pixel values to set the offsets (coefficients)
independent in each component of cIdx in units of LCUs,
25 and provides the offsets having been set, to the filter
processing unit 116 and the SAO control information
setting unit 114.

[0210]

In step S132, the SAO control information setting
30 unit 114 writes the offsets given by the offset setting
unit 113 to the offset buffer 115 for each component of Y,

Cb, Cr.

[0211]

In step S133, the SAO control information setting unit 114 determines whether the current LCU which is currently being processed can be merged with the left LCU adjacent to the left of the current LCU. More specifically, the SAO control information setting unit 114 refers to information written to the offset buffer 115, and determines whether the offset of the current LCU given by the offset setting unit 113 matches the offset of the left LCU.

[0212]

When the offset of the current LCU given by the offset setting unit 113 is determined to match the offset of the left LCU in step S133, and the current LCU is determined to be able to be merged with the left LCU, the processing in step S134 is subsequently performed.

[0213]

In step S134, the SAO control information setting unit 114 sets `sao_merge_left_flag`, and provides `sao_merge_left_flag`, which has been set, to the syntax write unit 121 as the SAO control information.

[0214]

On the other hand, when the current LCU is determined not to be able to be merged with the left LCU in step S133, the processing in step S135 is subsequently performed. In step S135, the SAO control information setting unit 114 determines whether the current LCU which is currently being processed can be merged with the upper LCU adjacent to the upper side of the current LCU. More specifically, the SAO control information setting unit

114 refers to the information written to the offset buffer 115, and determines whether the offset of the current LCU given by the offset setting unit 113 matches the offset of the upper LCU or not.

5 [0215]

When the offset of the current LCU given by the offset setting unit 113 is determined to match the offset of the upper LCU, and the current LCU is determined to be able to be merged with the upper LCU in step S135, the processing in step S136 is subsequently performed.

[0216]

In step S136, the SAO control information setting unit 114 sets `sao_merge_top_flag`, and provides `sao_merge_top_flag`, which has been set, to the syntax write unit 121 as the SAO control information.

15 [0217]

On the other hand, when the current LCU is determined not to be able to be merged with the upper LCU in step S135, the processing in step S137 is subsequently performed.

20 [0218]

In step S137, the SAO control information setting unit 114 provides the offset to the syntax write unit 121 as the SAO control information.

25 [0219]

[SAO control information transmission processing for each LCU]

Subsequently, SAO control information transmission processing for each LCU with the conventional example shown in Fig. 8 will be explained with reference to the flowchart of Fig. 12. This processing corresponds to the

30

sample adaptive offset filter processing explained with reference to Fig. 10, and is, for example, processing performed in the encoding processing of step S26 of Fig. 2.

5 [0220]

The syntax write unit 121 sets component ID(cIdx) = 0 in step S141.

[0221]

In step S142, the syntax write unit 121 determines
10 whether slice_sample_adaptive_offset_flag[0 or 1 or 2] = 1 holds or not. More specifically, a determination is made as to whether the sample adaptive offset filter of at least one of Y, Cb, Cr is applied or not.

[0222]

15 When slice_sample_adaptive_offset_flag[0 or 1 or 2] = 1 is determined to hold in step S142, the processing in step S143 is subsequently performed. In step S143, the syntax write unit 121 writes information indicating the type (sao_type) of the filter provided and set in step
20 S114 of Fig. 10 to the syntax, provides the information to the accumulation buffer 27 and transmits the information. The information indicating the type is transmitted in units of PPSes (Picture Parameter Sets) or slices.

25 [0223]

In step S142, when slice_sample_adaptive_offset_flag[0 and 1 and 2] = 0 is determined to hold, step S143 is skipped, and the processing in step S144 is subsequently performed.

30 [0224]

In step S144, the syntax write unit 121 determines

whether $cIdx < 3$ holds or not. When $cIdx < 3$ is determined to hold in step S144, the processing in step S145 is subsequently performed.

[0225]

5 In step S145, the syntax write unit 121 determines whether $slice_sample_adaptive_offset_flag[cIdx] = 1$ holds or not. When $slice_sample_adaptive_offset_flag[cIdx] = 1$ is determined to hold (more specifically, when the sample adaptive offset filter for $cIdx$ is to be performed) in
10 step S145, the processing in step S146 is subsequently performed.

[0226]

 In step S146, the syntax write unit 121 writes, to the syntax, the SAO control information (except sao_type)
15 for $cIdx$ ($= 0$ (which is the Y brightness signal)) set and provided in step S117 of Fig. 10, provides it to the accumulation buffer 27, and transmits it.

[0227]

 In step S147, the syntax write unit 121 performs
20 $cIdx++$ (more specifically, $cIdx = 1$ (which is the Cr color difference signal)), and the processing in step S144 is performed back again. When $cIdx < 3$ is determined not to hold in step S144, the processing in step S148 is subsequently performed.

25 [0228]

 In step S148, the syntax write unit 121 determines whether the current LCU which is currently being processed is the last LCU or not. When the current LCU which is currently being processed is determined not to
30 be the last LCU in step S148, the processing in step S149 is subsequently performed. In step S149, the current LCU

is changed to a subsequent LCU, and the processing in step S141 is performed back again.

[0229]

In step S148, when the current LCU which is
5 currently being processed is determined to be the last LCU, the SAO control information transmission processing for each LCU is terminated.

[0230]

As described above, the information indicating the
10 type of the filter is made common between the components of Y, Cb, Cr, and more specifically, the information indicating the type of the filter is made common between the components of the brightness and the color difference, and is transmitted, so that only the coefficients
15 (offsets) are transmitted independently between each of the components, whereby the amount of encoding can be reducing.

[0231]

[Another example of configuration of sample
20 adaptive offset filter]

Fig. 13 is a block diagram illustrating another example of configuration of a sample adaptive offset filter and a lossless coding unit of an image coding device of Fig. 1.

25 [0232]

In the example of Fig. 13, the sample adaptive offset filter 41 is the same as the sample adaptive offset filter 41 of Fig. 9 in that it includes an SAO flag setting unit 111, an offset setting unit 113, an SAO
30 control information setting unit 114, an offset buffer 115, a filter processing unit 116, and an image buffer

117.

[0233]

The sample adaptive offset filter 41 is different from the sample adaptive offset filter 41 of Fig. 9 in that the type setting unit 112 is replaced with the type setting unit 141.

[0234]

The lossless coding unit 26 of Fig. 13 is different from the lossless coding unit 26 of Fig. 9 in that the syntax write unit 121 is replaced with the syntax write unit 161.

[0235]

More specifically, the type setting unit 141 of Fig. 13 includes a Y type setting unit 151 configured to set the filter type independent in the brightness signal and a Cm type setting unit 152 configured to set the filter type common to the color difference signals of Cb and Cr.

[0236]

For example, the Y type setting unit 151 uses the brightness signal to calculate the cost function value for each LCU, and sets the optimum type for the LCU of the brightness signal of which cost function value is the least. For example, the Cm type setting unit 152 uses at least one of Cb and Cr to calculate the cost function value for each LCU, and sets the optimum type to the LCU of the color difference signal of which cost function value is the least.

[0237]

Like the syntax write unit 121 of Fig. 9, the syntax write unit 161 writes the SAO flag given by the SAO flag setting unit 111 to the header portion of the

coded stream for each slice. Like the syntax write unit 121, the syntax write unit 161 writes the offset or merge flag for each component, which is given by the SAO control information setting unit 114, to the header
5 portion of the coded stream.

[0238]

Unlike the syntax write unit 121, the syntax write unit 161 writes, for each LCU, the type information indicating the Y type for the brightness given by the
10 type setting unit 112 and the type information indicating Cm type common to the color difference components to the header portion of the coded stream.

[0239]

[Sample adaptive offset filter processing of
15 encoding side]

Subsequently, the sample adaptive offset filter processing performed by the sample adaptive offset filter 41 of Fig. 13 will be explained with reference to the flowchart of Fig. 14. It should be noted that the sample
20 adaptive offset filter processing is processing in step S23 of Fig. 2.

[0240]

In step S161, the SAO flag setting unit 111 uses the deblocked pixel values to set the SAO flag indicating
25 the ON/OFF state of the sample adaptive offset filter of each component in units of slices. The SAO flag setting unit 111 provides the SAO flag
(slice_sample_adaptive_offset_flag), which has been set, to the Y type setting unit 151, the Cm type setting unit
30 152, and the syntax write unit 161.

[0241]

In step S162, the Y type setting unit 151 and the Cm type setting unit 152 set component ID(cIdx) = 0.

[0242]

In step S163, the Y type setting unit 151
5 determines whether slice_sample_adaptive_offset_flag[0] = 1 holds or not. More specifically, in step S163, the sample adaptive offset filter of Y is to be applied or not is determined.

