



- (51) **International Patent Classification:**  
*H04N 7/32* (2006.01) *H04N 7/30* (2006.01)
- (21) **International Application Number:**  
PCT/JP2012/004471
- (22) **International Filing Date:**  
11 July 2012 (11.07.2012)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**  
2011-155012 13 July 2011 (13.07.2011) JP  
2011-243937 7 November 2011 (07.11.2011) JP
- (71) **Applicant (for all designated States except US):** CANON KABUSHIKI KAISHA [JP/JP]; 30-2, Shimomaruko 3-chome, Ohta-ku, Tokyo, 1468501 (JP).
- (72) **Inventor; and**
- (75) **Inventor/Applicant (for US only):** SHIMA, Masato [JP/JP]; C/O CANON KABUSHIKI KAISHA, 30-2, Shimomaruko 3-chome, Ohta-ku, Tokyo, 1468501 (JP).
- (74) **Agents:** ABE, Takuma et al.; C/O CANON KABUSHIKI KAISHA, 30-2, Shimomaruko 3-chome, Ohta-ku, Tokyo, 1468501 (JP).

(81) **Designated States** (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

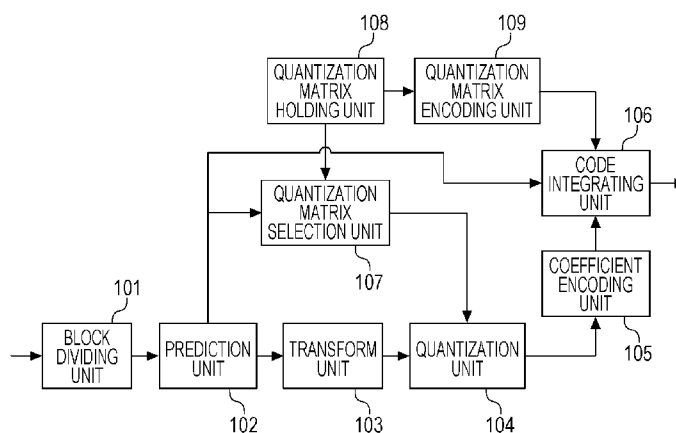
(84) **Designated States** (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Published:**

— with international search report (Art. 21(3))

(54) **Title:** IMAGE ENCODING APPARATUS, IMAGE ENCODING METHOD, IMAGE DECODING APPARATUS, IMAGE DECODING METHOD, AND PROGRAM

[Fig. 1]



(57) **Abstract:** An image encoding apparatus includes a block dividing unit that divides an input image into a plurality of blocks, a prediction unit that performs prediction on each of the divided blocks using encoded blocks and generates a prediction error, a transform unit that determines a shape of a processing block from a plurality of square or rectangular shapes of processing blocks for orthogonal transform, applies orthogonal transform of the determined shape to the prediction error generated by the prediction unit, and generates a transformed coefficient, a quantization matrix selection unit that selects a quantization matrix based on the shape of a processing block for orthogonal transform determined by the transform unit, a quantization unit that quantizes the transformed coefficient using the quantization matrix selected by the quantization matrix selection unit and generates a quantized coefficient, and a coefficient encoding unit that encodes the quantized coefficient generated by the quantization unit.



## Description

### Title of Invention: IMAGE ENCODING APPARATUS, IMAGE ENCODING METHOD, IMAGE DECODING APPARATUS, IMAGE DECODING METHOD, AND PROGRAM

#### Technical Field

[0001] The present invention relates to an image encoding apparatus, an image encoding method, an image decoding apparatus, an image decoding method, and a program and, in particular, to a quantization method and methods for encoding and decoding a quantization matrix.

#### Background Art

[0002] H.264/MPEG-4 AVC (hereinafter referred to as "H.264") has been used for encoding a moving image (refer to NPL 1). In H.264, each element of a quantization matrix can be changed to a desired value by encoding scaling\_list information. In H.264, by adding delta\_scale representing a difference value to the previous element, each of the elements of a quantization matrix can have a desired value.

[0003] In recent years, the activity for internationally standardizing a highly efficient encoding method, which is a successor of H.264, has been started. JCT-VC (Joint Collaborative Team on Video Coding) has been established by ISO/IEC and ITU-T. JCT-VC standardizes a coding method known as HEVC (High Efficiency Video Coding) method (hereinafter referred to as "HEVC").

[0004] Since a target display size has been increased, the display size is divided into blocks each having a block size that is larger than the existing macroblock size (16 \* 16 pixels). Such a basic block having the large size is referred to as a "Largest Coding Unit (LCU)". The largest size of the LCU is 64 \* 64 pixels (refer to, for example, NPL 2). An LCU is further divided into sub-blocks, each to be subjected to a prediction process and a transform process.

[0005] In order to increase the efficiency of encoding, an intra prediction/orthogonal transform coding scheme based on a rectangular sub-block (e.g., a short distance intra prediction scheme) has been developed in addition to the existing intra prediction/orthogonal transform coding scheme based on a square sub-block (refer to NPL 3). Fig. 2 illustrates an example of sub-block division used in the short distance intra prediction scheme. As illustrated in Fig. 2, for simplicity, a basic block 200 is a 16 \* 16 pixel block, and each quadrangle indicated by a bold line represents a sub-block. In Fig. 2, the basic block 200 illustrates an existing square sub-block dividing scheme. The 16 \* 16 pixel basic block is divided into 8 \* 8 pixel or 4 \* 4 square sub-blocks. In contrast, in Fig. 2, basic blocks 201 and 202 illustrate an example of a rectangle sub-block

dividing scheme using a short distance intra prediction scheme. The  $16 * 16$  basic block is divided into  $4 * 16$  or  $16 * 4$  rectangular sub-blocks. In this way, a coding process is performed using a rectangular sub-block in addition to a square sub-block.

- [0006] In HEVC, like H.264, a scheme for setting each element of a quantization matrix to a desired value has been developed. In addition, in HEVC, in addition to the square sub-block division scheme, a rectangular sub-block division scheme has been developed. At that time, distribution of coefficients resulted from orthogonal transform varies in accordance with the shape for the orthogonal transform. Accordingly, if the shape for orthogonal transform does not match a quantization matrix, quantization is not appropriately performed. As a result, the image quality is significantly degraded.

## Citation List

### Non Patent Literature

- [0007] NPL 1: ITU-T H.264 (03/2010) Advanced video coding for generic audio visual services  
 NPL 2: JCT-VC Contribution JCTVC-E603.doc available at [http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/5\\_Geneva/wg11/](http://phenix.int-evry.fr/jct/doc_end_user/documents/5_Geneva/wg11/)  
 NPL 3: JCT-VC Contribution JCTVC-E278.doc available at [http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/5\\_Geneva/wg11/](http://phenix.int-evry.fr/jct/doc_end_user/documents/5_Geneva/wg11/)

### Summary of Invention

- [0008] The present invention provides an image encoding apparatus, an image encoding method, an image decoding apparatus, an image decoding method, and a program that adaptively use a quantization matrix in accordance with a shape for orthogonal transform. In addition, the present invention provides an image encoding apparatus, an image encoding method, an image decoding apparatus, an image decoding method, and a program that reduce the amount of code when a quantization matrix is set for each of the shapes for orthogonal transform.
- [0009] According to an embodiment of the present invention, an image encoding apparatus includes a block dividing unit configured to divide an input image into a plurality of blocks, a prediction unit configured to perform prediction on each of the blocks divided by the block dividing unit using already encoded blocks and generate a prediction error, a transform unit configured to determine a shape of a processing block to be used from a plurality of square or rectangular shapes of processing blocks for orthogonal transform, apply orthogonal transform of the determined shape to the prediction error generated by the prediction unit, and generate a transformed coefficient, a quantization matrix selection unit configured to select a quantization matrix on the basis of the shape of a processing block for orthogonal transform determined by the transform unit, a quantization unit configured to quantize the transformed co-

efficient using the quantization matrix selected by the quantization matrix selection unit and generate a quantized coefficient, and a coefficient encoding unit configured to encode the quantized coefficient generated by the quantization unit.

[0010] According to the present invention, a quantization matrix can be set for each of the square or rectangular shapes of the processing block for orthogonal transform and, thus, a quantization process best suited for each of the shapes of a processing block can be performed.

### **Brief Description of Drawings**

[0011] [fig.1]Fig. 1 is a block diagram of the configuration of an image encoding apparatus according to a first exemplary embodiment of the present invention.

[fig.2]Fig. 2 illustrates an example of sub-block division.

[fig.3]Fig. 3 is a block diagram of the configuration of an image decoding apparatus according to a second exemplary embodiment of the present invention.

[fig.4]Fig. 4 illustrates an example of a quantization matrix.

[fig.5]Fig. 5 is a flowchart of an image encoding process performed by the image encoding apparatus according to the first exemplary embodiment.

[fig.6]Fig. 6 is a flowchart of an image decoding process performed by the image decoding apparatus according to the second exemplary embodiment.

[fig.7A]Fig. 7A illustrates an example of the structure of a bit stream.

[fig.7B]Fig. 7B illustrates an example of the structure of a bit stream.

[fig.7C]Fig. 7C illustrates an example of the structure of a bit stream.

[fig.8]Fig. 8 is a block diagram of the configuration of an image encoding apparatus according to a third exemplary embodiment of the present invention.

[fig.9]Fig. 9 is a block diagram of the configuration of an image decoding apparatus according to a fourth exemplary embodiment of the present invention.

[fig.10]Fig. 10 is a flowchart of an image encoding process performed by the image encoding apparatus according to the third exemplary embodiment.

[fig.11]Fig. 11 is a flowchart of an image decoding process performed by the image decoding apparatus according to the fourth exemplary embodiment.

[fig.12]Fig. 12 is a block diagram of exemplary computer hardware applicable to the image encoding apparatus and the image decoding apparatus according to the exemplary embodiments.

[fig.13]Fig. 13 is a block diagram of an image encoding apparatus according to a fifth exemplary embodiment of the present invention.

[fig.14]Fig. 14 is a block diagram of an image decoding apparatus according to a sixth exemplary embodiment of the present invention.

[fig.15]Fig. 15 is a flowchart of an exemplary image encoding process performed by

the image encoding apparatus according to the fifth exemplary embodiment.

[fig.16]Fig. 16 is a flowchart of an exemplary image decoding process performed by the image decoding apparatus according to the sixth exemplary embodiment.

[fig.17A]Fig. 17A illustrates an example of computation for obtaining a rectangular quantization matrix from a square quantization matrix.

[fig.17B]Fig. 17B illustrates an example of computation for obtaining a rectangular quantization matrix from a square quantization matrix.

[fig.17C]Fig. 17C illustrates an example of computation for obtaining a rectangular quantization matrix from a square quantization matrix.

## Description of Embodiments

[0012]

### First Exemplary Embodiment

[0013] Exemplary embodiments of the present invention are described below with reference to the accompanying drawings. Fig. 1 is a block diagram of the configuration of an image encoding apparatus according to a first exemplary embodiment of the present invention.

[0014] In Fig. 1, a block dividing unit 101 divides an input image into a plurality of basic blocks. According to the present exemplary embodiment, for simplicity, the block dividing unit 101 divides an input image into  $16 * 16$  pixel basic blocks. However, the size of the basic block is not limited thereto.

[0015] A prediction unit 102 divides each of the basic blocks obtained through a dividing operation performed by the block dividing unit 101 into sub-blocks each having a size smaller than or equal to the size of the basic block. Thereafter, the prediction unit 102 performs prediction using encoded blocks on a sub-block basis. In addition, the prediction unit 102 selects a prediction method. Thereafter, the prediction unit 102 performs prediction and computes a difference value in accordance with the selected prediction method and, thus, computes a prediction error. In the case of an intra frame of a still image or a moving image, intra prediction is performed. In addition, in the case of a moving image, motion compensation prediction is performed. In general, in order to perform intra prediction, among a plurality of reference methods, a method (a prediction method) for referencing reference pixels used for computing a prediction value using data of the reference pixels surrounding the target pixel is selected. According to the present exemplary embodiment, for simplicity, the prediction unit 102 performs prediction by dividing the basic block into  $16 * 16$  pixel,  $8 * 8$  pixel, and  $4 * 4$  pixel square sub-blocks. Alternatively, the prediction unit 102 performs prediction by dividing the basic block into  $16 * 4$  pixel and  $4 * 16$  pixel rectangular sub-blocks. However, the size of the sub-block is not limited thereto.

- [0016] A quantization matrix holding unit 108 generates quantization matrices and temporarily holds the generated quantization matrices. Any method for generating quantization matrices to be held can be used. For example, a user may input the quantization matrices. Alternatively, the quantization matrices may be computed using the characteristics of an input image, or quantization matrices predetermined as an initial value may be used.
- [0017] A transform unit 103 applies orthogonal transform to the prediction error of each of the sub-blocks. According to the present exemplary embodiment, the transform unit 103 applies orthogonal transform on the basis of the size of the sub-block determined by the prediction unit 102 and computes an orthogonal transformed coefficient. According to the present exemplary embodiment, the orthogonal transform is applied on  $16 * 16$  pixel,  $8 * 8$  pixel, and  $4 * 4$  pixel processing blocks. Alternatively, the orthogonal transform is applied on  $16 * 4$  pixel and  $4 * 16$  pixel processing blocks. As the orthogonal transform, discrete sine transform, Karhunen-Loeve transform, or Hadamard transform can be employed in addition to discrete cosine transform.
- [0018] A quantization matrix selection unit 107 selects a quantization matrix to be applied from among the quantization matrices stored in the quantization matrix holding unit 108 by inputting the sub-block size selected by the prediction unit 102.
- [0019] A quantization unit 104 quantizes the orthogonal transformed coefficient using the quantization matrix selected by the quantization matrix selection unit 107. Through such quantization, a quantized coefficient can be obtained.
- [0020] A coefficient encoding unit 105 encodes the quantized coefficient obtained in this manner and generates quantized coefficient code data. At that time, any technique for encoding the quantized coefficient can be employed. For example, Huffman coding or arithmetic coding can be employed.
- [0021] A quantization matrix encoding unit 109 encodes the quantization matrices stored in the quantization matrix holding unit 108 and generated quantization matrix code data. A technique for encoding the quantization matrices is described in more detail below. Any encoding technique can be employed. For example, Huffman coding or arithmetic coding can be directly applied to the value of each of the elements of the quantization matrix. Alternatively, Huffman coding or arithmetic coding may be applied to a difference between the value of the element and the value of the immediately previous element or a difference between the value of the element and the value of an element of a different quantization matrix.
- [0022] A code integrating unit 106 generates code related to header information, prediction, and transform. In addition, the code integrating unit 106 integrates the quantized coefficient code data generated by the coefficient encoding unit 105 with the quantization matrix code data generated by the quantization matrix encoding unit 109. Examples of

code related to prediction and transform include code of the sub-block size and a prediction method selected by the prediction unit 102.

- [0023] An exemplary operation for encoding an image performed by the above-described image encoding apparatus is described below. According to the present exemplary embodiment, moving image data is input on a frame basis. However, a configuration that receives still image data for one frame may be employed. In addition, according to the present exemplary embodiment, for simplicity, only an intra prediction encoding process is described. However, the following operation can be applied to an inter prediction coding process.
- [0024] Image data for one frame is input to the block dividing unit 101. The image data is divided into  $16 * 16$  pixel basic blocks. Thereafter, each of the basic blocks is input to the prediction unit 102.
- [0025] The prediction unit 102 determines the size of a sub-block used for prediction. The prediction unit 102 further determines a prediction method. Thereafter, the prediction unit 102 performs prediction on a sub-block basis and generates a prediction error.
- [0026] The transform unit 103 applies orthogonal transform on the prediction error generated by the prediction unit 102 for each of the sub-blocks and generates the orthogonal transformed coefficient. According to the present exemplary embodiment, the sub-block size and the shape are the same as the orthogonal transform size and shape. However, they are not necessarily the same. The generated orthogonal transformed coefficient is input to the quantization unit 104.
- [0027] The quantization matrix holding unit 108 stores the quantization matrices used for quantization of the frame, that is, the quantization matrices corresponding to the shapes of the sub-blocks. According to the present exemplary embodiment, five sub-block sizes, that is, the sizes of  $4 * 4$  pixels,  $8 * 8$  pixels,  $16 * 16$  pixels,  $16 * 4$  pixels, and  $4 * 16$  pixels are employed. Accordingly, the quantization matrix holding unit 108 stores a first quantization matrix for  $4 * 4$  pixel orthogonal transform, a second quantization matrix for  $8 * 8$  pixel orthogonal transform, and a third quantization matrix for  $16 * 16$  pixel orthogonal transform. In addition, the quantization matrix holding unit 108 stores a fourth quantization matrix for  $16 * 4$  pixel orthogonal transform, and a fifth quantization matrix for  $4 * 16$  pixel orthogonal transform. According to the present exemplary embodiment, the quantization matrix holding unit 108 stores the above-described five quantization matrices. However, the number of the stored quantization matrices varies with the number of the shapes of the sub-blocks.
- [0028] The quantization matrix encoding unit 109 sequentially reads the quantization matrices from the quantization matrix holding unit 108 in the order of the sub-block size and encodes each of the quantization matrices to generate the quantization matrix code data. At that time, according to the present exemplary embodiment, each of the

elements of the quantization matrix is directly encoded, or a difference between an element and an element immediately before the element is encoded. Alternatively, the quantization matrix is encoded using DPCM, or the difference between the quantization matrix and a different quantization matrix is encoded. However, the encoding technique is not limited thereto. At the same time, by inserting code that stops encoding when the same value continuously appears, the amount of code of the quantization matrix can be reduced.

[0029] A particular example of a method for encoding a quantization matrix is described next with reference to Fig. 4. In Fig. 4, a quantization matrix 400 corresponds to 16 pixels of a 4 \* 4 pixel sub-block. Each square indicated by a bold line represents an element. In this example, the method for encoding five quantization matrices (i.e., the quantization matrix 400 and quantization matrices 401 to 404 in Fig. 4) is described.

