

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 989 223**

51 Int. Cl.:

H04N 19/00

(2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **22.06.2012** **E 19220154 (9)**

97 Fecha y número de publicación de la concesión europea: **07.08.2024** **EP 3720127**

54 Título: **Dispositivo de decodificación de desplazamiento, dispositivo de codificación de desplazamiento, dispositivo de filtro de imagen y estructura de datos**

30 Prioridad:

23.06.2011 JP 2011139961

29.09.2011 JP 2011215476

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

25.11.2024

73 Titular/es:

HUAWEI TECHNOLOGIES CO., LTD. (100.0%)
Huawei Administration Building, Bantian,
Longgang District
Shenzhen, Guangdong 518129, CN

72 Inventor/es:

YAMAZAKI, TAKANORI;
IKAI, TOMOHIRO;
YAMAMOTO, TOMOYUKI y
YASUGI, YUKINOBU

74 Agente/Representante:

LEHMANN NOVO, María Isabel

ES 2 989 223 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Dispositivo de decodificación de desplazamiento, dispositivo de codificación de desplazamiento, dispositivo de filtro de imagen y estructura de datos

CAMPO TÉCNICO

La presente invención se refiere a un dispositivo de filtrado de imagen que realiza el filtrado en imágenes. La invención se refiere, además, a un dispositivo de decodificación de desplazamiento que decodifica desplazamientos referidos por un filtro de imagen, y un dispositivo de codificación de desplazamiento que codifica desplazamientos referidos por un filtro de imagen. La invención también se refiere a una estructura de datos de datos codificados.

ANTECEDENTES DE LA TÉCNICA

Actualmente, se utiliza un dispositivo de codificación de vídeo (dispositivo de codificación), que genera datos codificados mediante la codificación de imágenes de vídeo, con el fin de transmitir o grabar imágenes de vídeo de forma eficiente y, un dispositivo de decodificación de vídeo (dispositivo de decodificación) que genera imágenes decodificadas mediante la decodificación de los datos codificados. Ejemplos específicos de métodos de codificación de vídeo son un método definido en H. 264/MPEG-4. AVC, un método utilizado en el software KTA, que es un códec de desarrollo conjunto en VCEG (Grupo de Expertos en Codificación de Vídeo), un método utilizado en el software TMuC (Modelo de Prueba bajo Consideración), que es un códec sucesor del códec que se utiliza en el software KTA y un método utilizado en el software HM (HEVC TestModel).

En dichos métodos de codificación, las imágenes (pictures) que forman imágenes de vídeo se gestionan en una estructura jerárquica que está constituida por cortes obtenidos al dividir una imagen, unidades de codificación mayor (LCU: Largest Coding Unit, también denominada bloque en árbol), obtenidas mediante la división de un corte, unidades de codificación (CU: Coding Unit, también denominada nodo de codificación), que se obtienen al dividir una unidad de codificación mayor y, bloques y particiones obtenidas mediante la división de una unidad de codificación. En muchos casos, las imágenes se codifican utilizando bloques como la unidad de codificación más pequeña.

Además, en tales métodos de codificación, normalmente, una imagen de predicción se genera sobre la base de una imagen decodificada localmente, obtenida mediante la codificación y decodificación de una imagen de entrada y se codifican los datos de diferencia que indican una diferencia entre la imagen de predicción y la imagen de entrada. Como un método de generación de imágenes de predicción, se conocen la predicción inter-tramas (predicción inter) y la predicción intra-tramas (predicción intra).

En la predicción intra, sobre la base de una imagen decodificada, de forma local, dentro de la misma trama, se generan, de forma secuencial, imágenes de predicción en esta trama. Más específicamente, en la predicción intra, normalmente, para cada unidad de predicción (a modo de ejemplo, un bloque), se selecciona una de las direcciones de predicción (modos de predicción) incluidas en un grupo de dirección de predicción predeterminado y, además, el valor de píxel de un píxel de referencia en una imagen decodificada localmente se extrapola para la dirección de predicción seleccionada, con lo que se genera un valor de píxel de predicción en una zona sujeta a predicción. Por otro lado, en la predicción inter, mediante la aplicación del desplazamiento de movimiento utilizando vectores de movimiento para una imagen de referencia, dentro de una trama de referencia completamente decodificada (imagen decodificada), se genera una imagen de predicción dentro de una trama que ha de predecirse para cada unidad de predicción (a modo de ejemplo, un bloque).