[0243]

10 When slice_sample_adaptive_offset_flag[0] = 1 is determined to hold, and more specifically when the sample adaptive offset filter of Y is determined to be applied in step S163, the processing in step S164 is subsequently performed.

15 [0244]

In step S164, the Y type setting unit 151 uses the deblocked pixel values to set the type (sao_type_Y) of the filter of Y in units of LCUs. The Y type setting unit 151 provides the information indicating the type of
20 Y to the syntax write unit 161, and thereafter, the processing in step S165 is subsequently performed.

[0245]

In association with this, the syntax write unit 161 writes sao_type_Y to the syntax, and provides it to the
25 accumulation buffer 27 and transmits it in step S183 of Fig. 15 explained later.

[0246]

When slice_sample_adaptive_offset_flag[0] = 0 is determined to hold in step S163, the processing in step
30 S164 is skipped and step S165 is subsequently performed.

[0247]

In step S165, the Cm type setting unit 152 determines whether slice_sample_adaptive_offset_flag[1 or 2] = 1 holds or not. More specifically, in step S165, a determination is made as to whether the sample adaptive offset filter of Cr or Cb is to be applied or not.
[0248]

When slice_sample_adaptive_offset_flag[1 or 2] = 1 is determined to hold, and more specifically, when the sample adaptive offset filter of Cm is determined to be applied in step S165, the processing in step S166 is subsequently performed.
[0249]

In step S166, the Cm type setting unit 152 uses the deblocked pixel values to set the type (sao_type_Cm) of the filter common to Cb and Cr in units of LCUs. The Y type setting unit 151 provides information indicating the type of Cm to the syntax write unit 161, and thereafter, the processing in step S167 is subsequently performed.
[0250]

In association with this, the syntax write unit 161 writes sao_type_Cm to the syntax, provides it to the accumulation buffer 27, and transmits it in step S185 of Fig. 15 explained later.
[0251]

When slice_sample_adaptive_offset_flag[1 and 2] = 0 is determined to hold in step S165, the processing in step S166 is skipped and step S167 is subsequently performed.
[0252]

In S167 to S173 subsequent thereto, basically the same processing as step S115 to S121 of Fig. 10 is

performed, therefore, the explanation thereabout is omitted because it is simply a repetition.

[0253]

[SAO control information transmission processing
5 for each LCU]

Subsequently, SAO control information transmission processing for each LCU with the conventional example shown in Fig. 8 will be explained with reference to the flowchart of Fig. 15. This processing corresponds to the
10 sample adaptive offset filter processing explained with reference to Fig. 14, and is, for example, processing performed in the encoding processing of step S26 of Fig. 2.

[0254]

15 In step S181, the syntax write unit 161 sets component ID(cIdx) = 0.

[0255]

In step S182, the syntax write unit 161 determines whether slice_sample_adaptive_offset_flag[0] = 1 holds or
20 not. More specifically, a determination is made as to whether the sample adaptive offset filter of Y is applied or not.

[0256]

When slice_sample_adaptive_offset_flag[0] = 1 is
25 determined to hold in step S182, the processing in step S183 is subsequently performed. In step S183, the syntax write unit 161 writes information indicating the type (sao_type_Y) of the filter provided in step S164 of Fig. 14 to the syntax, provides the information to the
30 accumulation buffer 27, and transmits the information.

[0257]

When slice_sample_adaptive_offset_flag[0] = 0 is determined to hold in step S182, the processing in step S183 is skipped and step S184 is subsequently performed. [0258]

5 In step S184, the syntax write unit 161 determines whether slice_sample_adaptive_offset_flag[1 or 2] = 1 holds or not. More specifically, a determination is made as to whether the sample adaptive offset filter of Cm is applied or not.

10 [0259]

When slice_sample_adaptive_offset_flag[1 or 2] = 1 is determined to hold in step S184, the processing in step S185 is subsequently performed. In step S185, the syntax write unit 161 writes information indicating the type (sao_type_Cm) of the filter provided in step S166 of Fig. 14 to the syntax, provides the information to the accumulation buffer 27, and transmits the information. [0260]

When slice_sample_adaptive_offset_flag[1 and 2] = 0 is determined to hold in step S184, step S185 is skipped, and the processing in step S186 is subsequently performed. [0261]

In S186 to S191 subsequent thereto, basically the same processing as step S144 to S149 of Fig. 14 is performed, therefore, the explanation thereabout is omitted because it is simply a repetition. [0262]

As described above, the type information about the filter can also be made common only between the components of the color differences of Cb and Cr, and in this case, the amount of codes can also be reduced.

[0263]

<4. Fifth embodiment>

[Example of configuration of sample adaptive offset filter]

5 Fig. 16 is a block diagram illustrating an example of configuration of a lossless decoding unit and a sample adaptive offset filter in an image decoding device of Fig. 3. An example of configuration as shown in Fig. 16 corresponds to the sample adaptive offset filter and the
10 lossless coding unit of Fig. 9.

[0264]

 In the example of Fig. 16, the lossless decoding unit 62 includes at least a syntax acquisition unit 211.

[0265]

15 The sample adaptive offset filter 81 includes an SAO flag reception unit 221, a type reception unit 222, a merge flag reception unit 223, an offset buffer 224, an offset reception unit 225, a filter processing unit 226, and an image buffer 227.

20 [0266]

 The syntax acquisition unit 211 obtains the header portion of the coded stream from the syntax. Then, the syntax acquisition unit 211 provides the adaptive offset parameter in the obtained syntax to the SAO flag
25 reception unit 221, the type reception unit 222, the merge flag reception unit 223, and the offset reception unit 225.

[0267]

 More specifically, the syntax acquisition unit 211
30 provides an SAO flag (slice_sample_adaptive_offset_flag) which is one of the adaptive offset parameters to the SAO

flag reception unit 221. The syntax acquisition unit 211 provides the type information of the filter which is one of the adaptive offset parameters to the type reception unit 222. The syntax acquisition unit 211 provides the
 5 merge flag (sao_merge_left_flag or sao_merge_top_flag) which is one of the adaptive offset parameters to the merge flag reception unit 223. The syntax acquisition unit 211 provides the offset (coefficient) which is one of the adaptive offset parameters to the offset reception
 10 unit 225.

[0268]

It should be noted that the SAO flag of the adaptive offset parameters is set and transmitted for each slice. The type information, the merge flag, the
 15 offset, and the like in the adaptive offset parameters are set and transmitted for each LCU. The type information is transmitted, for example, in unit of PPS (Picture Parameter Set) or slice.

[0269]

20 The SAO flag reception unit 221 provides the filter processing unit 226 with the SAO flag indicating the ON/OFF state for each component provided from the syntax acquisition unit 211. The type reception unit 222 provides the filter processing unit 226 with the filter
 25 type information which is common to the components provided from the syntax acquisition unit 211. The merge flag reception unit 223 provides the offset buffer 224 with the merge flag provided from the syntax acquisition unit 211. The offset reception unit 225 provides the
 30 offset obtained from the syntax acquisition unit 211 to the offset buffer 224 and the filter processing unit 226.

[0270]

The offset buffer 224 holds the offset of an LCU adjacent to the current LCU. The offset buffer 224 may store at least the left LCU of the current LCU and the
5 offset of the upper LCU. When the offset buffer 224 receives the merge flag of the current LCU from the merge flag reception unit 223, the offset buffer 224 reads the offset stored in association with the adjacent LCU (left or upper side) indicated by the merge flag, and provides
10 it to the filter processing unit 226. Then, the offset buffer 224 stores the read offset in association with the current LCU. When the offset buffer 224 receives the offset of the current LCU from the offset reception unit 225, the offset buffer 224 stores the received offset in
15 association with the current LCU.

[0271]

The filter processing unit 226 refers to information given by the SAO flag reception unit 221, the type reception unit 222, the offset buffer 224, and the
20 offset reception unit 225, and performs filter processing on the deblocked pixel values from the deblock filter 66.

[0272]

More specifically, the filter processing unit 226 refers to the SAO flag given by the SAO flag reception
25 unit 221, the filter type which is made common between components and which is provided from the type reception unit 222, and the offset given by the offset buffer 224 or the offset reception unit 225.

[0273]

30 The filter processing unit 226 provides the pixels, which have been subjected to the offset processing, to

the image buffer 227.

[0274]

The image buffer 227 is configured basically in the same way as the image buffer 117 of Fig. 10. More specifically, the image buffer 227 temporarily stores the pixel values having been subjected to the offset processing by the filter processing unit 226, and provides the pixel values to the adaptive loop filter 82 with predetermined timing.