[0030] As illustrated in Fig. 4, the elements of the quantization matrix 400 start with a value of 1, and are incremented by one from element to element in the element sequence. For example, to encode the quantization matrix 400, values from 1 to 16 are encoded one by one. Alternatively, after the first element "1" is encoded, the difference between the value of an element and the value of the immediately previous element (i.e., "1" in this example) may be encoded 15 times. Furthermore, like DPCM, after the first element "1" and the difference of "1" between the first element and the second element are encoded, a prediction difference of "0" may be encoded.

[0031] As illustrated in Fig. 4, the elements of the quantization matrix 401 start with a value of 3 and are incremented by one from element to element in the element sequence. For example, to encode the quantization matrix 401, values from 3 to 18 are encoded one by one. Alternatively, after the first element "3" is encoded, the difference between the value of an element and the value of the immediately previous element (i.e., "1" in this example) may be encoded 15 times. Furthermore, the difference between the element of the quantization matrix 400 and the element of the quantization matrix 401 (i.e., "2" in this example) may be encoded 16 times.

[0032] The elements of the quantization matrix 402 start with a value of "3" and are incremented by one from element to element in the element sequence until the value reaches "10". Thereafter, the value "10" appears three times and, subsequently, the value is incremented by one from element to element until the value reaches "16". For example, to encode the quantization matrix 402, like the quantization matrices 400 and 401, the value of each of the elements is encoded one by one. Alternatively, the difference between the value of the element and the value of the immediately previous element or the difference between the value of the element and the value of the element of a different quantization matrix may be encoded. If the difference between the value of each of the elements of the quantization matrix 402 and the value of each

of the elements of the quantization matrix 400 is encoded, the difference value "2" is encoded for the first "3" to the first "10", and the difference "1" is encoded for the second "10". For the third "10" and the subsequent values, the difference value is "0". Therefore, the difference value "0" is encoded seven times. Alternatively, at the same time, by encoding code that stops the encoding operation, the amount of code of the difference values may be reduced.

- [0033] The elements of the quantization matrix 403 start with a value of "3" and are incremented by one from element to element in the element sequence until the value reaches "10". Thereafter, the value "10" repeatedly appears to the end. For example, to encode the quantization matrix 403, like the quantization matrices 400 and 401, the value of each of the elements is encoded one by one. Alternatively, the difference between the value of the element and the value of the immediately previous element or the difference between the value of the element and the value of the element of a different quantization matrix may be encoded. If the difference between the value of each of the elements of the quantization matrix 403 and the value of the immediately previous element is encoded, the difference value "1" is encoded for the first "3" to the first "10". For the subsequent elements, the difference value is "0". Therefore, the difference value "0" is encoded seven times. In addition, at the same time, by encoding code that stops the encoding operation, the amount of code of the difference values may be reduced.
- [0034] The elements of the quantization matrix 404 start with a value of "6" and are gradually increased. For example, to encode the quantization matrix 404, like the quantization matrices 400 and 401, the value of each of the elements is encoded one by one. Alternatively, the difference between the value of the element and the value of the immediately previous element or the difference between the value of the element and the value of the element of a different quantization matrix may be encoded. If the difference between the value of each of the elements of the quantization matrix 404 and the value of the immediately previous element is encoded, the value "6" of the first element is encoded first. Thereafter, the difference value between the element and the immediately previous element is encoded. That is, the values "7", "0", "7", "0", "0", "8", "0", "0", "0", "4", "0", "0", "5", "0", and "5" are encoded in this order. It will be appreciated that the above-described encoding methods of a quantization matrix is only for the purpose of illustration and shall not be considered limiting in any way.
- [0035] The quantization matrix selection unit 107 selects one of the quantization matrices used for quantization of the target sub-block using the sub-block size determined by the prediction unit 102. More specifically, the quantization matrix selection unit 107 selects, among the quantization matrices stored in the quantization matrix holding unit 108, the quantization matrix used for quantization of the target sub-block. For

example, according to the present exemplary embodiment, if the sub-block size is 16 \* 4 pixels, the quantization matrix selection unit 107 retrieves the fourth quantization matrix from the quantization matrix holding unit 108 and outputs the fourth quantization matrix to the quantization unit 104.

[0036] The quantization unit 104 quantizes the orthogonal transformed coefficient output from the transform unit 103 using the quantization matrix selected by the quantization matrix selection unit 107 and generates a quantized coefficient. The generated quantized coefficient is input to the coefficient encoding unit 105.

[0037] The coefficient encoding unit 105 encodes the quantized coefficient generated by the quantization unit 104 and generates quantized coefficient code data. Thereafter, the coefficient encoding unit 105 outputs the generated quantized coefficient code data to the code integrating unit 106.

[0038] In addition, the quantization matrix encoding unit 109 encodes each of the quantization matrices stored in the quantization matrix holding unit 108 and generates quantization matrix code data. Thereafter, the quantization matrix encoding unit 109 outputs the generated quantization matrix code data to the code integrating unit 106.

[0039] The code integrating unit 106 generates code, such as code for a sequence header, a frame header, a picture header, and a slice header of the image. In addition, the code integrating unit 106 inserts the quantization matrix code data generated by the quantization matrix encoding unit 109 into any one of the headers. The code integrating unit 106 integrates the code of such a header portion with the quantized coefficient code data generated by the coefficient encoding unit 105. In addition, the code integrating unit 106 encodes the sub-block size and prediction information generated by the prediction unit 102. Thereafter, the code integrating unit 106 generates a bit stream and outputs the generated bit stream.

[0040] Fig. 7A illustrates an example of a bit stream output in the first exemplary embodiment. According to the present exemplary embodiment, the quantization matrix is encoded into the sequence header portion of the bit stream. However, the quantization matrix may be encoded into any position. For example, the quantization matrix may be encoded into a picture header portion or another header portion. In addition, if a quantization matrix is modified in a sequence, the quantization matrix can be modified by newly encoding the quantization matrix. At that time, all of the quantization matrices may be rewritten or some of the quantization matrices may be modified by specifying the sub-block size of the quantization matrix to be rewritten. Furthermore, the quantization matrix can be partially modified by specifying the sub-block size and the position of an element to be modified of the quantization matrix to be rewritten.

[0041] Fig. 5 is a flowchart of an image encoding process performed by the image encoding apparatus according to the first exemplary embodiment. In step S501, the quantization

matrix holding unit 108 generates at least one quantization matrix. According to the present exemplary embodiment, five sub-block sizes are employed. Accordingly, as described above, five quantization matrices, that is, first to fifth quantization matrices are generated.

[0042] In step S502, the quantization matrix encoding unit 109 encodes each of the quantization matrices generated by the quantization matrix holding unit 108.

[0043] In step S503, the code integrating unit 106 encodes the header portion of a bit stream and outputs the encoded header portion. At that time, the quantization matrix code data encoded in step S502 is output together with the header portion.

[0044] In step S504, the block dividing unit 101 divides a frame-based input image into basic blocks (16 \* 16 pixels).

[0045] In step S505, the prediction unit 102 determines the size of the sub-block. In step S506, the prediction unit 102 performs prediction on a sub-block size basis generated in step S505.

[0046] In step S507, the image encoding apparatus selects a process to be performed next on the basis of the sub-block size determined in step S506. If the sub-block size is 4 \* 4 pixels, the processing proceeds to step S508. However, if the sub-block size is 8 \* 8 pixels, the processing proceeds to step S528. If the sub-block size is 16 \* 16 pixels, the processing proceeds to step S548. If the sub-block size is 16 \* 4 pixels, the processing proceeds to step S568. If the sub-block size is 4 \* 16 pixels, the processing proceeds to step S588.

[0047] In step S508, the transform unit 103 applies orthogonal transform to the prediction error generated in step S505 on a 4 \* 4 pixel sub-block basis and generates an orthogonal transformed coefficient.

[0048] In step S509, the quantization matrix selection unit 107 selects the first quantization matrix from among the quantization matrices generated in step S501.

[0049] In addition, in step S528, the transform unit 103 applies orthogonal transform to the prediction error generated in step S505 on an 8 \* 8 pixel sub-block basis and generates an orthogonal transformed coefficient. As in step S509, in step S529, the quantization matrix selection unit 107 selects the second quantization matrix from among the quantization matrices generated in step S501. In addition, in step S548, the transform unit 103 applies orthogonal transform to the prediction error generated in step S505 on a 16 \* 16 pixel sub-block basis and generates an orthogonal transformed coefficient. As in step S509, in step S549, the quantization matrix selection unit 107 selects the third quantization matrix from among the quantization matrices generated in step S501. In addition, in step S568, the transform unit 103 applies orthogonal transform to the prediction error generated in step S505 on a 16 \* 4 pixel sub-block basis and generates an orthogonal transformed coefficient. As in step S509, in step S569, the quantization

matrix selection unit 107 selects the fourth quantization matrix from among the quantization matrices generated in step S501. In addition, in step S588, the transform unit 103 applies orthogonal transform to the prediction error generated in step S505 on a 4 \* 16 pixel sub-block basis and generates an orthogonal transformed coefficient. As in step S509, in step S589, the quantization matrix selection unit 107 selects the fifth quantization matrix from among the quantization matrices generated in step S501.

[0050] In step S510, the quantization unit 104 receives the orthogonal transformed coefficient generated in step S508, S528, S548, S568, or S588. Thereafter, the quantization unit 104 quantizes the orthogonal transformed coefficients using the quantization matrices selected in steps S509, S529, S549, S569, or S589 and generates the quantized coefficient.

[0051] In step S511, the coefficient encoding unit 105 encodes the quantized coefficient generated in step S510 and generates the quantized coefficient code data.

[0052] In step S512, the image encoding apparatus determines whether all of the sub-blocks in the basic block have been encoded. If all of the sub-blocks in the basic block have been encoded, the processing proceeds to step S513. However, if all of the sub-blocks in the basic block have not been encoded, the processing returns to step S505 in order to process the next sub-block.

[0053] In step S513, the image encoding apparatus determines whether all of the basic blocks have been encoded. If all of the basic blocks have been encoded, all operations are stopped and the processing is completed. However, if all of the basic blocks have not been encoded, the processing returns to step S504, where processing of the next basic block is started.

[0054] Through such a configuration and operations, in particular, through processing in steps S501, S509, S529, S549, S569, and S589, a bit stream that is quantized using the quantization matrix best suited for the shape of a sub-block for orthogonal transform can be generated.

[0055] While the present exemplary embodiment has been described with reference to a frame to which only intra prediction is applied, it is appreciated that the present exemplary embodiment is applicable to a frame to which inter prediction can be applied.

[0056] In addition, while the present exemplary embodiment has been described with reference to a technique in which one of quantization matrices is used for one of the sub-block sizes, a plurality of the quantization matrices may be prepared for one of the sub-block sizes and, thereafter, one of the quantization matrices may be selected and used in accordance with a prediction method or a transform method.

[0057] Alternatively, even when the sub-block sizes of quantization matrices differ from each other, one of the quantization matrices may be shared for the sub-block sizes

having the same number of pixels. For example, according to the present exemplary embodiment, three 64-pixel sub-block sizes, that is, an  $8 \times 8$  pixel size, a  $16 \times 4$  pixel size, and a  $4 \times 16$  pixel size are employed. In this case, the quantization matrix for an  $8 \times 8$  pixel sub-block may be used for a  $16 \times 4$  pixel sub-block and a  $4 \times 16$  pixel sub-block. Similarly, the quantization matrix for a  $16 \times 4$  pixel sub-block may be used for a  $4 \times 16$  pixel sub-block. Even when the quantization matrix is not shared, a difference between the quantization matrix and the quantization matrix corresponding to a different sub-block may be encoded. For example, when the quantization matrix for  $16 \times 4$  pixels is encoded, a difference between each of the elements of the quantization matrix and the corresponding one of the elements of the quantization matrix for  $8 \times 8$  pixels may be encoded.

[0058] In addition, in order to encode two quantization matrices for the sub-block sizes having different numbers of pixels, one of the quantization matrices may be encoded through quantization for the other sub-block size. For example, an  $8 \times 8$  pixel quantization matrix may be encoded on an element basis. For a  $4 \times 4$  pixel quantization matrix, a prediction value may be computed by averaging  $2 \times 2$  elements in the  $8 \times 8$  pixel quantization matrix or thinning out the elements, and the difference therebetween may be encoded. As an example according to the present exemplary embodiment, each of the elements of the  $16 \times 16$  pixel quantization matrix is encoded. For the  $16 \times 4$  pixel quantization matrix and the  $4 \times 16$  pixel quantization matrix, the  $16 \times 16$  pixel quantization matrix is thinned out on a  $1 \times 4$  element basis or a  $4 \times 1$  element basis, or an average value is computed on a  $1 \times 4$  element basis or a  $4 \times 1$  element basis. Thereafter, the values may be directly used as the element of the quantization matrix. Alternatively, the values may be used as prediction values, and the differences may be encoded. Conversely, each of the elements of the  $4 \times 4$  pixel quantization matrix may be encoded. Thereafter, for the  $16 \times 4$  pixel quantization matrix and the  $4 \times 16$  pixel quantization matrix, each of the elements of the  $4 \times 4$  pixel quantization matrix may be subjected to interpolation, and the interpolated elements may be directly used as the elements of the  $16 \times 4$  pixel quantization matrix and the  $4 \times 16$  pixel quantization matrix. Alternatively, the values may be used as prediction values, and the differences may be encoded. Note that while description has been made with reference to a technique in which each of the elements is encoded in order to encode the quantization matrix, a technique is not limited thereto.

[0059] In addition, while the present exemplary embodiment has been described with reference to a technique for switching a quantization matrix input to the quantization unit 104 in accordance with the sub-block size, the technique is not limited thereto. For example, a quantization unit corresponding each of the quantization matrices may be provided, and one of the quantization units may be selected on the basis of the output

of the quantization matrix selection unit 107.

- [0060] In addition, while the present exemplary embodiment has been described with reference to a technique in which a quantization matrix is selected after the prediction error is subjected to orthogonal transform, the technique is not limited thereto. For example, orthogonal transform may be applied after the quantization matrix is selected, or selection of a quantization matrix and orthogonal transform may be performed at the same time.

### **Second Exemplary Embodiment**

- [0061] Fig. 3 is a block diagram of the configuration of an image decoding apparatus according to a second exemplary embodiment of the present invention. According to the present exemplary embodiment, decoding of the bit stream generated in the first exemplary embodiment illustrated in Fig. 1 is described.
- [0062] As illustrated in Fig. 3, a decoding/separating unit 301 decodes the header information of an input bit stream and separates necessary code from the bit stream. Thereafter, the decoding/separating unit 301 outputs the code to units disposed downstream thereof. The decoding/separating unit 301 performs an operation that is the reverse of the operation performed by the code integrating unit 106 illustrated in Fig. 1.
- [0063] A quantization matrix decoding unit 308 extracts the quantization matrix code data from the header information of the bit stream and decodes the extracted quantization matrix code data.
- [0064] A quantization matrix holding unit 309 temporarily stores a quantization matrices decoded by the quantization matrix decoding unit 308.
- [0065] A prediction/transform method extracting unit 306 decodes prediction/transform information from the code separated by the decoding/separating unit 301 on a basic block basis and determines the sub-block size on the basis of the prediction/transform information. A prediction method and prediction/transform are applied on the basis of the sub-block size. According to the present exemplary embodiment, the sub-block size and the shape are the same as the orthogonal transform size and shape. However, they are not necessarily the same.
- [0066] A quantization matrix selection unit 307 selects one of the quantization matrices stored in the quantization matrix holding unit 309 on the basis of the sub-block size determined by the prediction/transform method extracting unit 306.
- [0067] A coefficient decoding unit 302 decodes quantized coefficient code from the code separated by the decoding/separating unit 301 and reconstructs the quantized coefficient.
- [0068] An inverse quantization unit 303 inverse-quantizes the quantized coefficient using the quantization matrix selected by the quantization matrix selection unit 307 and re-

constructs the orthogonal transformed coefficient.

- [0069] An inverse transform unit 304 applies inverse orthogonal transform that is the reverse of the operation performed by the transform unit 103 illustrated in Fig. 1 and re-constructs the prediction error.
- [0070] A prediction reconstruction unit 305 reconstructs the image data of the basic block from the prediction information, such as the sub-block size by which prediction is performed, the prediction error, and already decoded image data.
- [0071] An exemplary image decoding operation performed by the above-described image decoding apparatus is described below. According to the present exemplary embodiment, the bit stream of a moving image generated in the first exemplary embodiment is input on a frame basis. However, a configuration for receiving the bit stream of a still image for one frame may be employed. In addition, according to the present exemplary embodiment, for simplicity, only an intra prediction decoding process is described below. However, the following image decoding operation can be applied to an inter prediction decoding process.
- [0072] As illustrated in Fig. 3, stream data for one frame is input to the decoding/separating unit 301. The decoding/separating unit 301 decodes the header information required for reconstructing the image. In addition, the decoding/separating unit 301 separates code used in the subsequent process and outputs the code. The quantization matrix code data included in the header information is input to the quantization matrix decoding unit 308. Thus, the quantization matrix corresponding to the sub-block size is reconstructed. The quantization matrix is used in the subsequent inverse quantization process. According to the present exemplary embodiment, the first quantization matrix for a  $4 * 4$  pixel sub-block, the second quantization matrix for an  $8 * 8$  pixel sub-block, and the third quantization matrix for a  $16 * 16$  pixel sub-block are reconstructed. In addition, the fourth quantization matrix for a  $16 * 4$  pixel sub-block, and the fifth quantization matrix for a  $4 * 16$  pixel sub-block are reconstructed. According to the present exemplary embodiment, five sub-block sizes, that is,  $4 * 4$  pixels,  $8 * 8$  pixels,  $16 * 16$  pixels,  $16 * 4$  pixels, and  $4 * 16$  pixels are employed. Accordingly, five types of quantization matrix are reconstructed. The number of types of quantization matrix varies with the number of the orthogonal transform shapes. The reconstructed quantization matrix is input to the quantization matrix holding unit 309 and is temporarily stored in the quantization matrix holding unit 309.
- [0073] Among codes separated by the decoding/separating unit 301, codes related to prediction and transform are input to the prediction/transform method extracting unit 306 and are decoded. Thus, information indicating the prediction method and the sub-block size is generated. Information indicating the sub-block size is output to the quantization matrix selection unit 307 and the coefficient decoding unit 302. In addition,

decoded information indicating the prediction method is output to the prediction reconstruction unit 305.