10 [0275]

[Sample adaptive offset filter processing of decoding side]

Subsequently, the sample adaptive offset filter processing performed by the sample adaptive offset filter 81 of Fig. 16 will be explained with reference to the flowchart of Fig. 17. It should be noted that the sample adaptive offset filter processing is processing in step S59 of Fig. 4.

[0276]

20 When the coded stream is decoded in step S52 of Fig. 4, the syntax acquisition unit 211 reads the syntax from the header portion of the coded stream, and among them, the syntax acquisition unit 211 provides the adaptive offset parameter to each unit of the sample adaptive offset filter 81.

25

[0277]

In step S211, the SAO flag reception unit 221 receives the SAO flag of the adaptive offset parameter (slice_sample_adaptive_offset_flag) provided from the syntax acquisition unit 211. The SAO flag reception unit 221 provides the received SAO flag to the filter

30

processing unit 226.

[0278]

In step S212, the filter processing unit 226 sets component ID(cIdx) = 0.

5 [0279]

In step S213, the filter processing unit 226 determines whether slice_sample_adaptive_offset_flag[0 or 1 or 2] = 1 or not. More specifically, in step S213, a determination is made as to whether the sample adaptive offset filter of at least one of Y, Cb, Cr is applied or not.

[0280]

When slice_sample_adaptive_offset_flag[0 or 1 or 2] = 1 is determined to hold in step S213, and more specifically, when the sample adaptive offset filter of at least one of Y, Cb, Cr is determined to be applied, the processing in step S214 is subsequently performed.

[0281]

In step S214, the filter processing unit 226 receives the filter type information (sao_type), which is made common between the components, given by the syntax acquisition unit 211 via the type reception unit 222.

[0282]

When slice_sample_adaptive_offset_flag[0 and 1 and 2] = 0 is determined to hold, and more specifically, when the sample adaptive offset filters of all of Y, Cb, Cr are determined not to be applied in step S213, the processing in step S214 is skipped and step S215 is subsequently performed.

30 [0283]

In step S215, the filter processing unit 226

determines whether $cIdx < 3$ holds or not. When $cIdx < 3$ is determined to hold in step S215, the processing in step S216 is subsequently performed.

[0284]

5 In step S216, the filter processing unit 226 determines whether $slice_sample_adaptive_offset_flag[cIdx] = 1$ holds or not.

[0285]

 When $slice_sample_adaptive_offset_flag[cIdx] = 1$ is
10 determined to hold (more specifically, the sample adaptive offset filter is performed on $cIdx$) in step S216, the processing in step S217 is subsequently performed.

[0286]

 In step S217, the filter processing unit 226
15 receives the SAO control information for $cIdx (= 0)$. This SAO control information reception processing will be explained later with reference to Fig. 18, and with this processing, the merge flag or the offset is received, and the offset which is read with the received merge flag or
20 the received offset is provided to the filter processing unit 226.

[0287]

 After step S217, the processing in step S218 is subsequently performed. When
25 $slice_sample_adaptive_offset_flag[cIdx] = 0$ is determined to hold (more specifically, the sample adaptive offset filter for $cIdx$ is not performed) in step S216, the processing in step S217 is skipped and step S218 is subsequently performed.

30 [0288]

 In step S218, the filter processing unit 226

performs the adaptive filter processing on the pixels having been subjected to the deblock processing given by the deblock filter 66. At this occasion, the adaptive filter processing is performed on the basis of the type
5 of the filter which has been received in step S214, the SAO flag corresponding to cIdx received in step S211, and further the offset having been obtained in step S217.

[0289]

The filter processing unit 226 provides the pixel
10 values having been subjected to the offset processing to the image buffer 227. The image buffer 227 temporarily stores the pixel values having been subjected to the offset processing by the filter processing unit 226, and provides the pixel values to the adaptive loop filter 42
15 with predetermined timing.

[0290]

In step S219, the filter processing unit 226 performs $cIdx++$ (more specifically, $cIdx = 1$), and the processing in step S215 is performed back again. When
20 $cIdx < 3$ is determined not to hold in step S215, the processing in step S220 is subsequently performed.

[0291]

In step S220, the filter processing unit 226 determines whether the current LCU which is currently
25 being processed is the last LCU or not. When the current LCU which is currently being processed is determined not to be the last LCU in step S220, the processing in step S221 is subsequently performed. In step S221, the current LCU is changed to a subsequent LCU, and the
30 processing in step S212 is performed back again.

[0292]

When the current LCU which is currently being processed is determined to be the last LCU in step S220, the sample adaptive offset filter processing is terminated.

5 [0293]

[SAO control information setting processing]

Subsequently, the SAO control information reception processing in step S217 of Fig. 17 will be explained with reference to the flowchart of Fig. 18.

10 [0294]

In step S231, the merge flag reception unit 223 receives the merge flag. In step S232, the merge flag reception unit 223 determines whether `sao_merge_left_flag` is one or not. When `sao_merge_left_flag` is determined to be one in step S232, the processing in step S233 is subsequently performed.

[0295]

In step S233, the merge flag reception unit 223 controls the offset buffer 224, and reads the offset of the left LCU, and the offset buffer 224 provides the offset of the left LCU which has been read to the filter processing unit 226.

[0296]

On the other hand, when `sao_merge_left_flag` is determined to be zero in step S232, the processing in step S234 is subsequently performed. In step S234, the merge flag reception unit 223 determines whether `sao_merge_top_flag` is one or not. When `sao_merge_top_flag` is determined to be one in step S234, the processing in step S235 is subsequently performed.

30

[0297]

In step S235, the merge flag reception unit 223 controls the offset buffer 224, and reads the offset of the upper LCU, and the offset buffer 224 provides the offset of the upper LCU which has been read to the filter processing unit 226.

[0298]

When sao_merge_top_flag is determined to be zero in step S234, the processing in step S236 is subsequently performed. In step S236, the offset reception unit 225 receives the offset from the syntax acquisition unit 211, and provides the offset to the filter processing unit 226. The offset reception unit 225 also provides the received offset to the offset buffer 224.

[0299]

In step S237, the offset buffer 224 writes the offset read in step S233 or S235 or the offset from the offset reception unit 225 to the buffer as the offset of the current LCU.

[0300]

As described above, the information indicating the type of the filter is sent upon being made common between the components of Y, Cb, Cr, and more specifically, the information indicating the type of the filter is sent upon being made common between the components of the brightness and the color difference, and therefore, only the coefficients (offsets) are transmitted independently between the components, so that the amount of encoding can be reduced.

[0301]

[Another example of configuration of sample

adaptive offset filter]

Fig. 19 is a block diagram illustrating an example of configuration of a lossless decoding unit and a sample adaptive offset filter in the image decoding device of Fig. 3. The example of configuration as shown in Fig. 19 corresponds to the sample adaptive offset filter and the lossless coding unit of Fig. 13.

[0302]

In the example of Fig. 19, the lossless decoding unit 62 is different from the lossless decoding unit 62 of Fig. 16 in that the syntax acquisition unit 211 and the syntax acquisition unit 241 are switched.

[0303]

The sample adaptive offset filter 81 is the same as the sample adaptive offset filter 81 of Fig. 16 in that the sample adaptive offset filter 81 includes an SAO flag reception unit 221, a merge flag reception unit 223, an offset buffer 224, an offset reception unit 225, a filter processing unit 226, and an image buffer 227.

[0304]

The sample adaptive offset filter 81 is different from the sample adaptive offset filter 81 of Fig. 16 in that the type reception unit 222 is replaced with the type reception unit 251 made of the Y type reception unit 261 and the Cm type reception unit 262.

[0305]

More specifically, in the example of Fig. 16, the type information about the filter which is made common between Y, Cb, Cr is sent as one of the adaptive offset parameters. In contrast, in the example of Fig. 19, the type information about the filter of Y and the type

information about the filter which is made common between Cb and Cr is sent.

[0306]

Like the syntax acquisition unit 211 of Fig. 16,
5 the syntax acquisition unit 241 obtains the SAO flag
(slice_sample_adaptive_offset_flag), and provides it to
the SAO flag reception unit 221. Like the syntax
acquisition unit 211 of Fig. 16, the syntax acquisition
unit 241 obtains the merge flag (sao_merge_left_flag or
10 sao_merge_top_flag), and provides it to the merge flag
reception unit 223. Like the syntax acquisition unit 211
of Fig. 16, the syntax acquisition unit 241 obtains the
offset, and provides it to the offset reception unit 225.

[0307]

15 On the other hand, unlike the syntax acquisition
unit 211 of Fig. 16, the syntax acquisition unit 241
obtains the type information about the filter, which is
made common to Cb and Cr, and the type information about
the filter of Y, and provides them to the Y type
20 reception unit 261 and the Cm type reception unit 262.