- [0074] The quantization matrix selection unit 307 selects one of the quantization matrices stored in the quantization matrix holding unit 309 using the input sub-block size information and outputs the selected quantization matrix to the inverse quantization unit 303. For example, according to the present exemplary embodiment, if the sub-block size is  $4 * 16$  pixels, the quantization matrix selection unit 307 selects the fifth quantization matrix in the quantization matrix holding unit 309 and outputs the fifth quantization matrix to the inverse quantization unit 303.
- [0075] Furthermore, among the codes separated by the decoding/separating unit 301, the quantized coefficient code data is input to the coefficient decoding unit 302. The coefficient decoding unit 302 decodes the quantized coefficient code data on a sub-block size basis input from the prediction/transform method extracting unit 306. Thus, the quantized coefficient is reconstructed. The reconstructed quantized coefficient is output to the inverse quantization unit 303.
- [0076] The inverse quantization unit 303 receives the quantization matrix selected by the quantization matrix selection unit 307 and the quantized coefficient reconstructed by the coefficient decoding unit 302. Thereafter, the inverse quantization unit 303 applies inverse quantization using the quantization matrix to reconstruct the orthogonal transformed coefficient and outputs the reconstructed orthogonal transformed coefficient to the inverse transform unit 304.
- [0077] The inverse transform unit 304 receives the reconstructed orthogonal transformed coefficient and applies inverse orthogonal transform, which is the reverse of the operation performed by the transform unit 103 illustrated in Fig. 1, to the orthogonal transformed coefficient. Thus, the inverse transform unit 304 reconstructs the prediction error and outputs the reconstructed prediction error to the prediction reconstruction unit 305.
- [0078] The prediction reconstruction unit 305 performs prediction on the input prediction error using decoded surrounding pixel data on a sub-block basis in accordance with the prediction method decoded by the prediction/transform method extracting unit 306. Thus, the prediction reconstruction unit 305 reconstructs image data on a basic block basis and outputs the image data.
- [0079] Fig. 6 is a flowchart of an image decoding process performed by the image decoding apparatus according to the second exemplary embodiment.
- [0080] In step S601, the decoding/separating unit 301 decodes the header information first.
- [0081] In step S602, the quantization matrix decoding unit 308 decodes the quantization matrix code data included in the header information and reconstructs the quantization matrix used in the subsequent inverse quantization process. The decoded quantization

matrix is stored in the quantization matrix holding unit 309. According to the present exemplary embodiment, the first quantization matrix for a  $4 \times 4$  pixel sub-block size, the second quantization matrix for an  $8 \times 8$  pixel sub-block size, and the third quantization matrix for a  $16 \times 16$  pixel sub-block size are decoded and reconstructed. In addition, the fourth quantization matrix for a  $16 \times 4$  pixel sub-block size and the fifth quantization matrix for a  $4 \times 16$  pixel sub-block size are decoded and reconstructed. Note that the decoding order is not limited thereto.

- [0082] In step S603, the prediction/transform method extracting unit 306 decodes information related to prediction and transform on a basic block basis and generates the sub-block size using the decoded information. The prediction method and prediction/orthogonal transform are applied on the basis of a sub-block size.
- [0083] In step S604, a process to be performed next is selected on the basis of the sub-block size generated in step S603. If the sub-block size is  $4 \times 4$  pixels, the processing proceeds to step S605. However, if the sub-block size is  $8 \times 8$  pixels, the processing proceeds to step S625. If the sub-block size is  $16 \times 16$  pixels, the processing proceeds to step S645. If the sub-block size is  $16 \times 4$  pixels, the processing proceeds to step S665. If the sub-block size is  $4 \times 16$  pixels, the processing proceeds to step S685.
- [0084] In step S605, the quantization matrix selection unit 307 selects one of the quantization matrices applied to the sub-block on the basis of the sub-block size extracted in step S603. More specifically, since the extracted sub-block size is  $4 \times 4$  pixels, the quantization matrix selection unit 307 selects the first quantization matrix.
- [0085] In step S625, the quantization matrix selection unit 307 selects the quantization matrix to be applied to the sub-block on the basis of the sub-block size. More specifically, since the extracted sub-block size is  $8 \times 8$  pixels, the quantization matrix selection unit 307 selects the second quantization matrix. In step S645, the quantization matrix selection unit 307 selects the quantization matrix applied to the sub-block on the basis of the sub-block size. More specifically, since the extracted sub-block size is  $16 \times 16$  pixels, the quantization matrix selection unit 307 selects the third quantization matrix. In step S665, the quantization matrix selection unit 307 selects the quantization matrix applied to the sub-block on the basis of the sub-block size. More specifically, since the extracted sub-block size is  $16 \times 4$  pixels, the quantization matrix selection unit 307 selects the fourth quantization matrix. In step S685, the quantization matrix selection unit 307 selects the quantization matrix applied to the sub-block on the basis of the sub-block size. More specifically, since the extracted sub-block size is  $4 \times 16$  pixels, the quantization matrix selection unit 307 selects the fifth quantization matrix.
- [0086] In step S606, the coefficient decoding unit 302 decodes the quantized coefficient code data on a sub-block basis and reconstructs the quantized coefficient.
- [0087] In step S607, the inverse quantization unit 303 applies inverse quantization to the

quantized coefficient reconstructed in step S606 using the quantization matrix selected in steps S605, S625, S645, S665, or S685. Thus, the inverse quantization unit 303 reconstructs the orthogonal transformed coefficient.

[0088] In step S608, a process to be performed next is selected again on the basis of the sub-block size generated in step S603. If the sub-block size is  $4 \times 4$  pixels, the processing proceeds to step S609. However, if the sub-block size is  $8 \times 8$  pixels, the processing proceeds to step S629. If the sub-block size is  $16 \times 16$  pixels, the processing proceeds to step S649. If the sub-block size is  $16 \times 4$  pixels, the processing proceeds to step S669. If the sub-block size is  $4 \times 16$  pixels, the processing proceeds to step S689.

[0089] In step S609, the inverse transform unit 304 applies inverse orthogonal transform to the orthogonal transformed coefficient reconstructed in step S607 on a  $4 \times 4$  sub-block basis and reconstructs the prediction error. In step S629, the inverse transform unit 304 applies inverse orthogonal transform to the orthogonal transformed coefficient reconstructed in step S607 on an  $8 \times 8$  sub-block basis and reconstructs the prediction error. In step S649, the inverse transform unit 304 applies inverse orthogonal transform to the orthogonal transformed coefficient reconstructed in step S607 on a  $16 \times 16$  sub-block basis and reconstructs the prediction error. In step S669, the inverse transform unit 304 applies inverse orthogonal transform to the orthogonal transformed coefficient reconstructed in step S607 on a  $16 \times 4$  sub-block basis and reconstructs the prediction error. In step S689, the inverse transform unit 304 applies inverse orthogonal transform to the orthogonal transformed coefficient reconstructed in step S607 on a  $4 \times 16$  sub-block basis and reconstructs the prediction error.

[0090] In step S610, the prediction reconstruction unit 305 performs prediction using the decoded surrounding pixel data in accordance with the prediction method reconstructed in step S603. The prediction value is added to the prediction errors reconstructed in step S609, S629, S649, S669, or S689. Thus, the decoded image of the sub-block is reconstructed.

[0091] In step S611, the image decoding apparatus determines whether all of the sub-blocks in the basic block have been decoded. If all of the sub-blocks in the basic block have been decoded, the processing proceeds to step S612. However, if all of the sub-blocks in the basic block have not been decoded, the processing returns to step S604, where the processing for the next sub-block is started.

[0092] In step S612, the image decoding apparatus determines whether all of the basic blocks have been decoded. If all of the basic blocks have been decoded, all operations are stopped and the processing is completed. However, if all of the basic blocks have not been decoded, the processing returns to step S603, where the processing for the next basic block is started.

[0093] Through such a configuration and operations, the bit stream generated through the

quantization process performed using a quantization matrix best suited for the orthogonal transform shape (the shape indicated by the sub-block size) in the first exemplary embodiment can be decoded, and a reconstructed image can be obtained.

[0094] Note that like the first exemplary embodiment, the size and shape of the basic block are not limited to those described above. In addition, the size and shape of the sub-block are not limited to those described above.

[0095] If a plurality of the quantization matrix code data items are included in a bit stream of one sequence, the quantization matrix may be updated. The decoding/separating unit 301 detects the quantization matrix code data, and the quantization matrix decoding unit 308 decodes the quantization matrix code data. The decoded quantization matrix data is replaced with the corresponding quantization matrix stored in the quantization matrix holding unit 309. At that time, all of the quantization matrices may be replaced. Alternatively, by detecting the sub-block size of the quantization matrix to be replaced, only part of the quantization matrix can be updated. Still alternatively, by detecting the sub-block size of the quantization matrix to be replaced and the position of an element to be updated, only some of the elements of the matrix can be updated.

[0096] In addition, while the present exemplary embodiment has been described with reference to a technique in which one of quantization matrices input to the inverse quantization unit 303 is selected in accordance with the sub-block size, the technique is not limited thereto. For example, a plurality of inverse quantization units each corresponding to one of the quantization matrices may be prepared and, thereafter, one of the inverse quantization units may be selected in accordance with the output of the quantization matrix selection unit 307. Similarly, in Fig. 6, a plurality of steps S606 and a plurality of steps S607 may be prepared. Thereafter, determination is made in step S604, and selection of the quantization matrix, decoding of the coefficient, inverse quantization, and inverse transform may be performed for each of the sub-block sizes at the same time.

[0097] Furthermore, while the present exemplary embodiment has been described with reference to decoding of the quantized coefficient performed after selection of the quantization matrix, the sequence is not limited thereto. For example, selection of the quantization matrix may be made after the quantized coefficient is decoded. Alternatively, selection of the quantization matrix and decoding of the quantized coefficient may be performed at the same time.

### **Third Exemplary Embodiment**

[0098] Fig. 8 is a block diagram of an image encoding apparatus according to the present exemplary embodiment. The same numbering is used for components which were previously illustrated in Fig. 1 (the first exemplary embodiment), and description of the component is not repeated.

- [0099] An encoding method generation unit 851 generates encoding information indicating how to encode each of the quantization matrices.
- [0100] A quantization matrix encoding unit 809 encodes the quantization matrix stored in the quantization matrix holding unit 108 on the basis of the encoding information generated by the encoding method generation unit 851 and generates quantization matrix code data.
- [0101] Like the code integrating unit 106 illustrated in Fig. 1, a code integrating unit 806 generates code related to the header information, prediction, and transform. The code integrating unit 806 differs from the code integrating unit 106 in that the encoding information is received from the encoding method generation unit 851 and is encoded.
- [0102] An exemplary image encoding operation performed by the image encoding apparatus is described below.
- [0103] The encoding method generation unit 851 generates the encoding information indicating how to encode each of the quantization matrices first. According to the present exemplary embodiment, if the encoding information is "0", each of the elements of a quantization matrix is directly encoded. However, if the encoding information is "1", the difference between an element and the immediately previous element is encoded. If the encoding information is "2", the difference between an element and an element of the immediately previously encoded quantization matrix having the same number of elements is encoded. If the encoding information is "3", the elements of the immediately previously encoded quantization matrix having a larger number of elements are thinned out to generate a prediction quantization matrix. Thereafter, the difference between an element and an element of the prediction quantization matrix is encoded. Furthermore, if the encoding information is "4", the elements of the immediately previously encoded quantization matrix having a smaller number of elements are increased by interpolation to generate a prediction quantization matrix. Thereafter, the difference between an element and an element of the prediction quantization matrix is encoded. However, the method for encoding each of the elements of the quantization matrix is not limited thereto. For example, like DPCM, after the difference between a first element and a second element is encoded, the prediction difference may be encoded. Alternatively, by referring to the immediately previous decoded quantization matrix or a decoded quantization matrix indicated by other code, encoding may be performed. In the prediction method for a quantization matrix having a smaller number of elements using a quantization matrix having a larger number of elements, the elements are thinned out. However, the average value or the median value may be used. Alternatively, without encoding a difference, an identifier indicating that the difference is shared by another quantization matrix may be encoded. In addition, a combination of the quantization matrix encoding method information and the quan-

tization matrix encoding method is not limited thereto. Any method for generating the quantization matrix encoding method information can be employed. For example, the user may input the quantization matrix encoding method information. Alternatively, a predetermined fixed value may be used. Still alternatively, a value computed using the characteristics of the quantization matrices stored in the quantization matrix holding unit 108 may be used. The generated quantization matrix encoding method information is input to the quantization matrix encoding unit 809 and the code integrating unit 806.

[0104] The quantization matrix encoding unit 809 encodes each of the quantization matrices stored in the quantization matrix holding unit 108 on the basis of the input encoding information and generates quantization matrix code data. Thereafter, the quantization matrix encoding unit 809 outputs the generated quantization matrix code data to the code integrating unit 806.

[0105] The code integrating unit 806 encodes the encoding information generated by the encoding method generation unit 851 and generates encoding information code. Thereafter, the code integrating unit 806 embeds the generated encoding information code into, for example, the header information and outputs the encoding information code. In addition, the code integrating unit 806 encodes the sub-block size information generated by the prediction unit 102 and outputs the encoded sub-block size information. At that time, any encoding method can be used. For example, Huffman coding or arithmetic coding can be employed. Fig. 7B illustrates an example of a bit stream including the encoding information code. The encoding information code may be included in any one of the headers, such as a sequence header or a picture header. However, the encoding information code is located prior to each of the quantization matrix code data items.

[0106] Fig. 10 is a flowchart of an exemplary image encoding process performed by the image encoding apparatus according to the third exemplary embodiment. The same numbering is used for elements which were previously illustrated in Fig. 5 (the first exemplary embodiment), and description of the elements is not repeated.

[0107] In step S1051, the encoding method generation unit 851 determines the encoding information to be used in the subsequent step S1002.

[0108] In step S1002, the quantization matrix encoding unit 809 encodes the quantization matrix generated in step S501 on the basis of the encoding information determined in step S1051.

[0109] In step S1003, the code integrating unit 806 encodes the encoding information and generates the encoding information code. As for other code, the code integrating unit 806 embeds the encoding information code into the header portion and outputs the encoding information code.

[0110] Through such a configuration and operations, each of the quantization matrices is

encoded using an encoding method best suited for the quantization matrix. Thus, a bit stream subjected to a quantization process using an optimum quantization matrix corresponding to orthogonal transform indicated by the sub-block size can be generated. In this way, the amount of code generated by encoding the quantization matrix can be reduced.

- [0111] While the present exemplary embodiment has been described with reference to the case in which a single encoding method is applied to all of the quantization matrices to be encoded, the encoding process is not limited thereto. For example, an encoding method may be switched on a quantization matrix basis or a sub-block size basis.

#### **Fourth Exemplary Embodiment**

- [0112] Fig. 9 is a block diagram of an image decoding apparatus according to the present exemplary embodiment. The same numbering is used for components which were previously illustrated in Fig. 3 (the second exemplary embodiment), and description of the components is not repeated.
- [0113] A decoding/separating unit 901 decodes the header information of an input bit stream and separates necessary code from the bit stream. Thereafter, the decoding/separating unit 901 outputs the separated code to a unit downstream of the decoding/separating unit 901. The decoding/separating unit 901 differs from the decoding/separating unit 301 illustrated in Fig. 3 in that the encoding information code is separated from the header information of the bit stream and is output to a unit downstream of the decoding/separating unit 901.
- [0114] A decoding method decoding unit 951 decodes the encoding information code separated by the decoding/separating unit 901 and reconstructs the quantization matrix encoding method information.
- [0115] A quantization matrix decoding unit 908 decodes the quantization matrix code data separated from the header information of the bit stream by the decoding/separating unit 901 on the basis of the encoding method information.
- [0116] An exemplary image decoding operation performed by the image decoding apparatus is described below.
- [0117] As illustrated in Fig. 9, input stream data for one frame is input to the decoding/separating unit 901. The header information necessary for reconstructing the image is decoded. In addition, code to be used in the subsequent process is separated and output. The encoding information code included in the header information is input to the decoding method decoding unit 951, and the quantization matrix encoding method information is reconstructed. Thereafter, the reconstructed quantization matrix encoding method information is input to the quantization matrix decoding unit 908.
- [0118] In contrast, the quantization matrix code data included in the header information is input to the quantization matrix decoding unit 908. The quantization matrix decoding

unit 908 selects one of the quantization matrix decoding methods on the basis of the input quantization matrix encoding method information and decodes the quantization matrix code data. Thus, the quantization matrices to be used in the subsequent inverse quantization process are reconstructed. The reconstructed quantization matrices are temporarily stored in the quantization matrix holding unit 309.

- [0119] Fig. 11 is a flowchart of an exemplary image decoding process performed by the image decoding apparatus according to the fourth exemplary embodiment. The same numbering is used for elements which were previously illustrated in Fig. 6 (the second exemplary embodiment), and description of the elements is not repeated.
- [0120] In step S1101, the decoding/separating unit 901 decodes the header information.
- [0121] In step S1151, the decoding method decoding unit 951 decodes the quantization matrix encoding information code included in the header information and reconstructs the quantization matrix encoding method information.
- [0122] In step S1102, the quantization matrix decoding unit 908 selects one of the quantization matrix decoding methods on the basis of the quantization matrix encoding method information reconstructed in step S1151. Thereafter, the quantization matrix decoding unit 908 decodes the quantization matrix code data included in the header information and reconstructs the quantization matrices.
- [0123] Through such a configuration and operations, the bit stream generated through the quantization process performed using quantization matrices encoded using an encoding method best suited for the sub-block size in the third exemplary embodiment can be decoded, and a reconstructed image can be obtained. In this way, a bit stream having a reduced amount of code through an encoding process adaptive to quantization matrices can be obtained.
- [0124] While the present exemplary embodiment has been described with reference to the case in which only one decoding method is selected for all of the quantization matrix code data items to be decoded, selection of a decoding method is not limited thereto. A decoding method may be selected on a quantization matrix basis. Alternatively, a decoding method may be selected for each of the sub-block sizes.