[0308]

The Y type reception unit 261 provides the filter
type information about Y provided from the syntax
acquisition unit 211 to the filter processing unit 226.
25 The Cm type reception unit 262 provides the filter type
information of Cm, which is made common to Cb and Cr
provided from the syntax acquisition unit 211, to the
filter processing unit 226.

[0309]

30 [Sample adaptive offset filter processing of
decoding side]

Subsequently, the sample adaptive offset filter processing performed by the sample adaptive offset filter 81 of Fig. 19 will be explained with reference to the flowchart of Fig. 20. It should be noted that this sample adaptive offset filter processing is processing in step S59 of Fig. 4.

[0310]

In step S261, the SAO flag reception unit 221 receives the SAO flag (`slice_sample_adaptive_offset_flag`) of the adaptive offset parameter provided from the syntax acquisition unit 211. The SAO flag reception unit 221 provides the received SAO flag to the filter processing unit 226.

[0311]

In step S262, the filter processing unit 226 sets component ID(`cIdx`) = 0. In step S263, the filter processing unit 226 determines whether `slice_sample_adaptive_offset_flag[0]` = 1 holds or not. More specifically, in step S263, a determination is made as to whether the sample adaptive offset filter of Y is applied or not.

[0312]

When `slice_sample_adaptive_offset_flag[0]` = 1 is determined to hold, and more specifically, the sample adaptive offset filter of Y is determined to be applied in step S263, the processing in step S264 is subsequently performed.

[0313]

In step S264, the filter processing unit 226 receives the filter type information of Y (`sao_type_Y`) given by the syntax acquisition unit 241 via the Y type

reception unit 261.

[0314]

On the other hand, in step S263, when
slice_sample_adaptive_offset_flag[0] = 0 is determined to
5 hold, and more specifically, the sample adaptive offset
filter of Y is determined not to be applied, the
processing in step S264 is skipped and step S265 is
subsequently performed.

[0315]

10 In step S265, the filter processing unit 226
determines whether slice_sample_adaptive_offset_flag[1 or
2] = 1 holds or not. In step S265, a determination is
made as to whether the sample adaptive offset filter of
Cb or Cr is applied or not.

15 [0316]

When slice_sample_adaptive_offset_flag[1 or 2] = 1
is determined to hold, and more specifically, when the
sample adaptive offset filter of Cb or Cr is determined
to be applied in step S265, the processing in step S266
20 is subsequently performed.

[0317]

In step S266, the filter processing unit 226
receives the filter type information of Cm (sao_type_Cm)
given by the syntax acquisition unit 241 via the Cm type
25 reception unit 262.

[0318]

When slice_sample_adaptive_offset_flag[1 and 2] = 0
is determined to hold, and more specifically, when the
sample adaptive offset filter of Cb and Cr is determined
30 not to be applied in step S265, the processing in step
S266 is skipped and step S267 is subsequently performed.

[0319]

The processing in S267 to S273 subsequent thereto is basically the same processing as steps S215 to S221 of Fig. 17, and therefore, the explanation thereabout is
5 omitted because it is simply a repetition.

[0320]

As described above, the type information about the filter can be made common only between the components of color differences of Cb and Cr. In this case, the amount
10 of codes can be reduced.

[0321]

In the above explanation, the example in which the type is made common between all the components of the brightness and the chromaticity and the example in which
15 the type is made common between the components of the color differences have been explained. Alternatively, both of the modes employing the above may be provided, and the modes may be selectively used.

[0322]

20 More specifically, a mode in which the type is made common between all the components of the brightness and the chromaticity is defined as an all-common mode, and a mode in which the type is made common only between the components of the color difference is defined as a color
25 difference common mode, and these two types of modes may be provided. In this case, the mode identification information for identifying which of the modes is used by the encoding side may be sent to the decoding side.

[0323]

30 In this case, the decoding side receives the mode identification information which has been transmitted,

sets the mode using the received mode identification information, and performs the sample adaptive offset filter.

[0324]

5 As described above, the type information about the filter is made common between the components of the brightness signal and the color difference signal and is transmitted. Therefore, the amount of codes can be reduced, and the efficiency of encoding can be increased.

10 [0325]

In the above explanation, the color format is 4:2:0. Alternatively, the present technique can be applied to any other color formats such as 4:2:2 and 4:4:4.

[0326]

15 For example, when `choma_format_idc` is used: in a case of 4:2:0 (`choma_format_idc==1`), the SAO type may be made common between all of the components; in a case of 4:4:4 (`choma_format_idc==3`), the SAO type may not be made common; and in a case of 4:2:2 (`choma_format_idc==2`), the
20 SAO type may be made common only between Cb/Cr. More specifically, according to the color format, the all-common mode or the color difference common mode explained above may be determined.

[0327]

25 In the above explanation, an example in which the brightness and the color difference are color space of Y, Cb, Cr has been explained. Alternatively, the present technique can also be applied to another color space, e.g., where the brightness and color difference are Y, U,
30 V.

[0328]

In the above explanation, the HEVC method is used as the basis of the encoding method. However, the present disclosure is not limited thereto. Other encoding methods/decoding methods including at least the sample adaptive offset filter as in-loop filter may be applied.

[0329]

Further, for example, the present disclosure can be applied to an image coding device and an image decoding device which are used for receiving image information (bit stream) compressed by orthogonal transformation such as discrete cosine transform and motion compensation similarly to HEVC method and the like, via network media such as satellite broadcasting, cable television, the Internet, and cellular phone. The present disclosure can be applied to an image coding device and an image decoding device used for processing on recording media such as optical, magnetic disks, and flash memories.

[0330]

<5. Third embodiment>

[Application to multi-view image point coding/multi-viewpoint image decoding]

The above series of processing can be applied to multi-viewpoint image coding/multi-viewpoint image decoding. Fig. 21 illustrates an example of multi-viewpoint image coding method.

[0331]

As illustrated in Fig. 21, a multi-viewpoint image includes images for multiple view points, and images of predetermined viewpoint of the multiple viewpoints are designated as base view images. Images of viewpoints

other than the base view image are treated as non-base view images.

[0332]

In the case where the multi-viewpoint image
5 encoding as shown in Fig. 21 is performed, the sample
adaptive offset filter parameters (the SAO flag, the type
information of the filter, the offset, the mode
identification information, and the like explained above)
can be set in each view (the same view). In each view (a
10 different view), the sample adaptive offset filter
parameters which are set in another view may be shared.

[0333]

In this case, the sample adaptive offset filter
parameters which are set in the base view are used in at
15 least a single non-base view. Alternatively, for example,
the sample adaptive offset filter parameters which are
set in a non-base view (view_id = i) are used in at least
one of the base view and the non-base view (view_id = j).

[0334]

20 Therefore, the amount of codes can be reduced.

[0335]

[Multi-viewpoint image coding device]

Fig. 22 is a figure illustrating a multi-viewpoint
image coding device performing the multi-viewpoint image
25 coding explained above. As illustrated in Fig. 22, a
multi-viewpoint image coding device 600 includes an
encoding unit 601, an encoding unit 602, and a
multiplexing unit 603.

[0336]

30 The encoding unit 601 encodes base view images, and
generates a base view image coded stream. The encoding

unit 602 encodes non-base view images, and generates a non-base view image coded stream. The multiplexing unit 603 multiplexes the base view image coded stream generated by the encoding unit 601 and the non-base view image coded stream generated by the encoding unit 602, and generates a multi-viewpoint image coded stream.

[0337]

The image coding device 11 (Fig. 1) can be applied to the encoding unit 601 and encoding unit 602 of the multi-viewpoint image coding device 600. In this case, the multi-viewpoint image coding device 600 sets and transmits the sample adaptive offset filter parameters which are set by the encoding unit 601 and the sample adaptive offset filter parameters which are set by the encoding unit 602.

[0338]

The sample adaptive offset filter parameters which are set by the encoding unit 601 as described above may be set and transmitted so that the sample adaptive offset filter parameters are may be shared and used by the encoding unit 601 and the encoding unit 602. On the contrary, the sample adaptive offset filter parameters which are collectively set by the encoding unit 602 may be set and transmitted so that the sample adaptive offset filter parameters are shared and used by the encoding unit 601 and the encoding unit 602.

[0339]

[Multi-viewpoint image decoding device]

Fig. 23 is a figure illustrating a multi-viewpoint image decoding device that performs the multi-viewpoint image decoding explained above. As illustrated in Fig.

23, the multi-viewpoint image decoding device 610 includes a demultiplexing unit 611, a decoding unit 612, and a decoding unit 613.