### **Fifth Exemplary Embodiment**

- [0125] Fig. 13 is a block diagram of an image encoding apparatus according to the present exemplary embodiment. The same numbering is used for components which were previously illustrated in Fig. 1 (the first exemplary embodiment), and description of the components is not repeated.
- [0126] A quantization matrix holding unit 1308 generates a quantization matrix and temporarily holds the generated quantization matrix. Any method for generating a quantization matrix to be held can be employed. For example, the user may input the quantization matrix. Alternatively, the quantization matrix may be computed using the char-

acteristics of an input image, or a quantization matrix predetermined as an initial value may be used. According to the present exemplary embodiment, quantization matrices corresponding to square orthogonal transform, such as  $4 * 4$  pixel,  $8 * 8$  pixel, and  $16 * 16$  pixel orthogonal transform, are generated and stored. In addition, quantization matrices that correspond to rectangular orthogonal transform, such as  $16 * 4$  pixel and  $4 * 16$  pixel orthogonal transform, and that are generated by a quantization matrix reduction unit 1351 (described in more detail below) are input and stored. However, generated and stored quantization matrices are not limited thereto. For example, a  $32 * 32$  pixel square quantization matrix may be further generated and, thereafter, the reduced  $32 * 8$  pixel and  $8 * 32$  pixel rectangular quantization matrices may be further stored.

- [0127] A quantization matrix reduction unit 1351 reduces one side of a square quantization matrix and generates a rectangular quantization matrix. According to the present exemplary embodiment, the quantization matrix reduction unit 1351 receives a  $16 * 16$  pixel square quantization matrix generated by the quantization matrix holding unit 1308 and performs a reduction process to generate a  $4 * 16$  pixel quantization matrix and a  $16 * 4$  pixel quantization matrix. The reduction process is described below with reference to Figs. 17A to 17C.
- [0128] Fig. 17A illustrates a  $16 * 16$  pixel quantization matrix modeled using symbols. The symbols "double circle", "circle", "triangle", and "cross" represent elements of a quantization matrix. According to the present exemplary embodiment, the quantization matrix reduction unit 1351 extracts the elements at the positions indicated by a double circle and a circle from the  $16 * 16$  pixel quantization matrix illustrated in Fig. 17A in the vertical direction and, thus, generates a  $16 * 4$  pixel quantization matrix illustrated in Fig. 17B. In addition, the quantization matrix reduction unit 1351 extracts the elements at the positions indicated by a double circle and a triangle from the  $16 * 16$  pixel quantization matrix illustrated in Fig. 17A in the horizontal direction and, thus, generates a  $4 * 16$  pixel quantization matrix illustrated in Fig. 17C.
- [0129] A quantization matrix encoding unit 1309 encodes a square quantization matrix among the quantization matrices stored in the quantization matrix holding unit 1308 and generates quantization matrix code data. Like the quantization matrix encoding unit 109 illustrated in Fig. 1, any method for encoding the quantization matrix can be employed. For example, Huffman coding or arithmetic coding can be directly applied to the value of each of the elements of the quantization matrix. Alternatively, Huffman coding or arithmetic coding may be applied to a difference between the value of the element and the value of the immediately previous element or a difference between the value of the element and the value of an element of a different quantization matrix.
- [0130] A code integrating unit 1306 generates code related to header information,

prediction, and transform. In addition, the code integrating unit 1306 integrates the quantized coefficient code data generated by the coefficient encoding unit 105 with the quantization matrix code data generated by the quantization matrix encoding unit 1309.

[0131] An exemplary image encoding operation performed by the image encoding apparatus is described below.

[0132] The quantization matrix holding unit 1308 stores the quantization matrices for square sub-block shapes among the quantization matrices used for quantizing the frame. According to the present exemplary embodiment, five sub-block sizes, that is, the  $4 \times 4$  pixel size, the  $8 \times 8$  pixel size, the  $16 \times 16$  pixel size, the  $16 \times 4$  pixel size, and the  $4 \times 16$  pixel size, are employed. Thus, the quantization matrix holding unit 1308 holds the first quantization matrix corresponding to  $4 \times 4$  pixel orthogonal transform, the second quantization matrix corresponding to  $8 \times 8$  pixel orthogonal transform, and the third quantization matrix corresponding to  $16 \times 16$  pixel orthogonal transform. According to the present exemplary embodiment, the quantization matrix holding unit 1308 holds these three quantization matrices. If the number of the shapes of the sub-blocks increases or decreases, the number of quantization matrices held in the quantization matrix holding unit 1308 increases or decreases. In addition, since the quantization matrices corresponding to  $16 \times 4$  pixel orthogonal transform and  $4 \times 16$  orthogonal transform need to be generated, the quantization matrix holding unit 1308 outputs the  $16 \times 16$  pixel quantization matrix to the quantization matrix reduction unit 1351.

[0133] The quantization matrix reduction unit 1351 reduces the input  $16 \times 16$  pixel quantization matrix using the above-described method and generates a  $16 \times 4$  pixel quantization matrix and a  $4 \times 16$  pixel quantization matrix. Thereafter, the quantization matrix reduction unit 1351 outputs the  $16 \times 4$  pixel quantization matrix and the  $4 \times 16$  pixel quantization matrix to the quantization matrix holding unit 1308.

[0134] The quantization matrix holding unit 1308 further stores the  $16 \times 4$  pixel quantization matrix and the  $4 \times 16$  pixel quantization matrix input from the quantization matrix reduction unit 1351. That is, the quantization matrix holding unit 1308 further holds the fourth quantization matrix corresponding to  $16 \times 4$  pixel orthogonal transform and the fifth quantization matrix corresponding to  $4 \times 16$  pixel orthogonal transform.

[0135] The quantization matrix encoding unit 1309 sequentially reads the quantization matrices corresponding to square orthogonal transform from the quantization matrix holding unit 1308 in the order of increasing sub-block size and encodes the quantization matrix. Thus, the quantization matrix encoding unit 1309 generates the quantization matrix code data. According to the present exemplary embodiment, three quantization matrices, that is, the first quantization matrix corresponding to  $4 \times 4$  pixel orthogonal transform, the second quantization matrix corresponding to  $8 \times 8$  pixel orthogonal transform, and the third quantization matrix corresponding to  $16 \times 16$  pixel

orthogonal transform are encoded. As in the quantization matrix encoding unit 109 illustrated in Fig. 1, the encoding method is not limited to the above-described encoding method.

- [0136] Like the code integrating unit 106 illustrated in Fig. 1, the code integrating unit 1306 generates code related to the sequence header, the frame header, the picture header, and a slice header. In addition, the code integrating unit 1306 embeds the quantization matrix code data generated by the quantization matrix encoding unit 1309 into any one of the headers. Fig. 7C illustrates an example of a bit stream output in the present exemplary embodiment. According to the present exemplary embodiment, the quantization matrices are encoded into the sequence header of the bit stream. However, the encoding location is not limited thereto.
- [0137] Fig. 15 is a flowchart of an exemplary image encoding process performed by the image encoding apparatus according to the fifth exemplary embodiment. The same numbering is used for elements which were previously illustrated in Fig. 5 (the first exemplary embodiment), and description of the elements is not repeated.
- [0138] In step S1501, the quantization matrix holding unit 1308 generates at least one quantization matrix. According to the present exemplary embodiment, five sub-block sizes are employed. Among these sub-block sizes, three sub-block sizes have a square shape. Accordingly, as described above, the first to third quantization matrices are generated.
- [0139] In step S1502, the quantization matrix encoding unit 1309 encodes each of the square quantization matrices generated by the quantization matrix holding unit 1308.
- [0140] In step S1551, the quantization matrix reduction unit 1351 reduces the third quantization matrix in the above-described manner. Thus, the quantization matrix reduction unit 1351 generates the fourth quantization matrix corresponding to  $16 * 4$  orthogonal transform and the fifth quantization matrix corresponding to  $4 * 16$  orthogonal transform.
- [0141] In step S1503, the code integrating unit 1306 encodes the header portion of the bit stream and outputs the encoded header portion. At that time, the code integrating unit 1306 outputs the quantization matrix code data encoded in step S1502 together with the header portion.
- [0142] According to the present exemplary embodiment, step S1551 is performed between steps S1502 and S1503. However, the order of performance of step S1551 is not limited thereto. Step S1551 can be performed after step S1501 is performed and before step S504 is performed.
- [0143] Through such a configuration and operations, in order to generate a bit stream, quantization can be performed using the quantization matrix best suited for the orthogonal transform indicated by each of the sub-block sizes, and only square quantization matrices can be encoded. In this way, the amount of code generated by encoding the

quantization matrices can be reduced.

- [0144] While the present exemplary embodiment has been described with reference to a process in which a quantization matrix that corresponds to rectangular orthogonal transform and that is generated by the quantization matrix reduction unit 1351 is stored in the quantization matrix holding unit 1308, the process is not limited thereto. For example, the quantization matrix holding unit 1308 may hold only square quantization matrices. Each time a rectangular quantization matrix is used, a square quantization matrix may be reduced to generate the rectangular quantization matrix. In this way, the quantization matrix holding unit 1308 can reduce an amount of memory required for holding the quantization matrices.
- [0145] In addition, according to the present exemplary embodiment, in order to reduce (thin out) 16 pixels to 4 pixels,  $4n$ -th elements (where  $n$  is an integer) are extracted, and the other elements are removed. In this way, the reduction process is performed. However, the reduction process is not limited thereto. For example,  $(4n + 1)$ -th elements,  $(4n + 2)$ -th elements, or  $(4n + 3)$ -th elements may be extracted, and the other elements may be removed. Thus, the reduction process may be performed. Alternatively, by averaging the following four elements:  $4n$ -th,  $(4n + 1)$ -th,  $(4n + 2)$ -th, and  $(4n + 3)$ -th elements, the reduction process may be performed.
- [0146] Furthermore, while the present exemplary embodiment has been described with reference to a  $1/4$  time reduction process in which  $16 * 16$  pixels are reduced to  $16 * 4$  pixels or  $4 * 16$  pixels, the scaling factor is not limited thereto. Even when a  $1/8$  time reduction process in which  $16 * 16$  pixels are reduced to  $16 * 2$  pixels or  $2 * 16$  pixels is employed or a  $1/2$  time reduction process in which  $16 * 16$  pixels are reduced to  $16 * 8$  pixels or  $8 * 16$  pixels is employed, the advantage of the present invention can be still provided. Still alternatively, even when  $16 * 16$  pixels are reduced to  $16 * 4$  pixels and, subsequently, the generated  $16 * 4$  pixels are converted into  $16 * 2$  pixels, the advantage of the present invention can be still provided.
- [0147] Still furthermore, while the present exemplary embodiment has been described with reference to the reduction process in which a  $16 * 16$  pixel quantization matrix is reduced, the reduction process is not limited thereto. For example, a quantization matrix having an  $8 * 8$  pixel size, a  $32 * 32$  pixel size, or a larger pixel size may be reduced to generate a rectangular quantization matrix.

### **Sixth Exemplary Embodiment**

- [0148] Fig. 14 is a block diagram of an image decoding apparatus according to a sixth exemplary embodiment of the present invention. According to the present exemplary embodiment, the bit stream generated in the fifth exemplary embodiment illustrated in Fig. 13 is described. The same numbering is used for components which were previously illustrated in Fig. 3 (the second exemplary embodiment), and description of

the components is not repeated.

- [0149] As illustrated in Fig. 14, a decoding/separating unit 14001 decodes the header information of an input bit stream and separates necessary code from the bit stream. Thereafter, the decoding/separating unit 14001 outputs the separated code to units downstream of the decoding/separating unit 14001. The decoding/separating unit 14001 performs a process that is the reverse of the process performed by the code integrating unit 1306 illustrated in Fig. 13.
- [0150] A quantization matrix decoding unit 14008 extracts the quantization matrix code data from the header information of the bit stream and decodes the quantization matrix code data. In the bit stream generated in the sixth exemplary embodiment, quantization matrices corresponding to  $4 * 4$  pixel,  $8 * 8$  pixel, and  $16 * 16$  pixel square orthogonal transform are encoded. Therefore, according to the present exemplary embodiment, these three types of quantization matrices are reconstructed.
- [0151] A quantization matrix reduction unit 14051 reduces one side of a square quantization matrix and generates a rectangular quantization matrix. The quantization matrix reduction unit 14051 performs a process that is the same as the process performed by the quantization matrix reduction unit 1351 of the sixth exemplary embodiment illustrated in Fig. 13. According to the present exemplary embodiment, the quantization matrix reduction unit 14051 receives a  $16 * 16$  pixel square quantization matrix generated by the quantization matrix decoding unit 14008 and performs a reduction process to generate a  $4 * 16$  pixel quantization matrix and a  $16 * 4$  pixel quantization matrix. The reduction process is the same as the reduction process performed by the quantization matrix reduction unit 1351 illustrated in Fig. 13.
- [0152] A quantization matrix holding unit 14009 temporarily holds the square quantization matrices reconstructed by the quantization matrix decoding unit 14008 and the rectangular quantization matrices generated by the quantization matrix reduction unit 14051.
- [0153] An exemplary image decoding operation performed by the image decoding apparatus is described below. According to the present exemplary embodiment, the moving image bit stream generated in the fifth exemplary embodiment is input on a frame basis. However, a still image bit stream for one frame may be input. Alternatively, bit streams separated on a slice basis or on a packet basis may be sequentially input.
- [0154] As illustrated in Fig. 14, the input stream data for one frame is input to the decoding/separating unit 14001. The decoding/separating unit 14001 decodes the header information required for reconstructing the image. In addition, the decoding/separating unit 14001 separates code to be used in the subsequent process and outputs the code.
- [0155] The quantization matrix code data included in the header information is input to the quantization matrix decoding unit 14008. The quantization matrix decoding unit 14008

reconstructs the quantization matrix that corresponds to the sub-block size and that is to be used in the subsequent inverse quantization process. According to the present exemplary embodiment, three types of square quantization matrix, that is, the first quantization matrix corresponding to a  $4 \times 4$  pixel sub-block, the second quantization matrix corresponding to an  $8 \times 8$  pixel sub-block, the third quantization matrix corresponding to a  $16 \times 16$  pixel sub-block, are reconstructed. According to the present exemplary embodiment, five sub-block sizes, that is, the  $4 \times 4$  pixel size, the  $8 \times 8$  pixel size, the  $16 \times 16$  pixel size, the  $16 \times 4$  pixel size, and the  $4 \times 16$  pixel size are employed. Among these sub-block sizes, the number of square sub-block sizes is three. Accordingly, the three types of quantization matrix are reconstructed. However, if the number of the orthogonal transform shapes increases or decreases, the number of types of quantization matrix to be reconstructed increases or decreases.

[0156] The reconstructed square quantization matrix is input to the quantization matrix reduction unit 14051. The quantization matrix reduction unit 14051 generates a quantization matrix corresponding to rectangular orthogonal transform from the input square quantization matrix. According to the present exemplary embodiment, two type of orthogonal transform size, that is,  $16 \times 4$  pixels and  $4 \times 16$  pixels, are employed. Thus, a  $16 \times 4$  pixel quantization matrix and a  $4 \times 16$  pixel quantization matrix are generated by reducing the  $16 \times 16$  pixel quantization matrix. The generated rectangular quantization matrices are input to the quantization matrix holding unit 14009 together with the square quantization matrix reconstructed by the quantization matrix decoding unit 14008 and are temporarily stored in the quantization matrix holding unit 14009.

[0157] In addition, among codes separated by the decoding/separating unit 14001, the codes related to prediction and transform are input to the prediction/transform method extracting unit 306 and are decoded. Thus, the information indicating a prediction method and the sub-block size is generated. In addition, among codes separated by the decoding/separating unit 14001, the quantized coefficient code data is input to the coefficient decoding unit 302.

[0158] Fig. 16 is a flowchart of an exemplary image decoding process performed by the image decoding apparatus according to the sixth exemplary embodiment. The same numbering is used for elements which were previously illustrated in Fig. 6 (the second exemplary embodiment), and description of the elements is not repeated.

[0159] In step S1601, the decoding/separating unit 14001 decodes the header information first.

[0160] In step S1602, the quantization matrix decoding unit 14008 decodes the quantization matrix code data included in the header information and generates the quantization matrix to be used in the subsequent inverse quantization process. According to the present exemplary embodiment, quantization matrices corresponding to square sub-

block sizes, that is, the first quantization matrix corresponding to  $4 * 4$  pixels, the second quantization matrix corresponding to  $8 * 8$  pixels, and the third quantization matrix corresponding to  $16 * 16$  pixels, are decoded and reconstructed. Note that at that time, any decoding order can be employed.

[0161] In step S1651, the quantization matrix reduction unit 14051 reduces the third quantization matrix in the above-described manner and generates the fourth quantization matrix corresponding to  $16 * 4$  pixel orthogonal transform and the fifth quantization matrix corresponding to  $4 * 16$  pixel orthogonal transform. The process performed in step S1651 is the same as the process performed in step S1551 illustrated in Fig. 15.

[0162] Through such a configuration and operations, the bit stream generated through the quantization process performed using a quantization matrix best suited for the orthogonal transform shape (the shape indicated by the sub-block size) and an encoding process in which only square quantization matrices are encoded in the fifth exemplary embodiment can be decoded, and a reconstructed image can be obtained. In this way, a bit stream having a reduced amount of code generated when the quantization matrices are encoded can be decoded.

[0163] While the present exemplary embodiment has been described with reference to a process in which a quantization matrix that corresponds to rectangular orthogonal transform and that is generated by the quantization matrix reduction unit 14051 is stored in the quantization matrix holding unit 14009, the process is not limited thereto. For example, the quantization matrix holding unit 14009 may store only square quantization matrices. Each time a rectangular quantization matrix is used, a square quantization matrix may be reduced to generate the rectangular quantization matrix. In this way, the quantization matrix holding unit 14009 can reduce an amount of memory required for holding the quantization matrices.

[0164] In addition, according to the present exemplary embodiment, in order to reduce (thin out)  $16$  pixels to  $4$  pixels,  $4n$ -th elements (where  $n$  is an integer) are extracted, and the other elements are removed. In this way, the reduction process is performed. However, the reduction process is not limited thereto. For example,  $(4n + 1)$ -th elements,  $(4n + 2)$ -th elements, or  $(4n + 3)$ -th elements may be extracted, and the other elements may be removed. Thus, the reduction process may be performed. Alternatively, by averaging the following four elements:  $4n$ -th,  $(4n + 1)$ -th,  $(4n + 2)$ -th, and  $(4n + 3)$ -th elements, the reduction process may be performed.

[0165] Furthermore, while the present exemplary embodiment has been described with reference to a  $1/4$  time reduction process in which  $16 * 16$  pixels are reduced to  $16 * 4$  pixels or  $4 * 16$  pixels, the scaling factor is not limited thereto. Even when a  $1/8$  time reduction process in which  $16 * 16$  pixels are reduced to  $16 * 2$  pixels or  $2 * 16$  pixels is employed or a  $1/2$  time reduction process in which  $16 * 16$  pixels are reduced to  $16 *$

8 pixels or  $8 * 16$  pixels is employed, the advantage of the present invention can be still provided.

- [0166] Still alternatively, while the present exemplary embodiment has been described with reference to reduction of a  $16 * 16$  pixel quantization matrix, a  $32 * 32$  pixel quantization matrix and an  $8 * 8$  pixel quantization matrix may be reduced to generate a rectangular quantization matrix.

### **Seventh Exemplary Embodiment**

- [0167] While the above exemplary embodiments have been described with reference to the processing units (illustrated in Figs. 1, 3, 8, 9, 13, and 14) formed as hardware, the processes performed by these processing units may be realized by computer programs.
- [0168] Fig. 12 is a block diagram of exemplary computer hardware applicable to the image encoding apparatus and the image decoding apparatus according to the above-described exemplary embodiments.
- [0169] A central processing unit (CPU) 1401 performs overall control of the computer using computer programs and data stored in a random access memory (RAM) 1402 and a read only memory (ROM) 1403. In addition, the CPU 1401 performs the above-described processing performed by the image encoding apparatus and the image decoding apparatus according to the above-described exemplary embodiments. That is, the CPU 1401 functions as the processing units illustrated in Figs. 1, 3, 8, 9, 13, and 14.
- [0170] The RAM 1402 has a memory area that temporarily holds a computer program and data loaded from an external storage unit 1406 and data externally acquired via an interface (I/F) 1409. In addition, the RAM 1402 has a work area used when the CPU 1401 performs a variety of processes. That is, the RAM 1402 is used as, for example, a frame memory or a variety of memory areas.
- [0171] The ROM 1403 stores the setting data and a boot program of the computer. An operation unit 1404 includes, for example, a keyboard and a mouse. By manipulating the operation unit 1404, a user of the computer can input a variety of instructions to the CPU 1401. A display unit 1405 displays the result of processing performed by the CPU 1401. The display unit 1405 is formed from, for example, a display device, such as a liquid crystal display.
- [0172] The external storage unit 1406 is a high-capacity information storage unit, such as a hard disk drive. The external storage unit 1406 stores an operating system (OS) and the computer programs used by the CPU 1401 that realizes the functions of the units illustrated in Figs. 1, 3, 8, 9, 13, and 14. Furthermore, the external storage unit 1406 may store image data items to be processed.
- [0173] The computer program and data stored in the external storage unit 1406 are loaded into the RAM 1402 as needed under the control of the CPU 1401 and are processed by

the CPU 1401. The I/F 1407 can have a local area network (LAN), a network such as the Internet, a projector, and a display unit connected thereto. The computer can receive and transmit a variety of information items via the I/F 1407. A bus 1408 connects the above-described units with one another.

[0174] The CPU 1401 mainly controls the operations illustrated in the above-described flowcharts.

### **Other Exemplary Embodiments**

[0175] The functions of the above-described exemplary embodiments can be also realized by mounting a storage medium having a computer program that realizes the functions in a system that reads the code of the computer program and executes the functions. In such a case, the code of the computer program read out of the storage medium realizes the functions of the above-described exemplary embodiments, and the storage medium storing the code of the computer program provides the present invention. In addition, the case in which, for example, the operating system (OS) that runs in a computer in accordance with the instructions of the program code executes some of or all of the above-described functions to realize the functions is encompassed within the invention.

[0176] Furthermore, the present invention may be realized through the following embodiment. That is, the computer program code read out of the storage medium is written to a memory of an add-on expansion board mounted in a computer or a memory of an add-on expansion unit connected to a computer. After the program code is written, a CPU in the add-on expansion board or in the add-on expansion unit executes some of or all of the functions of the above-described embodiments under the control of the computer program code.

[0177] When the present invention is applied to the above-described storage medium, the storage medium stores program code of the computer programs corresponding to the above-described flowcharts.

[0178] While the present invention has been described with reference to exemplary embodiments, it is to be understood that the invention is not limited to the disclosed exemplary embodiments. The scope of the following claims is to be accorded the broadest interpretation so as to encompass all such modifications and equivalent structures and functions.

[0179] This application claims the benefit of Japanese Patent Application No. 2011-155012, filed July 13, 2011 and No. 2011-243937, filed November 7, 2011, which are hereby incorporated by reference herein in their entirety.

## Claims

- [Claim 1] An image encoding apparatus comprising:  
a block dividing unit configured to divide an input image into a plurality of blocks;  
a prediction unit configured to perform prediction on each of the blocks divided by the block dividing unit using already encoded blocks and generate a prediction error;  
a transform unit configured to determine a shape of a processing block to be used from a plurality of square or rectangular shapes of processing blocks for orthogonal transform, apply orthogonal transform of the determined shape to the prediction error generated by the prediction unit, and generate a transformed coefficient;  
a quantization matrix selection unit configured to select a quantization matrix on the basis of the shape of a processing block for orthogonal transform determined by the transform unit;  
a quantization unit configured to quantize the transformed coefficient using the quantization matrix selected by the quantization matrix selection unit and generate a quantized coefficient; and  
a coefficient encoding unit configured to encode the quantized coefficient generated by the quantization unit.
- [Claim 2] The image encoding apparatus according to Claim 1, further comprising:  
a quantization matrix generation unit configured to generate quantization matrices corresponding to the square or rectangular shapes of the processing blocks for orthogonal transform used when the transformed coefficient is quantized;  
a quantization matrix encoding unit configured to encode the quantization matrices generated by the quantization matrix generation unit; and  
a code integrating unit configured to integrate the quantization matrix encoded by the quantization matrix encoding unit with the quantized coefficient encoded by the coefficient encoding unit.
- [Claim 3] An image decoding apparatus comprising:  
a decoding unit configured to separate necessary code from an input bit stream;  
a determination unit configured to determine a shape of a processing block for inverse orthogonal transform from information decoded by

the decoding unit;  
a coefficient decoding unit configured to decode a quantized coefficient from information decoded by the decoding unit;  
a quantization matrix decoding unit configured to decode a quantization matrix used for inverse quantizing the quantized coefficient decoded by the coefficient decoding unit;  
a quantization matrix selection unit configured to select a quantization matrix on the basis of the shape of a processing block for inverse orthogonal transform determined by the determination unit;  
an inverse quantization unit configured to inverse quantize the quantized coefficient decoded by the coefficient decoding unit using the quantization matrix selected by the quantization matrix selection unit and generate a transformed coefficient; and  
an inverse orthogonal transform unit configured to apply inverse orthogonal transform to the transformed coefficient generated by the inverse quantization unit on the basis of the shape of a processing block for inverse orthogonal transform determined by the determination unit and generate a prediction error.

[Claim 4]

An image encoding method for use in an image encoding apparatus, comprising:  
dividing an input image into a plurality of blocks;  
performing prediction on each of the divided blocks using already encoded blocks and generating a prediction error;  
determining a shape of a processing block to be used from a plurality of square or rectangular shapes of processing blocks for orthogonal transform, applying orthogonal transform of the determined shape to the generated prediction error, and generating a transformed coefficient;  
selecting a quantization matrix on the basis of the determined shape of a processing block for orthogonal transform;  
quantizing the transformed coefficient using the selected quantization matrix and generating a quantized coefficient; and  
encoding the generated quantized coefficient.

[Claim 5]

An image decoding method for use in an image decoding apparatus, comprising:  
separating necessary code from an input bit stream;  
determining a shape of a processing block for inverse orthogonal transform from decoded information;

decoding a quantized coefficient from decoded information;  
decoding a quantization matrix used for inverse quantizing the decoded quantized coefficient;  
selecting a quantization matrix on the basis of the determined shape of a processing block for inverse orthogonal transform;  
inverse quantizing the decoded quantized coefficient using the selected quantization matrix and generating a transformed coefficient; and  
applying inverse orthogonal transform to the generated transformed coefficient on the basis of the determined shape of a processing block for inverse orthogonal transform and generating a prediction error.

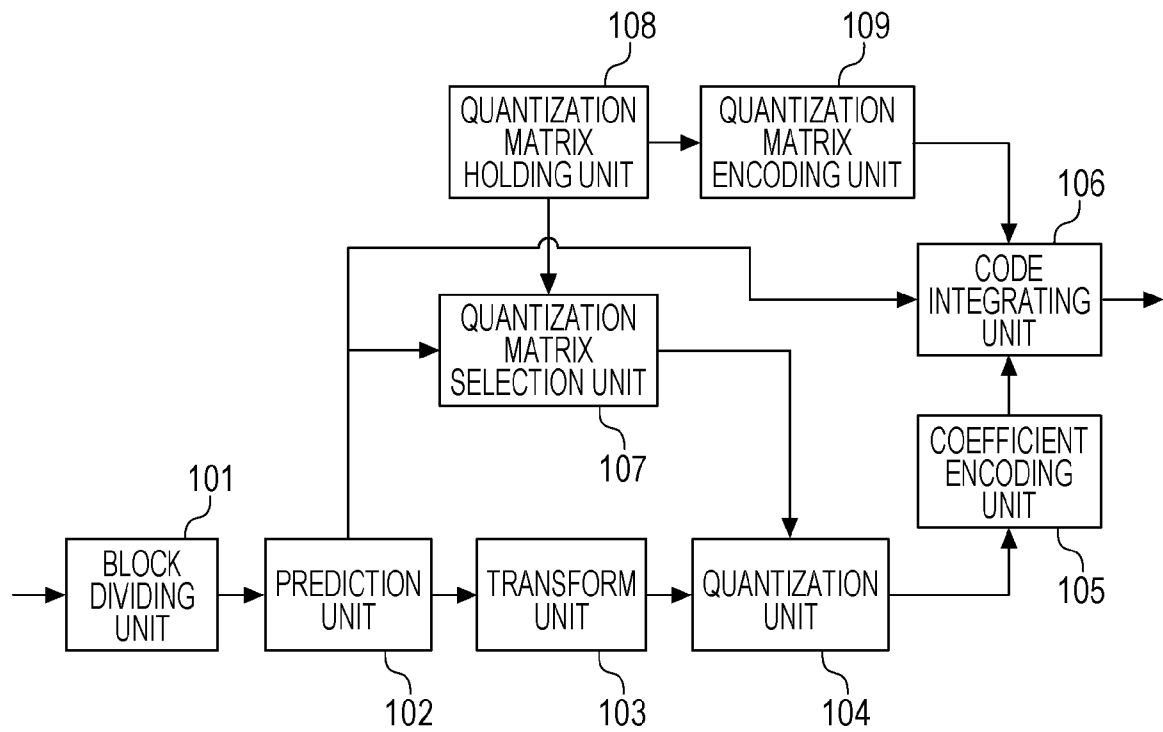
[Claim 6]

A computer-readable program comprising:  
program code for causing a computer to function as the image encoding apparatus according to Claim 1.

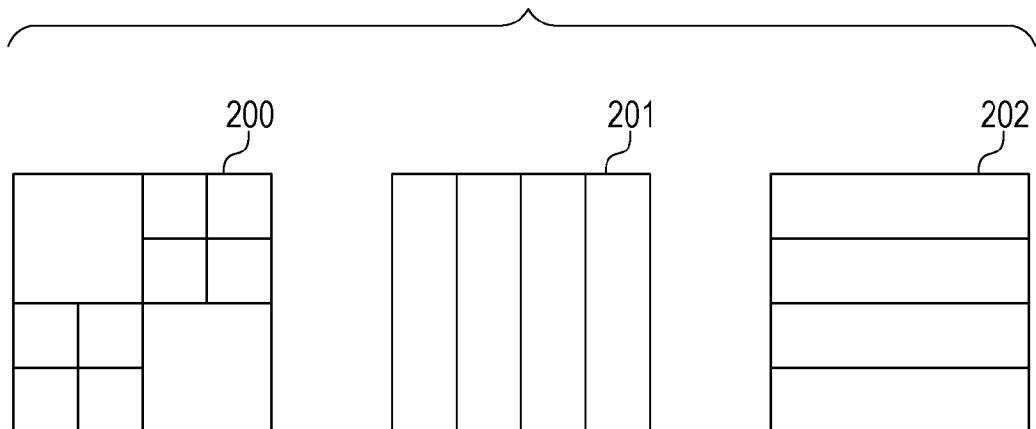
[Claim 7]

A computer-readable program comprising:  
program code for causing a computer to function as the image decoding apparatus according to Claim 3.

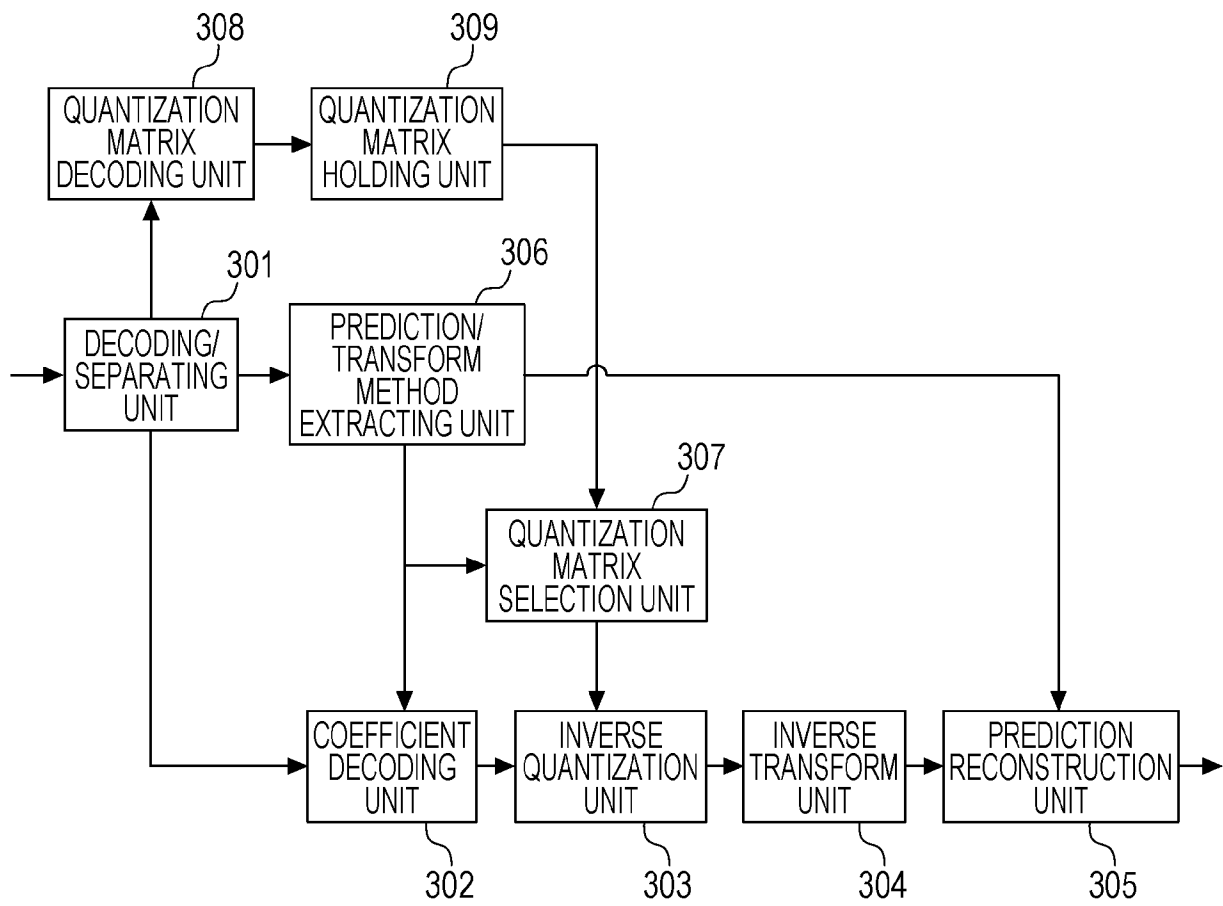
[Fig. 1]