[0340]

5 The demultiplexing unit 611 demultiplexes the multi-viewpoint image coded stream obtained by multiplexing the base view image coded stream and the non-base view image coded stream, and extracts the base view image coded stream and the non-base view image coded
10 stream. The decoding unit 612 decodes the base view image coded stream extracted by the demultiplexing unit 611, and obtains the base view images. The decoding unit 613 decodes the non-base view image coded stream extracted by the demultiplexing unit 611, and obtains the
15 non-base view images.

[0341]

 The image decoding device 51 (Fig. 3) can be applied to the decoding unit 612 and decoding unit 613 of the multi-viewpoint image decoding device 610. In this
20 case, the multi-viewpoint image decoding device 610 performs processing using the sample adaptive offset filter parameters which is set by the encoding unit 601 and decoded by the decoding unit 612 and the sample adaptive offset filter parameters which is set by the
25 encoding unit 602 and decoded by the decoding unit 613.

[0342]

 The sample adaptive offset filter parameters which are set by the encoding unit 601 (or the encoding unit 602) may be set and transmitted so that the sample
30 adaptive offset filter parameters are shared and used by the encoding unit 601 and the encoding unit 602. In this

case, the multi-viewpoint image decoding device 610 performs processing using the sample adaptive offset filter parameters which are set by the encoding unit 601 (or the encoding unit 602) and decoded by the decoding unit 612 (or the decoding unit 613).

[0343]

<6. Fourth embodiment>

[Application to hierarchical image point coding/hierarchical image decoding]

10 The above series of processing can be applied to hierarchical image coding/hierarchical image decoding. Fig. 24 illustrates an example of multi-viewpoint image coding method.

[0344]

15 As illustrated in Fig. 24, a hierarchical image includes images of multiple hierarchical (resolution), and a hierarchical image of a predetermined one of the multiple resolution is designated as a base layer image. Images of hierarchies other than the base layer image are
20 treated as non-base layer images.

[0345]

 When the hierarchical image encoding (spatial scalability) as shown in Fig. 24 is performed, the sample adaptive offset filter parameters can be set in each
25 layer (the same layer). In each layer (a different layer), the sample adaptive offset filter parameters which are set in another layer can be shared.

[0346]

 In this case, the sample adaptive offset filter
30 parameters which are set in the base layer are used in at least a single non-base layer. Alternatively, for

example, the sample adaptive offset filter parameters which are set in a non-base layer ($\text{layer_id} = i$) are used in at least one of base layer and non-base layer ($\text{layer_id} = j$).

5 [0347]

Therefore, the amount of codes can be reduced.

[0348]

[Hierarchical image coding device]

Fig. 25 is a figure illustrating a hierarchical
10 image coding device that performs the hierarchical image coding explained above. As illustrated in Fig. 25, the hierarchical image coding device 620 includes an encoding unit 621, an encoding unit 622, and a multiplexing unit 623.

15 [0349]

The encoding unit 621 encodes base layer images, and generates a base layer image coded stream. The encoding unit 622 encodes non-base layer images, and generates a non-base layer image coded stream. The
20 multiplexing unit 623 multiplexes the base layer image coded stream generated by the encoding unit 621 and the non-base layer image coded stream generated by the encoding unit 622, and generates a hierarchical image coded stream.

25 [0350]

The image coding device 11 (Fig. 1) can be applied to the encoding unit 621 and the encoding unit 622 of the hierarchical image coding device 620. In this case, the hierarchical image coding device 620 sets the
30 identification information which is set by the encoding unit 621 and the sample adaptive offset filter parameters

which is set by the encoding unit 622, and transmits the sample adaptive offset filter parameters.

[0351]

It should be noted that such configuration may be adopted that the sample adaptive offset filter parameters which is set by the encoding unit 621 as described above is set so as to be shared and used in the encoding unit 621 and the encoding unit 622, and is transmitted. On the contrary, such configuration may be adopted that the sample adaptive offset filter parameters which is set by the encoding unit 622 is set so as to be shared and used in the encoding unit 621 and the encoding unit 622, and is transmitted.

[0352]

[Hierarchical image decoding device]

Fig. 26 is a figure illustrating a hierarchical image decoding device that performs the hierarchical image decoding explained above. As illustrated in Fig. 26, the hierarchical image decoding device 630 includes a demultiplexing unit 631, a decoding unit 632, and a decoding unit 633.

[0353]

The demultiplexing unit 631 demultiplexes the hierarchical image coded stream obtained by multiplexing the base layer image coded stream and the non-base layer image coded stream, and extracts the base layer image coded stream and the non-base layer image coded stream. The decoding unit 632 decodes the base layer image coded stream extracted by the demultiplexing unit 631, and obtains the base layer image. The decoding unit 633 decodes the non-base layer image coded stream extracted

by the demultiplexing unit 631, and obtains the non-base layer image.

[0354]

The image decoding device 51 (Fig. 3) can be
5 applied to the decoding unit 632 and the decoding unit 633 of the hierarchical image decoding device 630. In this case, the hierarchical image decoding device 630 performs processing by using the sample adaptive offset filter parameters which is set by the encoding unit 621
10 and which is decoded by the decoding unit 632 and the decoded sample adaptive offset filter parameters which is set by the encoding unit 622 and which is decoded by the decoding unit 633.

[0355]

15 It should be noted that, the sample adaptive offset filter parameters which is set by the encoding unit 621 (or, the encoding unit 622) described above may be set so as to be shared and used in the encoding unit 621 and the encoding unit 622, and is transmitted. In this case, in
20 the hierarchical image decoding device 630 the processing is performed by using the sample adaptive offset filter parameters which is set by the encoding unit 621 (or, the encoding unit 622) and decoded by the decoding unit 632 (or, the decoding unit 633).

25 [0356]

<7. Fifth embodiment>

[Configuration of computer]

The above series of processing may be executed by hardware, or may be executed by software. When the
30 series of processing is executed by software, programs constituting the software are installed to the computer.

Here, the computer includes a computer incorporated into dedicated hardware and, for example, a general-purpose personal computer capable of executing various kinds of functions by installing various kinds of programs.

5 [0357]

Fig. 27 is a block diagram illustrating an example of configuration of hardware of a computer executing the above series of processing using a program.

[0358]

10 In a computer 800, a CPU (Central Processing Unit) 801, a ROM (Read Only Memory) 802, and a RAM (Random Access Memory) 803 are connected with each other via a bus 804.

[0359]

15 The bus 804 is further connected with an input/output interface 805. The input/output interface 805 is connected with an input unit 806, an output unit 807, a storage unit 808, a communication unit 809, and a drive 810.

20 [0360]

The input unit 806 is constituted by a keyboard, a mouse, a microphone, and the like. The output unit 807 is constituted by a display, a speaker, and the like. The storage unit 808 is constituted by a hard disk, a nonvolatile memory, and the like. The communication unit 25 809 is constituted by a network interface and the like. The drive 810 drives a removable medium 811 such as a magnetic disk, an optical disk, a magneto-optical disk, and a semiconductor memory.

30 [0361]

In the computer configured as described above, the

CPU 801 performs the above series of processing by, e.g., executing the program stored in the storage unit 808 by loading the program to the RAM 803 via the input/output interface 805 and the bus 804.

5 [0362]

The program executed by the computer 800 (CPU 801) may be provided as being recorded to the removable medium 811 serving as, for example, a package medium. Further, the program can be provided via wired or wireless transmission media such as local area network, the Internet, and digital satellite broadcasting.

[0363]

In the computer, the program can be installed to the storage unit 808 via the input/output interface 805 by loading the removable medium 811 to the drive 810. Further, the program can be installed to the storage unit 808 by receiving the program with the communication unit 809 via wired or wireless transmission media. Also, the program can be installed to the ROM 802 and the storage unit 808 beforehand.

[0364]

The program executed by the computer may be a program with which processing is performed in time sequence according to the order explained in this specification, or may be a program with which processing is performed in parallel or with necessary timing, e.g., upon call.

[0365]

In this specification, steps describing the program recorded in the recording medium include processing performed in time sequence according to the described

order. The steps may not be necessarily performed in time sequence, and the steps include processing executed in parallel or individually.

[0366]

5 In this specification, the system includes the entire apparatus constituted by a plurality of devices.

[0367]

 A configuration explained as a device (or a processing unit) in the above explanation may be divided, and structured as multiple devices (or processing units).
10 A configuration explained as multiple devices (or processing units) in the above explanation may be combined, and structured as a device (or a processing unit). Alternatively, it is to be understood that the
15 configuration of each device (or each processing unit) may be added with any configuration other than the above. Further, when the configuration and operation of the entire system are substantially the same, a part of configuration of a certain device (or processing unit)
20 may be included in the configuration of another device (or another processing unit). More specifically, this technique is not limited to the above embodiment, and may be changed in various manners as long as it is within the gist of this technique.

25 [0368]

 The image coding device and image decoding device according to the embodiments explained above can be applied to various kinds of electronic devices such as a transmitter or a receiver for distribution to terminals
30 by satellite broadcasting, cable broadcasting such as cable television, distribution on the Internet, cellular

communication, recording devices for recording images to a medium such as an optical disk, magnetic disk, and flash memory, or a reproduction device for reproducing images from these recording media. Hereinafter, four
5 examples of applications will be explained.

[0369]

<8. Example of application>

[First example of application: television reception device]

10 Fig. 28 illustrates an example of schematic configuration illustrating a television device to which the above embodiments are applied. The television device 900 includes an antenna 901, a tuner 902, a demultiplexer 903, a decoder 904, a video signal processing unit 905, a
15 display unit 906, an audio signal processing unit 907, a speaker 908, an external interface 909, a control unit 910, a user interface 911, and a bus 912.

[0370]

 The tuner 902 extracts a signal of a desired
20 channel from a broadcasting signal received via the antenna 901, and demodulates the extracted signal. Then the tuner 902 outputs the encoded bit stream obtained from the demodulation to the demultiplexer 903. More specifically, the tuner 902 plays a role of a
25 transmission means in the television device 900 for receiving the coded stream in which an image is encoded.

[0371]

 The demultiplexer 903 separates a video stream and an audio stream of a program of viewing target from the
30 encoded bit stream, and outputs the separated streams to the decoder 904. Further, the demultiplexer 903 extracts

auxiliary data such as EPG (Electronic Program Guide) from the encoded bit stream, and provides the extracted data to the control unit 910. It should be noted that the demultiplexer 903 may perform descrambling in a case
5 where the encoded bit stream is scrambled.
[0372]

The decoder 904 decodes the video stream and the audio stream received from the demultiplexer 903. Then, decoder 904 outputs the video data generated from the
10 decoding processing to the video signal processing unit 905. The decoder 904 outputs the audio data generated from the decoding processing to the audio signal processing unit 907.
[0373]

15 The video signal processing unit 905 plays the video data received from the decoder 904, and causes the display unit 906 to display the video. The video signal processing unit 905 may display, on the display unit 906, an application screen provided via the network. The
20 video signal processing unit 905 may perform additional processing such as noise reduction on the video data in accordance with setting. Further, the video signal processing unit 905 generates an image of GUI (Graphical User Interface) such as menu, buttons, or cursor, and
25 overlays the generated image on the output image.
[0374]

The display unit 906 is driven by a driving signal provided from the video signal processing unit 905, and displays video or image on a video screen of a display
30 device (such as liquid crystal display, plasma display or OLED (Organic ElectroLuminescence Display) (organic EL

display) and the like).

[0375]

The audio signal processing unit 907 performs reproduction processing such as D/A conversion and amplification of audio data received from the decoder 904, and causes the speaker 908 to output audio. The audio signal processing unit 907 may perform additional processing such as noise reduction on the audio data.

[0376]

The external interface 909 is an interface for connection between the television device 900 and external device or network. For example, a video stream or an audio stream received via the external interface 909 may be decoded by the decoder 904. More specifically, the external interface 909 also plays a role of a transmission means in the television device 900 for receiving the coded stream in which an image is encoded.

[0377]

The control unit 910 has a memory such as a processor for a CPU and the like, and a RAM and a ROM. The memory stores, e.g., programs executed by the CPU, program data, EPG data, and data obtained via the network. The program stored in the memory may be, for example, read and executed by the CPU when the television device 900 is activated. The CPU executes the program to control operation of the television device 900 in accordance with operation signal received from the user interface 911, for example.

[0378]

The user interface 911 is connected to the control unit 910. The user interface 911 includes, e.g., buttons

and switches with which the user operates the television device 900, and a reception unit for receiving a remote control signal. The user interface 911 generates an operation signal by detecting user's operation via these constituent elements, and outputs the generated operation signal to the control unit 910.

[0379]

The bus 912 connects the tuner 902, the demultiplexer 903, the decoder 904, the video signal processing unit 905, the audio signal processing unit 907, the external interface 909, and the control unit 910 with each other.

[0380]

In the television device 900 configured as described above, the decoder 904 has a function of an image decoding device according to the embodiments explained above. Therefore, the amount of codes can be reduced when images are decoded by the television device 900.

[0381]

[Second example of application : cellular phone]

Fig. 29 illustrates an example of schematic configuration illustrating a cellular phone to which the above embodiments are applied. The cellular phone 920 includes an antenna 921, a communication unit 922, an audio codec 923, speaker 924, a microphone 925, a camera unit 926, an image processing unit 927, a demultiplexer 928, a recording/reproducing unit 929, a display unit 930, a control unit 931, an operation unit 932, and a bus 933.

[0382]

The antenna 921 is connected to the communication

unit 922. The speaker 924 and the microphone 925 are connected to the audio codec 923. The operation unit 932 is connected to the control unit 931. The bus 933 connects the communication unit 922, the audio codec 923, 5 the camera unit 926, the image processing unit 927, the demultiplexer 928, the recording/reproducing unit 929, the display unit 930, and the control unit 931 with each other.

[0383]

10 The cellular phone 920 performs operation such as transmission/reception of audio signals, transmission/reception of e-mails or image data, capturing images, and recording data in various kinds of modes including audio phone call mode, data communication 15 mode, shooting mode, and video call mode.

[0384]

 In the audio phone call mode, an analog audio signal generated by the microphone 925 is provided to the audio codec 923. The audio codec 923 converts an analog 20 audio signal into audio data, performs A/D conversion on the converted audio data, and compresses the audio data. Then, the audio codec 923 outputs the compressed audio data to the communication unit 922. The communication unit 922 encodes and modulates the audio data, and 25 generates a transmission signal. Then, the communication unit 922 transmits the generated transmission signal via the antenna 921 to the base station (not shown). The communication unit 922 amplifies a radio signal received via the antenna 921, and converts the frequency, and 30 obtains a reception signal. Then, the communication unit 922 generates audio data by demodulating and decoding a

reception signal, and outputs the generated audio data to the audio codec 923. The audio codec 923 decompresses the audio data, performs D/A conversion, and generates an analog audio signal. Then, the audio codec 923 provides
5 the generated audio signal to the speaker 924, and outputs audio.

[0385]

In the data communication mode, for example, the control unit 931 generates text data constituting an e-
10 mail in accordance given with user's operation with operation unit 932. The control unit 931 displays characters on the display unit 930. The control unit 931 generates e-mail data in accordance with user's transmission instruction given with the operation unit
15 932, and outputs the generated e-mail data to the communication unit 922. The communication unit 922 encodes and modulates e-mail data, and generates a transmission signal. Then, the communication unit 922 transmits the generated transmission signal via the
20 antenna 921 to the base station (not shown). The communication unit 922 amplifies a radio signal received via the antenna 921, and converts the frequency, and obtains a reception signal. Then, the communication unit 922 restores e-mail data by demodulating and decoding the
25 reception signal, and outputs the restored e-mail data to the control unit 931. The control unit 931 displays the contents of the e-mail on the display unit 930, and stores the e-mail data to the recording medium of the recording/reproducing unit 929.

30 [0386]

The recording/reproducing unit 929 has any given

recording medium that can be read and written. For example, the recording medium may be an internal recording medium such as a RAM or a flash memory, and may be an externally-attached recording medium such as a hard disk, a magnetic disk, a magneto-optical disk, an optical disk, a USB (Universal Serial Bus) memory, or a memory card.

[0387]

In the shooting mode, for example, the camera unit 926 captures an image of a subject, generates image data, and outputs the generated image data to the image processing unit 927. The image processing unit 927 encodes the image data, which are input from the camera unit 926, and stores the coded stream in the storage medium of the storing/reproducing unit 929.

[0388]

In the video call mode, for example, the demultiplexer 928 multiplexes the video stream encoded by the image processing unit 927 and the audio stream received from the audio codec 923, and outputs the multiplexed stream to the communication unit 922. The communication unit 922 encodes and modulates the stream, and generates a transmission signal. Then, the communication unit 922 transmits the generated transmission signal via the antenna 921 to the base station (not shown). The communication unit 922 amplifies a radio signal received via the antenna 921, and converts the frequency, and obtains a reception signal. The transmission signal and the reception signal may include the encoded bit stream. Then, the communication unit 922 restores the stream by

demodulating and decoding the reception signal, and
outputs the restored stream to the demultiplexer 928.
The demultiplexer 928 separates the video stream and the
audio stream from the received stream, and outputs the
5 video stream to the image processing unit 927 and the
audio stream to the audio codec 923. The image
processing unit 927 decodes the video stream, and
generates video data. The video data are provided to the
display unit 930, and the display unit 930 displays a
10 series of images. The audio codec 923 decompresses the
audio stream, performs D/A conversion, and generates an
analog audio signal. Then, the audio codec 923 provides
the generated audio signal to the speaker 924, and
outputs audio.