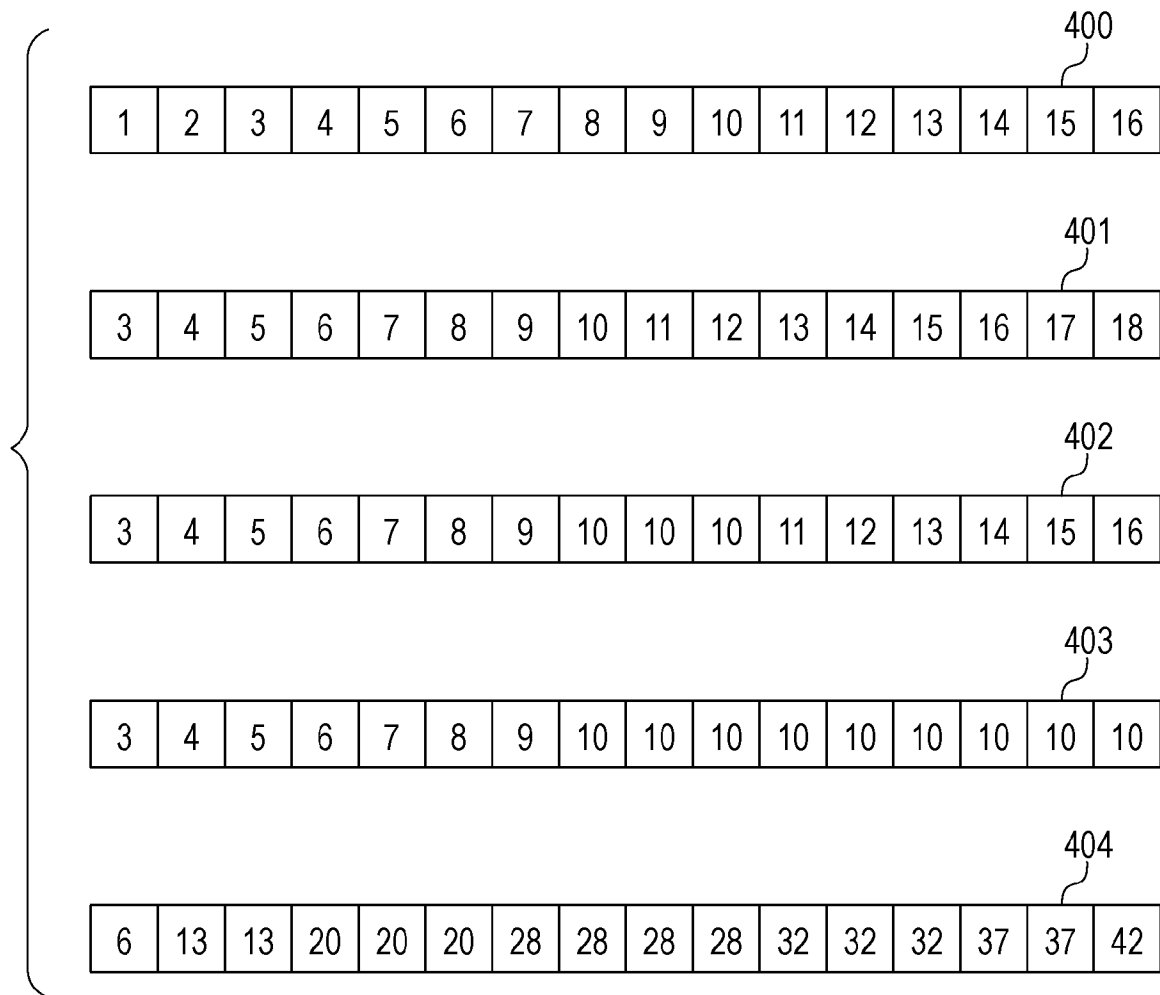
[Fig. 2]



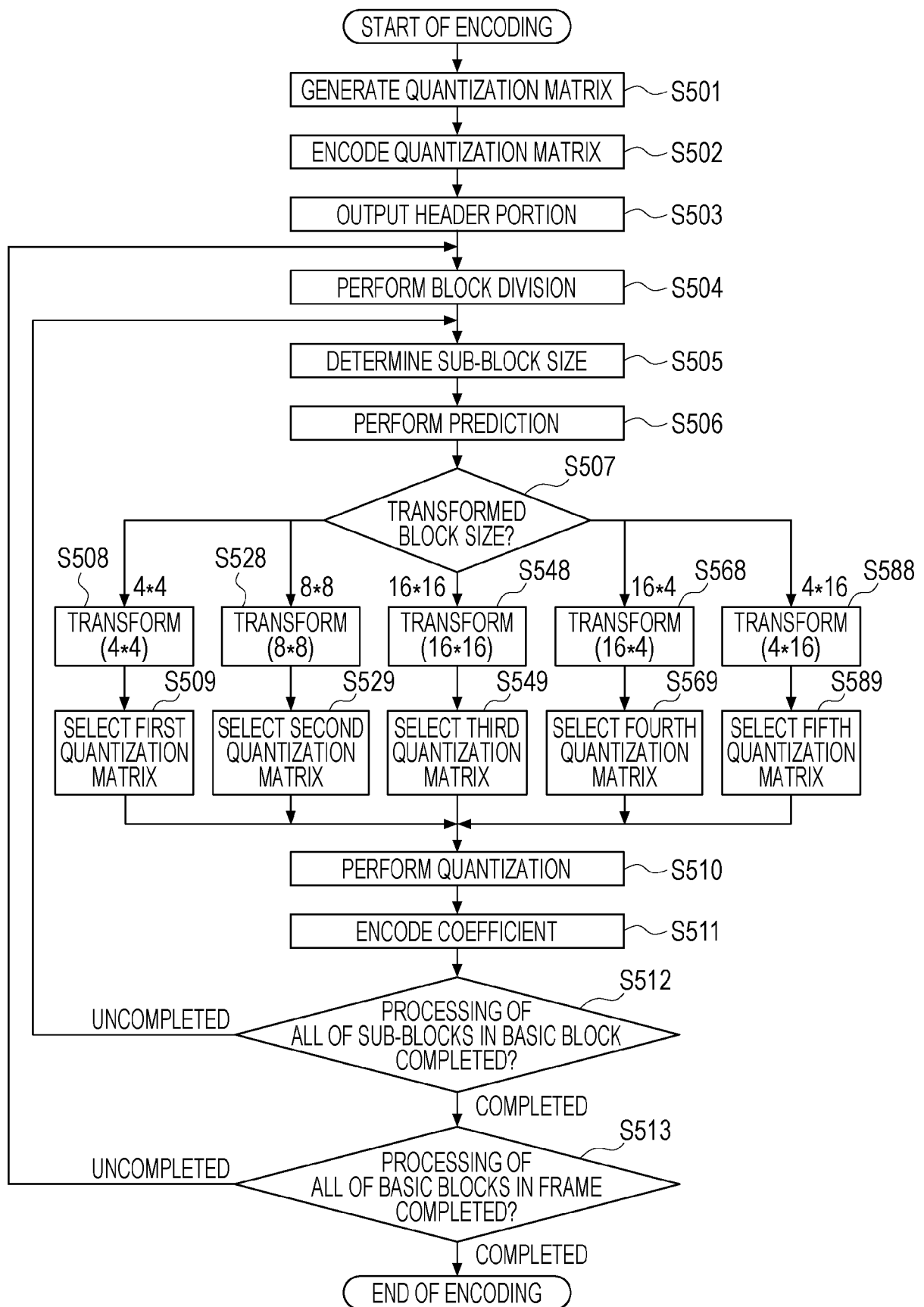
[Fig. 3]



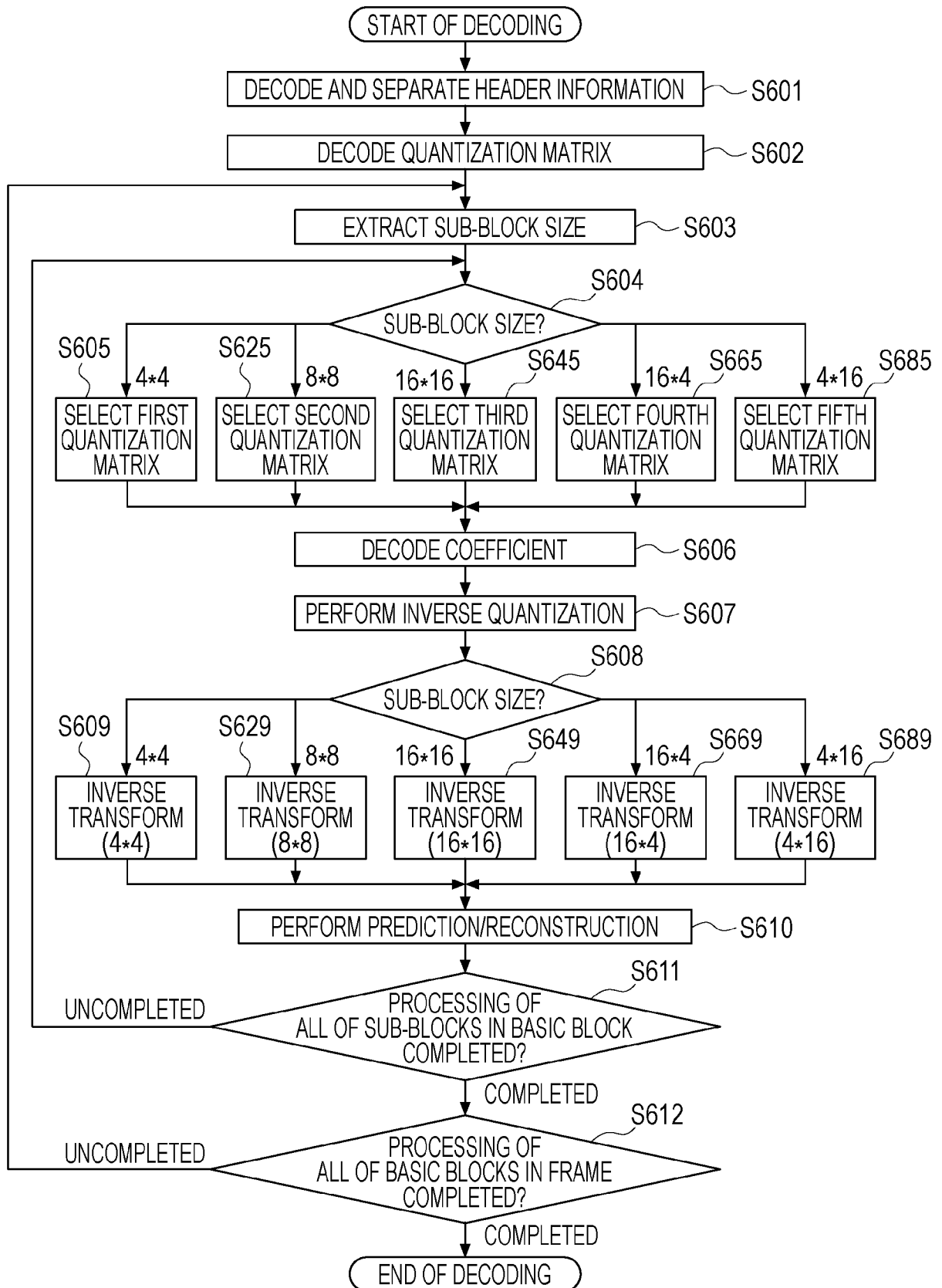
[Fig. 4]



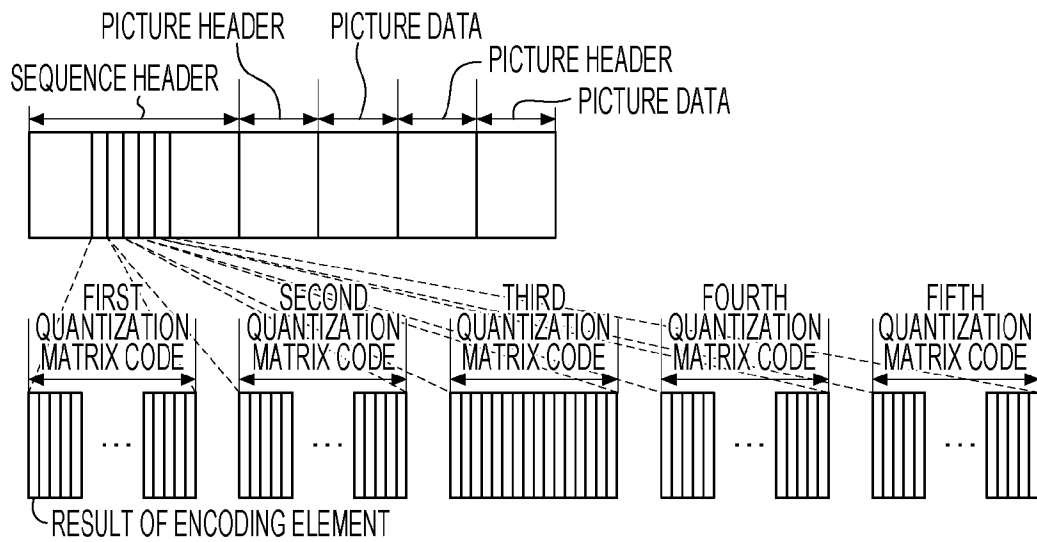
[Fig. 5]



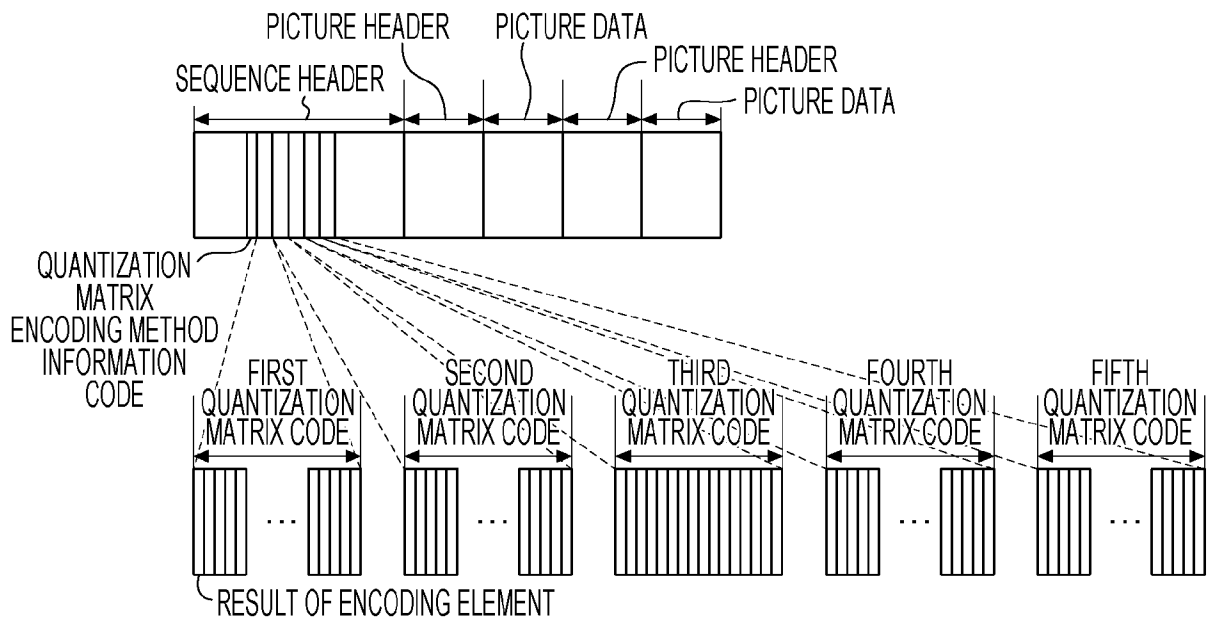
[Fig. 6]



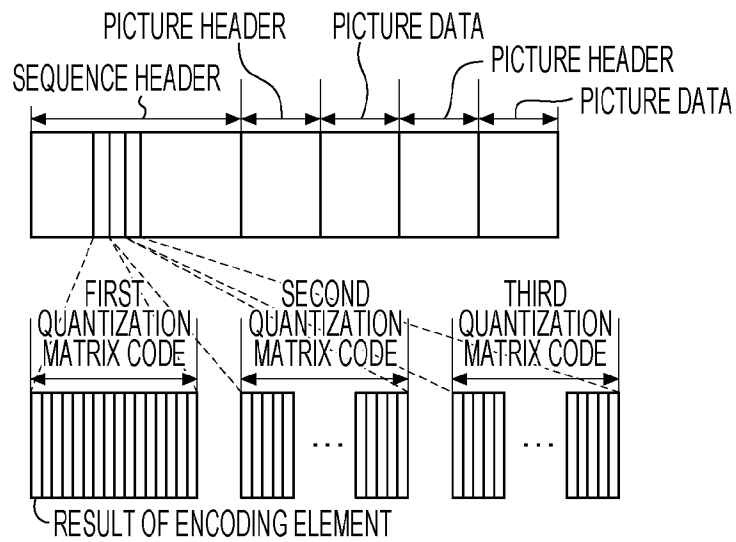
[Fig. 7A]



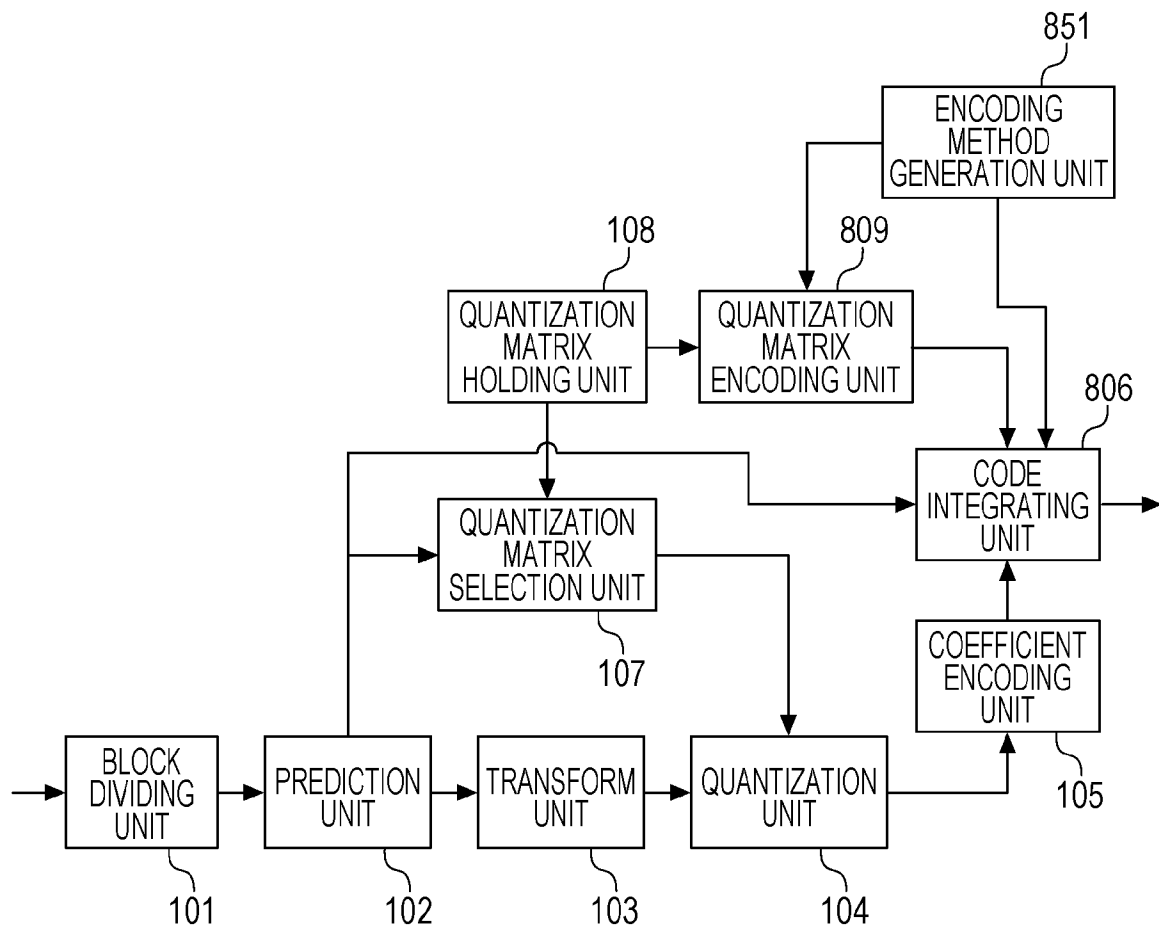
[Fig. 7B]



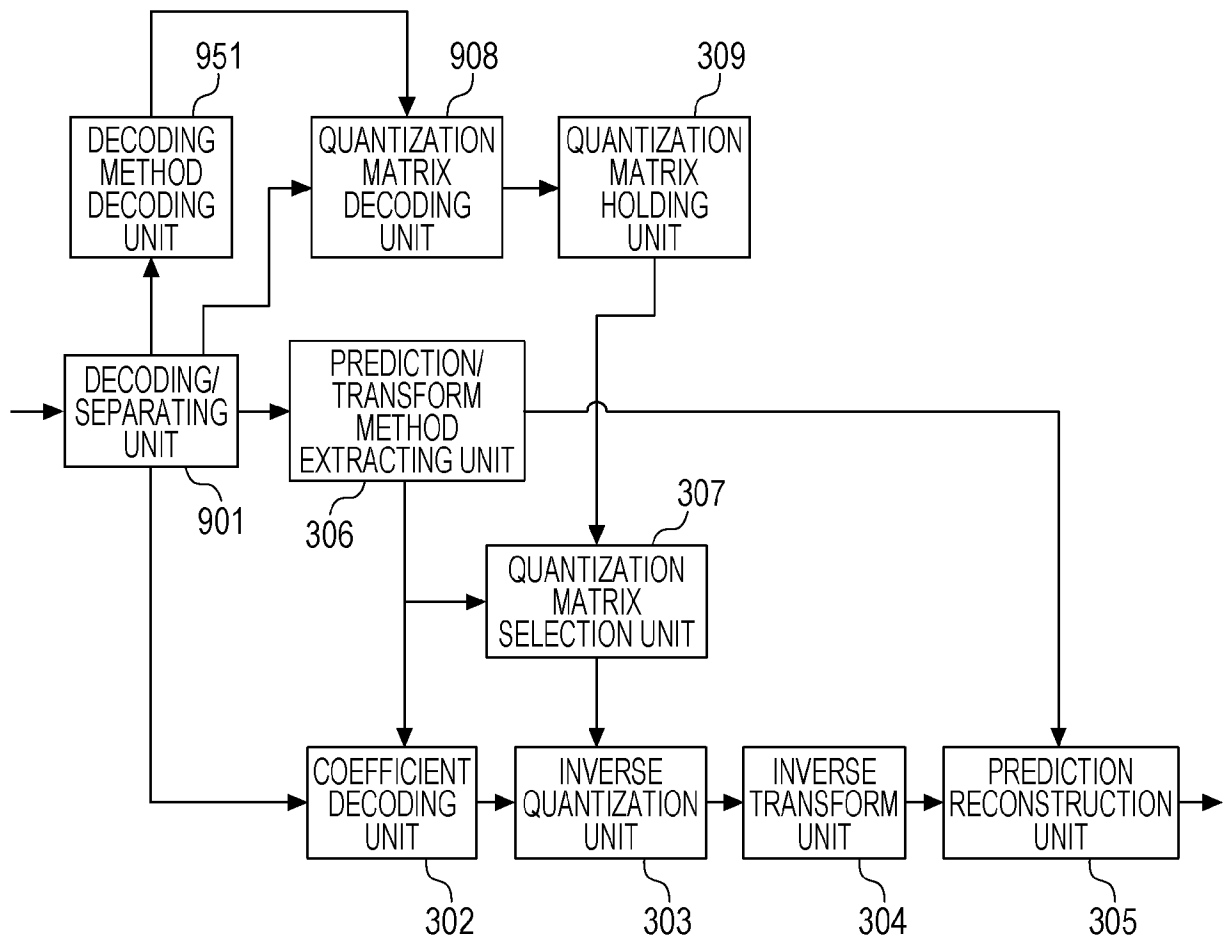
[Fig. 7C]



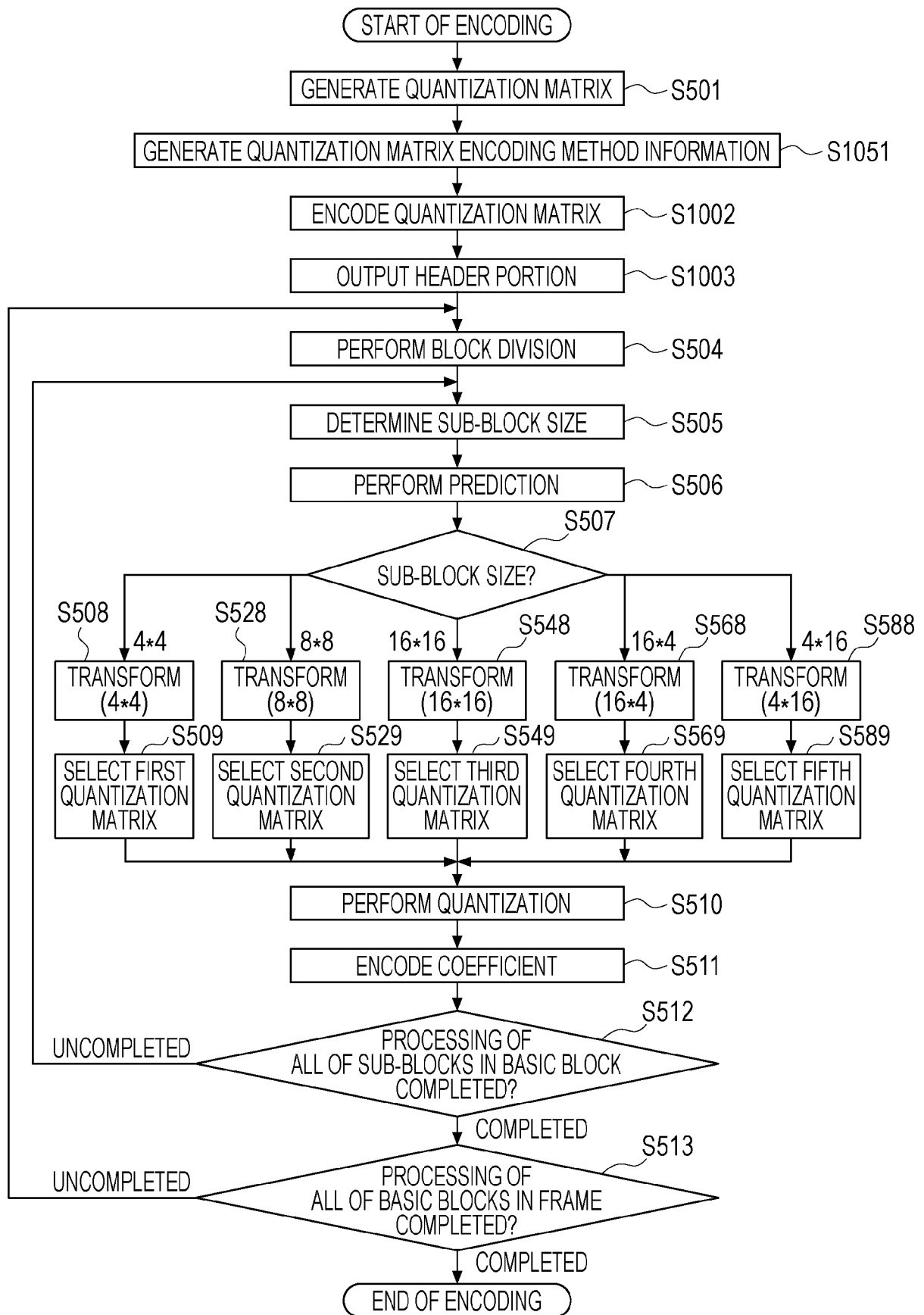
[Fig. 8]



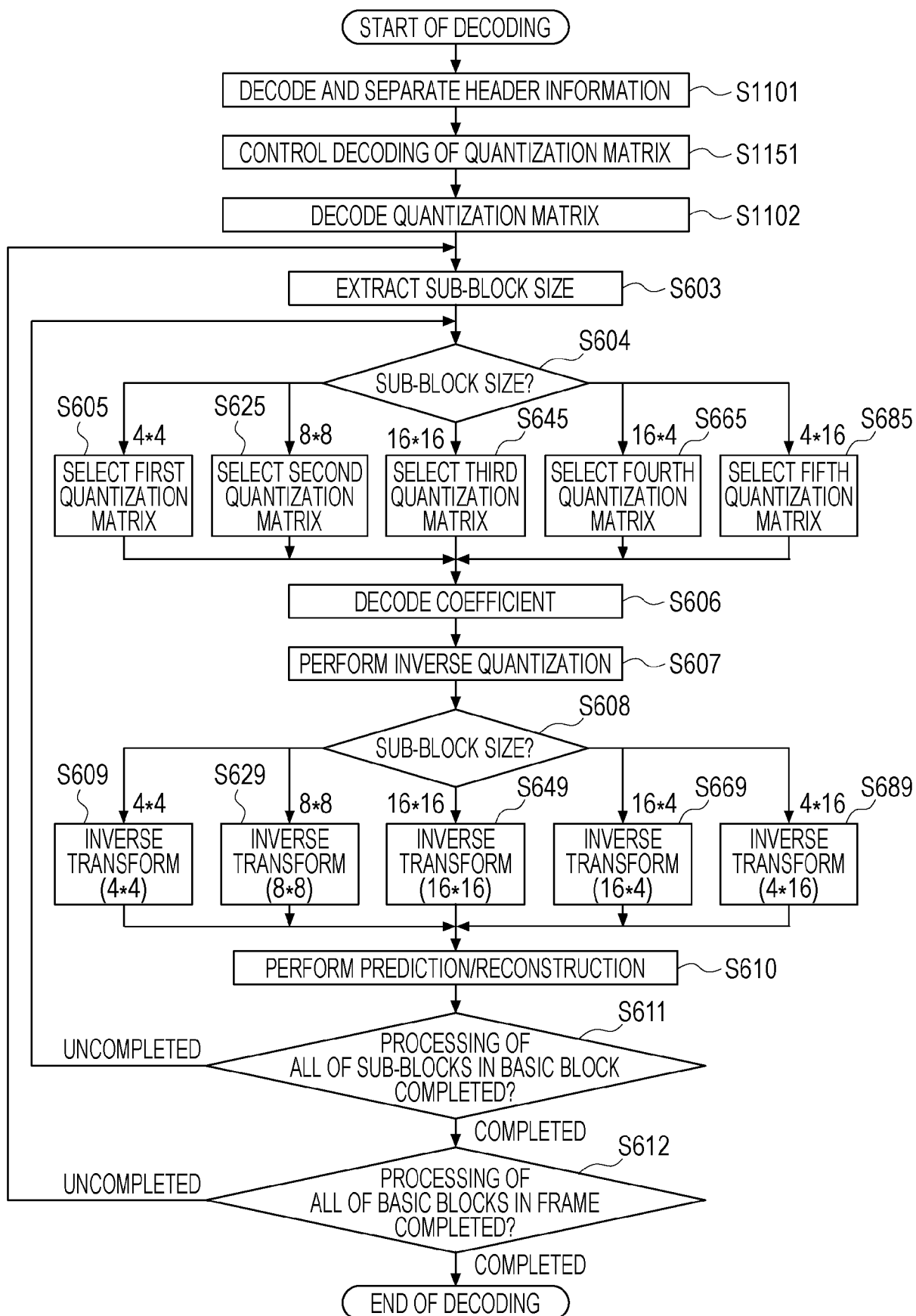
[Fig. 9]



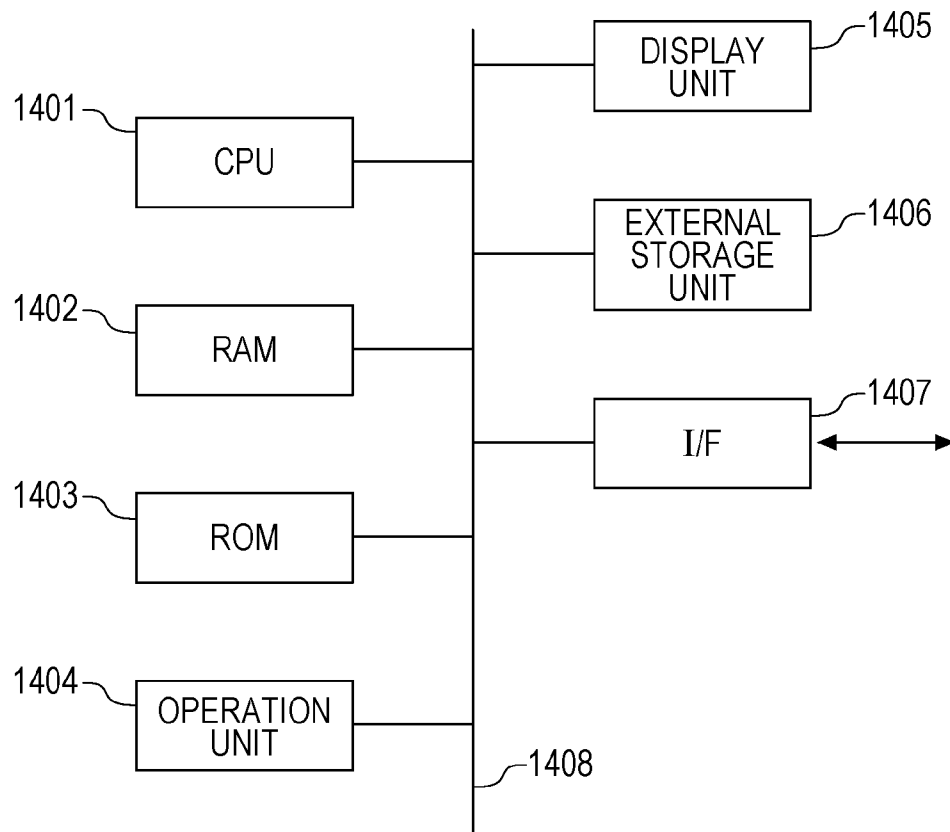
[Fig. 10]



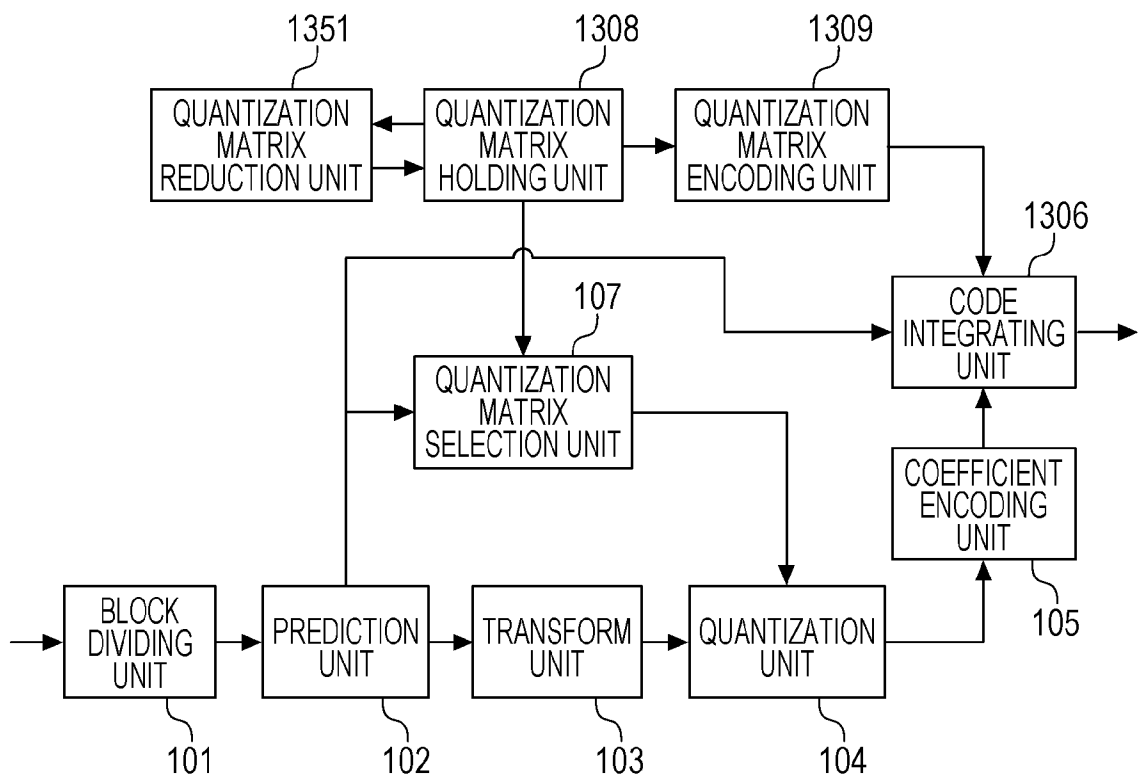
[Fig. 11]