15 [0389]

In the cellular phone 920 configured as described
above, the image processing unit 927 has a function of
the image coding device and the image decoding device
according to the embodiments explained above.

20 Accordingly, the amount of codes can be reduced when
images are encoded and decoded by the cellular phone 920.
[0390]

[Third example of application:
recording/reproducing device]

25 Fig. 30 illustrates an example of schematic
configuration illustrating a recording/reproducing device
to which the above embodiments are applied. For example,
the recording/reproducing device 940 encodes the audio
data and the video data of received broadcasting program,
30 and records them to the recording medium. For example,
the recording/reproducing device 940 may encode the audio

data and the video data of obtained from another device, and may record them to the recording medium. For example, the recording/reproducing device 940 reproduces the data recorded on the recording medium using the monitor and
5 the speaker in accordance with user's instruction. At this occasion, the recording/reproducing device 940 decodes the audio data and the video data.

[0391]

The recording/reproducing device 940 includes a
10 tuner 941, an external interface 942, an encoder 943, an HDD (Hard Disk Drive) 944, a disk drive 945, a selector 946, a decoder 947, an OSD (On-Screen Display) 948, a control unit 949, and a user interface 950.

[0392]

15 The tuner 941 extracts a signal of a desired channel from a broadcasting signal received via an antenna (not shown), and demodulates the extracted signal. Then, the tuner 941 outputs the encoded bit stream obtained from the decoding to the selector 946. More
20 specifically, the tuner 941 plays a role of a transmission means in the recording/reproducing device 940.

[0393]

The external interface 942 is an interface for
25 connection between the recording/reproducing device 940 and external device or network. The external interface 942 may be, for example, an IEEE1394 interface, a network interface, a USB interface, a flash memory interface, or the like. For example, the video data and audio data
30 received via the external interface 942 are input into the encoder 943. More specifically, the external

interface 942 plays a role of a transmission means in the recording/reproducing device 940.

[0394]

When the video data and the audio data received
5 from the external interface 942 are not encoded, the encoder 943 encodes the video data and the audio data. Then, the encoder 943 outputs the encoded bit stream to the selector 946.

[0395]

10 The HDD 944 records the encoded bit stream obtained by compressing content data such as video and audio, various kinds of programs, and other data to the hard disk provided therein. When the video and audio are reproduced, the HDD 944 reads the data from the hard disk.

15 [0396]

The disk drive 945 records and reads data to/from the recording medium loaded. The recording medium loaded to the disk drive 945 may be, for example, a DVD disk (DVD-Video, DVD-RAM, DVD-R, DVD-RW, DVD+R, DVD+RW, and
20 the like) or Blu-ray (registered trademark) disk.

[0397]

When the video and the audio are recorded, the selector 946 selects the encoded bit stream which is input from the tuner 941 or the encoder 943, and outputs
25 the selected encoded bit stream to the HDD 944 or the disk drive 945. Further, when the video and the audio are reproduced, the selector 946 outputs the encoded bit stream which is input from the HDD 944 or the disk drive 945 to the decoder 947.

30 [0398]

The decoder 947 decodes the encoded bit stream, and

generates video data and audio data. Then, the decoder 947 outputs the generated video data to an OSD 948. The decoder 904 outputs the generated audio data to an external speaker.

5 [0399]

The OSD 948 reproduces the video data received from the decoder 947, and displays video. The OSD 948 may overlays images of GUI such as menu, buttons, or cursor, on the displayed video.

10 [0400]

The control unit 949 has a memory such as a processor for a CPU and the like, and a RAM and a ROM. The memory records programs executed by the CPU, program data, and the like. The program stored in the memory may be, for example, read and executed by the CPU when the recording/reproducing device 940 is activated. The CPU executes the program to control operation of the recording/reproducing device 940 in accordance with operation signal received from the user interface 971, for example.

20 [0401]

The user interface 950 is connected to the control unit 949. The user interface 950 includes, e.g., buttons and switches with which the user operates the recording/reproducing device 940, and a reception unit for receiving a remote control signal. The user interface 950 generates an operation signal by detecting user's operation via these constituent elements, and outputs the generated operation signal to the control unit 949.

30 [0402]

In the recording/reproducing device 940 configured as described above, the encoder 943 has a function of the image coding device according to the above embodiment. The decoder 947 has a function of an image decoding device according to the embodiments explained above. Accordingly, the amount of codes can be reduced when images are encoded and decoded by the recording/reproducing device 940.

[0403]

10 [Fourth example of application : image-capturing device]

Fig. 31 illustrates an example of schematic configuration illustrating an image-capturing device to which the above embodiments are applied. An image-capturing device 960 captures an image of a subject, generates image data, and records the image data to a recording medium.

[0404]

The image-capturing device 960 includes an optical block 961, an image-capturing unit 962, a signal processing unit 963, an image processing unit 964, a display unit 965, an external interface 966, a memory 967, a medium drive 968, an OSD 969, a control unit 970, a user interface 971, and a bus 972.

25 [0405]

The optical block 961 is connected to the image-capturing unit 962. The image-capturing unit 962 is connected to the signal processing unit 963. The display unit 965 is connected to the image processing unit 964. The user interface 971 is connected to the control unit 970. The bus 972 connects the image processing unit 964,

the external interface 966, the memory 967, the medium drive 968, the OSD 969, and the control unit 970 with each other.

[0406]

5 The optical block 961 includes a focus lens and a diaphragm mechanism. The optical block 961 causes an optical image of a subject to be formed on an image-capturing surface of the image-capturing unit 962. The image-capturing unit 962 includes an image sensor such as
10 a CCD (Charge Coupled Device) or a CMOS (Complementary Metal Oxide Semiconductor), and converts the optical image formed on the image-capturing surface into an image signal which is an electric signal by photoelectric conversion. Then, the image-capturing unit 962 outputs
15 the image signal to the signal processing unit 963.

[0407]

 The signal processing unit 963 performs various kinds of camera signal processing such as knee correction, gamma correction, and color correction on an image signal
20 received from the image-capturing unit 962. The signal processing unit 963 outputs the image data which have been subjected to the camera signal processing to the image processing unit 964.

[0408]

25 The image processing unit 964 encodes the image data received from the signal processing unit 963, and generates coded data. Then, the image processing unit 964 outputs the generated coded data to the external interface 966 or the medium drive 968. The image
30 processing unit 964 decodes the coded data received from the external interface 966 or the medium drive 968, and

generates image data. Then, the image processing unit 964 outputs the generated image data to the display unit 965. The image processing unit 964 may output the image data received from the signal processing unit 963 to the display unit 965, and may display the image thereon. The image processing unit 964 may also overlay display data obtained from the OSD 969 on the image which is to be output to the display unit 965.

[0409]

10 For example, the OSD 969 may generate images of GUI such as menu, buttons, or cursor, and output the generated image to the image processing unit 964.

[0410]

15 The external interface 966 is configured as, for example, a USB input/output terminal. The external interface 966 connects the image-capturing device 960 and a printer during printing of an image, for example. The external interface 966 is connected to a drive, as necessary. In the drive, for example, a removable medium 20 such as a magnetic disk or an optical disk may be loaded. A program which is read from the removable medium may be installed to the image-capturing device 960. Further, the external interface 966 may be configured as a network interface connected to a network such as a LAN or the 25 Internet. More specifically, the external interface 966 plays a role of a transmission means in the image-capturing device 960.

[0411]

30 The recording medium loaded to the medium drive 968 may be any given removable medium which can be read and written, such as a magnetic disk, an optical magnetic

disk, an optical disk, or a semiconductor memory. The recording medium loaded to the medium drive 968 in a fixed manner, and, for example, a non-removable storage unit such as an internal hard disk drive or SSD (Solid State Drive) may be configured.

[0412]

The control unit 970 has a memory such as a processor for a CPU and the like, and a RAM and a ROM. The memory records programs executed by the CPU, program data, and the like. The program stored in the memory may be, for example, read and executed by the CPU when the image-capturing device 960 is activated. The CPU executes the program to control operation of the image-capturing device 960 in accordance with operation signal received from the user interface 950, for example.

[0413]

The user interface 971 is connected to the control unit 970. The user interface 971 includes, e.g., buttons and switches with which the user operates the image-capturing device 960. The user interface 971 generates an operation signal by detecting user's operation via these constituent elements, and outputs the generated operation signal to the control unit 970.

[0414]

In the image-capturing device 960 configured as described above, the image processing unit 964 has a function of the image coding device and the image decoding device according to the embodiments explained above. Accordingly, the amount of codes can be reduced when images are encoded and decoded by the image-capturing device 960.