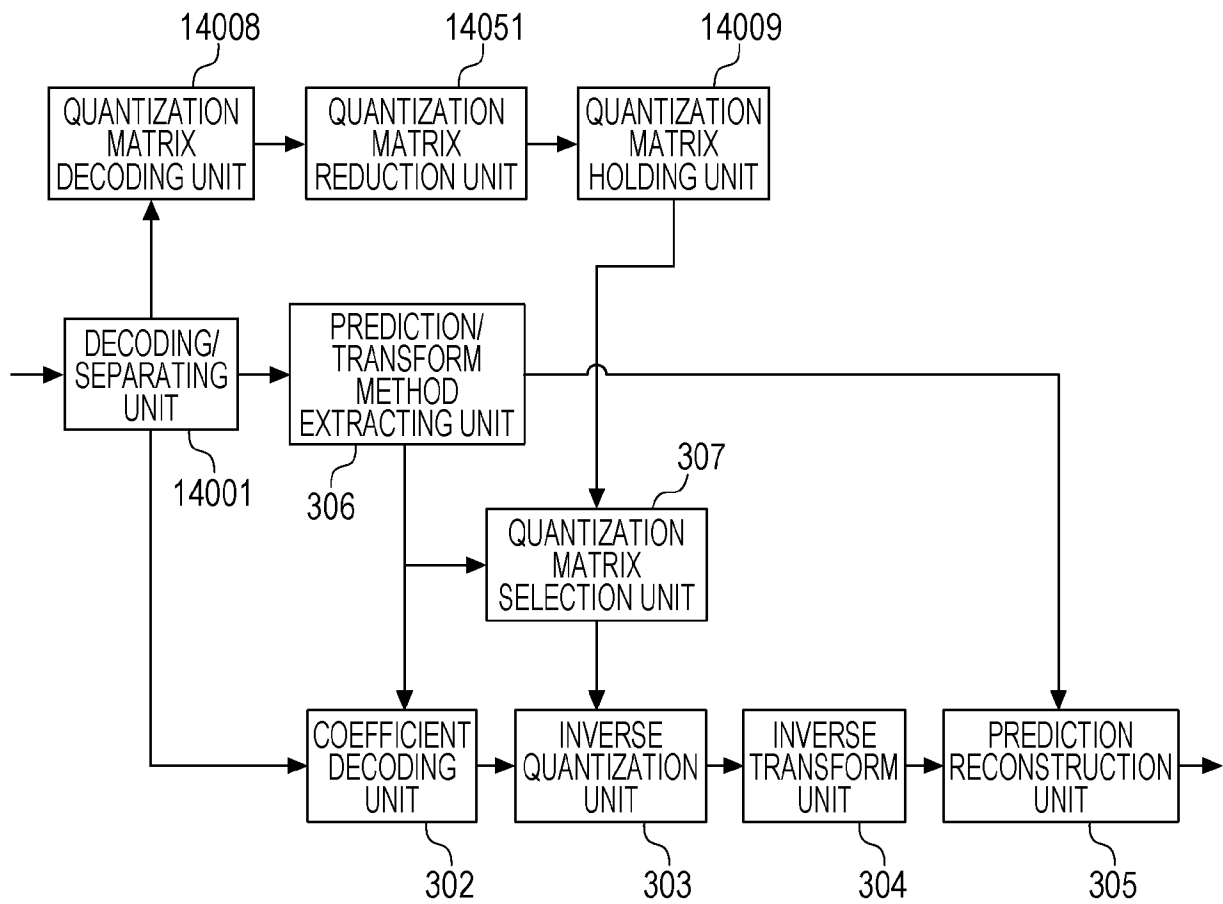
[Fig. 12]



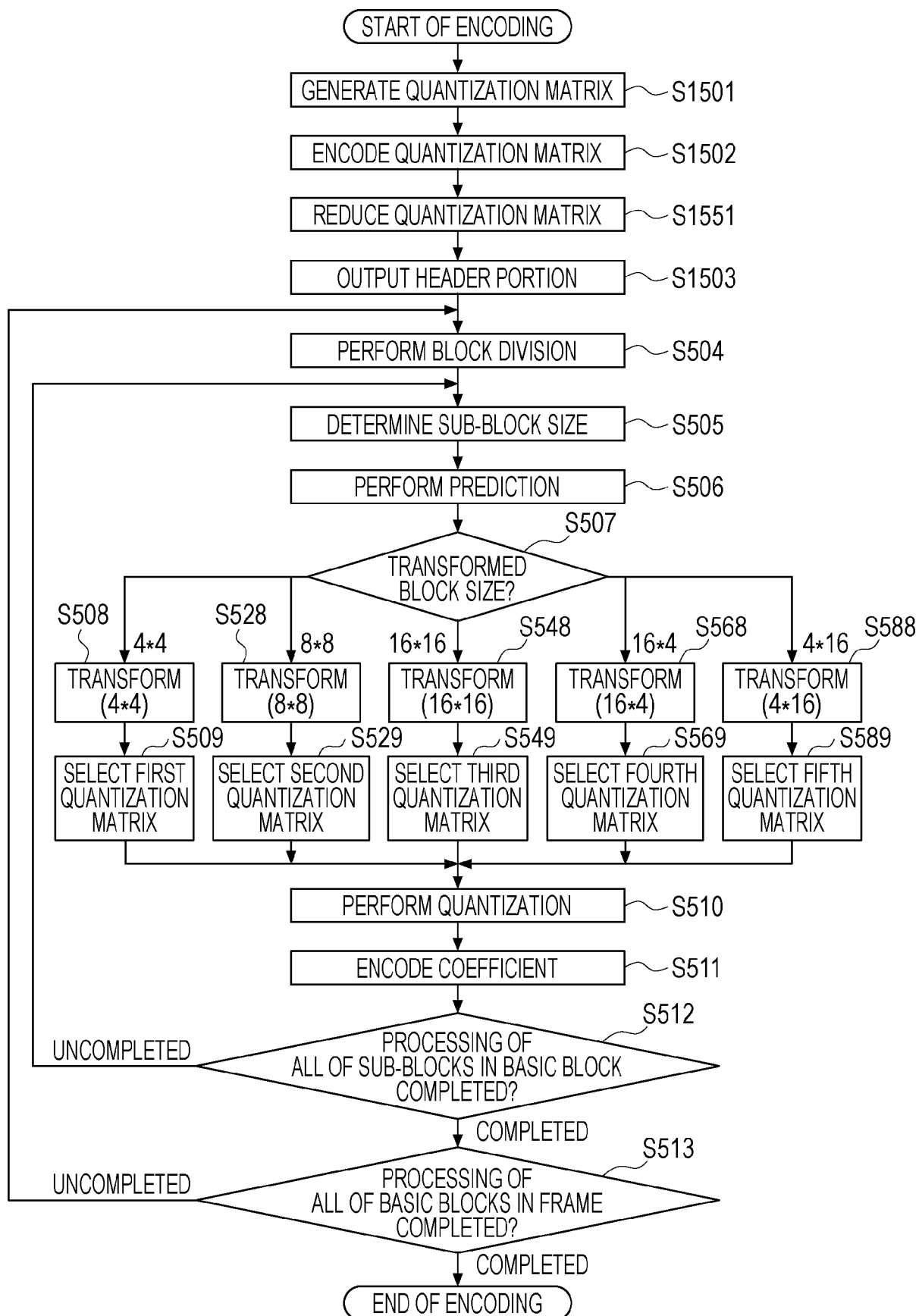
[Fig. 13]



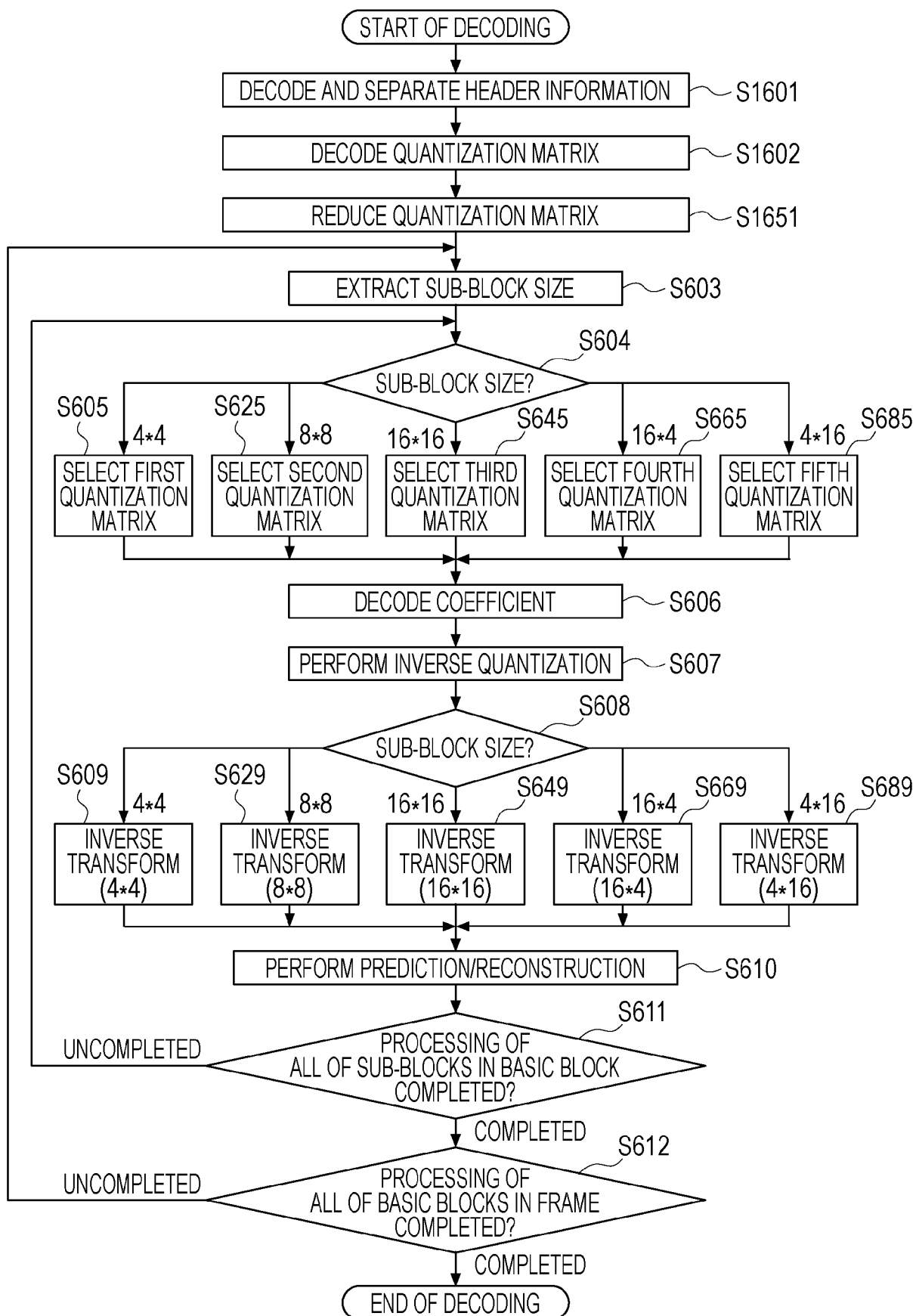
[Fig. 14]



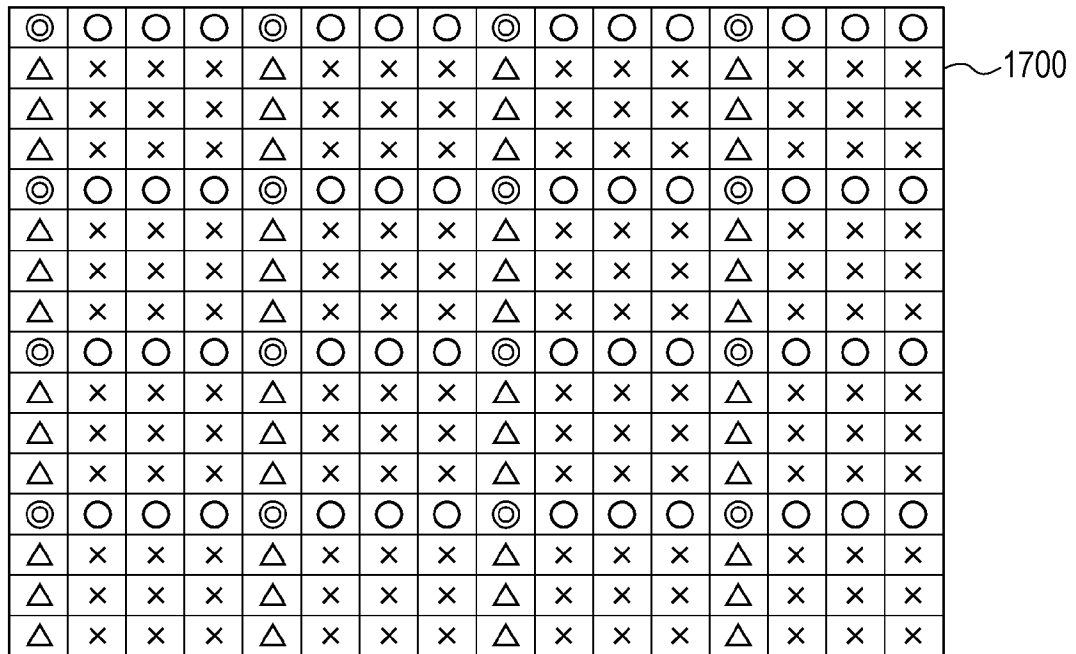
[Fig. 15]



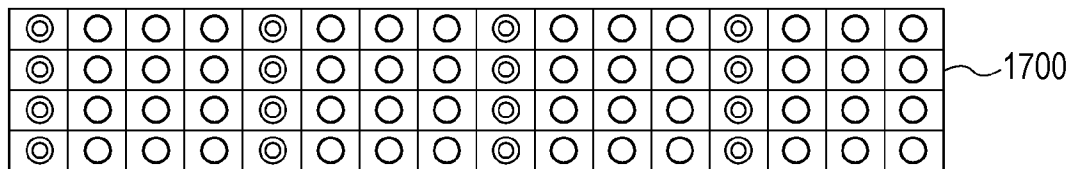
[Fig. 16]



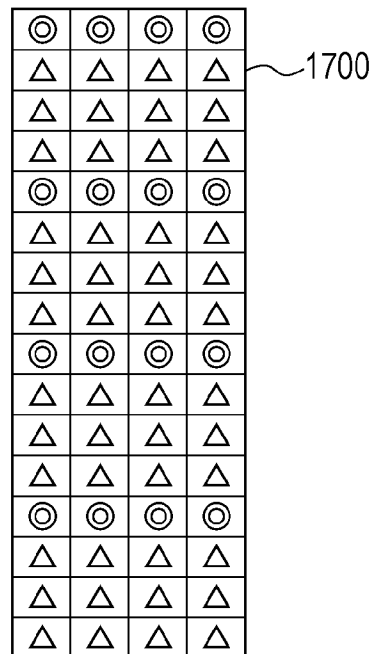
[Fig. 17A]



[Fig. 17B]



[Fig. 17C]



## INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2012/004471

## A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl. H04N7/32 (2006.01) i, H04N7/30 (2006.01) i

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

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl. H04N7/24-7/68

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Published examined utility model applications of Japan 1922-1996  
 Published unexamined utility model applications of Japan 1971-2012  
 Registered utility model specifications of Japan 1996-2012  
 Published registered utility model applications of Japan 1994-2012

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages                                                                                                                                                  | Relevant to claim No. |
|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| Y         | WO 2004/029880 A1 (DOCOMO COMMUNICATIONS LABORATORIES USA, INC.) 2004.04.08, paragraphs [0059]-[0070] & US 2004/0151253 A1 & KR 10-2005-0065548 A & CN 1685369 A                                                                    | 1-7                   |
| Y         | JP 2003-533141 A (ROBERT BOSCH GMBH) 2003.11.05, paragraphs [0021]-[0022] & US 2004/0062309 A1 & WO 2001/086961 A1                                                                                                                  | 1-7                   |
| Y         | Junichi Tanaka, Yoshitaka Morigami, Teruhiko Suzuki, Quantization Matrix for HEVC, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 5th Meeting: Geneva, CH, 2011.03.16, [JCTVC-E073] | 1-7                   |



Further documents are listed in the continuation of Box C.



See patent family annex.

\* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&amp;” document member of the same patent family

Date of the actual completion of the international search

03.10.2012

Date of mailing of the international search report

16.10.2012

Name and mailing address of the ISA/JP

**Japan Patent Office**

3-4-3, Kasumigaseki, Chiyoda-ku, Tokyo 100-8915, Japan

Authorized officer

**Arimitsu Yokota**

Telephone No. +81-3-3581-1101 Ext. 3541

5C

3863

## INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2012/004471

| C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT |                                                                                                                                                                                                                                              |                       |
|-------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| Category*                                             | Citation of document, with indication, where appropriate, of the relevant passages                                                                                                                                                           | Relevant to claim No. |
| P,X                                                   | Ximin Zhang, Shan Liu, Shawmin Lei, Method and syntax for quantization matrices representation, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 7th Meeting: Geneva, 2011.11.21, [JCTVC-G152] | 1-7                   |