[0415]

In this specification, an example has been explained in which various kinds of information such as the parameters and the like of the sample adaptive offset filter are multiplexed into the coded stream and transmitted from the encoding side to the decoding side. However, the method for transmitting information is not limited to such example. For example, the information may not be multiplexed into the encoded bit stream, and may be transmitted or recorded as separate data associated with the encoded bit stream. In this case, the term "associated" means that the image included in the bit stream (which may be a part of image such as slice or block) and information corresponding to the image is linked during decoding. More specifically, the information may be transmitted through a transmission path which is separate from the image (or bit stream). The information may be recorded to another recording medium which is different from the image (or bit stream) (or another recording area of the same recording medium). Further, the information and the image (or bit stream) may be associated with each other in any given unit such as multiple frames, a frame, or a portion of a frame.

[0416]

The preferred embodiments of the present disclosure have been hereinabove described in detail with reference to attached drawings, but the present disclosure is not limited to such example. It is evident that a person of ordinarily skilled in the art to which the technique of the present disclosure pertains can conceive of various kinds of changes or modifications within the scope of the

technical gist described in the claims, and it is understood that various kinds of changes or modifications within the scope of the technical gist described in the claims are also included in the technical scope of the present disclosure.

[0417]

It should be noted that this technique can also be configured as follows.

(1) An image processing apparatus comprising:
a decoding unit configured to perform decoding processing on a coded stream to generate an image; and
a sample adaptive offset filter unit configured to apply sample adaptive offset filter to the image generated by the decoding unit in accordance with a type of a sample adaptive offset filter which is common to components of the image.

(2) The image processing apparatus according to (1), wherein the type of the sample adaptive offset filter is common to a brightness component and a color difference component of the image.

(3) The image processing apparatus according to (1), wherein the type of the sample adaptive offset filter is common to color difference components of the image.

(4) The image processing apparatus according to (3), wherein the type of the offset filter is common to a first color difference component and a second color difference component.

(5) The image processing apparatus according to (4), wherein when the type of the sample adaptive offset filter corresponding to the first color difference component is a band offset, the type of the sample

adaptive offset filter corresponding to the second color difference component is a band offset.

(6) The image processing apparatus according to (4), wherein when the type of the sample adaptive offset
5 filter corresponding to the first color difference component is an edge offset, the type of the sample adaptive offset filter corresponding to the second color difference component is an edge offset.

(7) The image processing apparatus according to (4),
10 wherein when the type of the sample adaptive offset filter is an edge offset, a pattern rule of the edge offset is common between color difference components of the image.

(8) The image processing apparatus according to (7),
15 wherein when the type of the offset filter corresponding to the first color difference component is one-dimensional pattern of an edge offset, the type of the offset filter corresponding to the second color difference component is one-dimensional pattern of the
20 edge offset.

(9) The image processing apparatus according to (7), wherein when the type of the offset filter corresponding to the first color difference component is two-dimensional pattern of the edge offset, the type of the
25 offset filter corresponding to the second color difference component is two-dimensional pattern of the edge offset.

(10) The image processing apparatus according to (4), wherein when the type of the offset filter
30 corresponding to the first color difference component is a type for not applying an offset, the type of the offset

filter corresponding to the second color difference component is a type for not applying an offset.

(11) The image processing apparatus according to any of (1) to (10), wherein a color space of the image is
5 in a Y/Cb/Cr format.

(12) The image processing apparatus according to any of (1) to (11) further comprising a reception unit configured to receive the coded stream and type information indicating the type of the sample adaptive
10 offset filter common between components of the image,
wherein the decoding unit uses the type information received by the reception unit to decode the coded stream received by the reception unit.

(13) The image processing apparatus according to
15 any of (1) to (12) further comprising a deblock filter unit configured to apply deblock filter to the image generated by the decoding unit, wherein the sample adaptive offset filter unit applies sample adaptive offset filter to the image to which the deblock filter is
20 applied by the deblock filter unit.

(14) The image processing apparatus according to any of (1) to (13), wherein the reception unit receives a merge flag indicating a same offset as an adjacent coding unit adjacent to a current coding unit, and
25 the sample adaptive offset filter unit uses the merge flag received by the reception unit to apply sample adaptive offset filter to the image generated by the decoding unit.

(15) An image processing method comprising:
30 performing decoding processing on a coded stream to generate an image; and

applying sample adaptive offset filter to the generated image in accordance with a type of a sample adaptive offset filter which is common to components of the image.

5

REFERENCE SIGNS LIST

[0418]

	11	Image coding device
	26	Lossless coding unit
10	31	Deblock filter
	41	Sample adaptive offset filter
	42	Adaptive loop filter
	51	Image decoding device
	62	Lossless decoding unit
15	66	Deblock filter
	81	Sample adaptive offset filter
	82	Adaptive loop filter
	111	SAO flag
	112	Type setting unit
20	113	Offset setting unit
	114	SAO control information setting unit
	115	Offset buffer
	116	Filter processing unit
	117	Image buffer
25	141	Type setting unit
	151	Y type setting unit
	152	Cm type setting unit
	161	Syntax write unit
	211	Syntax acquisition unit
30	221	SAO flag reception unit
	222	Type reception unit

	223	Merge flag reception unit
	224	Offset buffer
	225	Offset reception unit
	226	Filter processing unit
5	227	Image buffer
	241	Syntax acquisition unit
	251	Type reception unit
	261	Y type reception unit
	262	Cm type reception unit
10		

CLAIMS

1. An image processing apparatus comprising:
a decoding unit configured to perform decoding
5 processing on a coded stream to generate an image; and
a sample adaptive offset filter unit configured to
apply sample adaptive offset filter to the image
generated by the decoding unit in accordance with a type
of a sample adaptive offset filter which is common to
10 components of the image.
2. The image processing apparatus according to claim 1,
wherein the type of the sample adaptive offset filter is
common to a brightness component and a color difference
15 component of the image.
3. The image processing apparatus according to claim 1,
wherein the type of the sample adaptive offset filter is
common to color difference components of the image.
20
4. The image processing apparatus according to claim 3,
wherein the type of the offset filter is common to a
first color difference component and a second color
difference component.
25
5. The image processing apparatus according to claim 4,
wherein when the type of the sample adaptive offset
filter corresponding to the first color difference
component is a band offset, the type of the sample
30 adaptive offset filter corresponding to the second color
difference component is a band offset.

6. The image processing apparatus according to claim 4,
wherein when the type of the sample adaptive offset
filter corresponding to the first color difference
5 component is an edge offset, the type of the sample
adaptive offset filter corresponding to the second color
difference component is an edge offset.

7. The image processing apparatus according to claim 4,
10 wherein when the type of the sample adaptive offset
filter is an edge offset, a pattern rule of the edge
offset is common between color difference components of
the image.

15 8. The image processing apparatus according to claim 7,
wherein when the type of the offset filter corresponding
to the first color difference component is one-
dimensional pattern of an edge offset, the type of the
offset filter corresponding to the second color
20 difference component is one-dimensional pattern of the
edge offset.

9. The image processing apparatus according to claim 7,
wherein when the type of the offset filter corresponding
25 to the first color difference component is two-
dimensional pattern of the edge offset, the type of the
offset filter corresponding to the second color
difference component is two-dimensional pattern of the
edge offset.

30

10. The image processing apparatus according to claim 4,

wherein when the type of the offset filter corresponding to the first color difference component is a type for not applying an offset, the type of the offset filter corresponding to the second color difference component is
5 a type for not applying an offset.

11. The image processing apparatus according to claim 1, wherein a color space of the image is in a Y/Cb/Cr format.

10 12. The image processing apparatus according to claim 1 further comprising a reception unit configured to receive the coded stream and type information indicating the type of the sample adaptive offset filter common between components of the image,

15 wherein the decoding unit uses the type information received by the reception unit to decode the coded stream received by the reception unit.

13. The image processing apparatus according to claim 1
20 further comprising a deblock filter unit configured to apply deblock filter to the image generated by the decoding unit,

wherein the sample adaptive offset filter unit applies sample adaptive offset filter to the image to
25 which the deblock filter is applied by the deblock filter unit.

14. The image processing apparatus according to claim 1, wherein the reception unit receives a merge flag
30 indicating a same offset as an adjacent coding unit adjacent to a current coding unit, and

the sample adaptive offset filter unit uses the merge flag received by the reception unit to apply sample adaptive offset filter to the image generated by the decoding unit.

5

15. An image processing method comprising:

performing decoding processing on a coded stream to generate an image; and

10 applying sample adaptive offset filter to the generated image in accordance with a type of a sample adaptive offset filter which is common to components of the image.