NPL 1 y NPL 2 dan a conocer un filtro de desplazamiento adaptativo dispuesto en una etapa posterior a un filtro de desbloqueo que reduce la distorsión de bloque de una imagen decodificada, y en una etapa previa a un filtro de bucle adaptativo (también denominado como un "filtro adaptativo") que realiza el procesamiento de filtrado utilizando un coeficiente de filtro determinado de forma adaptativa. Este filtro de desplazamiento adaptativo añade un desplazamiento, establecido de forma adaptativa, al valor de píxel de cada píxel de una salida de imagen procedente del filtro de desbloqueo.

Al proporcionar dicho filtro de desplazamiento adaptativo, es posible suprimir la distorsión de bloque de forma más efectiva.

Lista de referencias

Información no de patente

NPL 1: "JCTVC-D122", Equipo de colaboración conjunto sobre codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 4ª reunión: Daegu, KR, 01/2011.

NPL 2: "JCTVC-E049", Equipo de colaboración conjunto sobre codificación de vídeo (JCT-VC) de ITU-T SG16 WP3

e ISO/IEC JTC1/SC29/WG11, 5ª reunión: Ginebra, CH, 03/2011.

SUMARIO DE LA INVENCION

5 Problema técnico

Sin embargo, no se establece el margen de valores de un desplazamiento utilizado en un filtro de desplazamiento conocido y, por lo tanto, el número de bits de un desplazamiento es grande, lo que requiere un gran tamaño de memoria para la memorización de un desplazamiento.

10 La presente invención se ha realizado en vista del problema descrito anteriormente. Un objeto de la presente invención es realizar un dispositivo de filtrado de imagen que sea capaz de reducir la distorsión de bloque mientras suprime un aumento en el tamaño de la memoria.

15 SOLUCIÓN AL PROBLEMA

La presente invención proporciona un dispositivo de filtro de imagen y un método de filtrado de conformidad con las reivindicaciones adjuntas. La invención se define mediante las formas de realización 1 y 3 tal como se presenta a continuación. El contenido presentado como las formas de realización 2, 4-7 no define la invención, aunque es útil para comprender la invención.

20 Con la configuración descrita anteriormente, es posible hacer que un filtro de imagen realice un procesamiento de filtrado de desplazamiento adecuado al mismo tiempo que se reduce el tamaño de la memoria de una memoria para almacenar desplazamientos.

25 BREVE DESCRIPCIÓN DE LOS DIBUJOS

La Figura 1 es un diagrama de bloques que ilustra la configuración de un filtro de desplazamiento adaptativo de conformidad con una primera forma de realización de la presente invención.

30 La Figura 2 ilustra una estructura de datos de datos codificados generados por un dispositivo de codificación de vídeo, de conformidad con la primera forma de realización de la presente invención, y decodificados por un dispositivo de decodificación de vídeo de conformidad con la primera forma de realización de la presente invención: partes (a) a (d) que ilustran, respectivamente, una capa de imagen, una capa de corte, una capa de bloque de árbol y una capa CU; la parte (e) ilustra la configuración de información de QAOU con respecto a un QAOU que no es una hoja del árbol; y la parte (f) ilustra la configuración de información de QAOU con respecto a un QAOU que es una hoja de árbol.

40 La Figura 3 ilustra cada sintaxis incluida en información de desplazamiento OI con respecto a los datos codificados, de conformidad con la primera forma de realización de la presente invención.

45 La Figura 4 ilustra modos divididos de unidades de desplazamiento de conformidad con la primera forma de realización de la presente invención: la parte (a) muestra un modo dividido cuando `sao_curr_depth=0`; la parte (b) muestra un modo dividido cuando `sao_curr_depth=1`; la parte (c) muestra un modo dividido cuando `sao_curr_depth=2`; la parte (d) muestra un modo dividido cuando `sao_curr_depth=3`; y la parte (e) muestra un modo dividido cuando `sao_curr_depth=4`.

50 La Figura 5 es un diagrama de bloques que ilustra la configuración del dispositivo de decodificación de vídeo de conformidad con la primera forma de realización de la presente invención.

55 La Figura 6 ilustra la primera forma de realización de la presente invención: la parte (a) ilustra QAOMUs que tienen una profundidad de división de tres y forman una unidad objeto de procesamiento e índices de QAOU asignados a QAOMUs; y la parte (b) ilustra tipos de desplazamiento asociados con índices QAOU de 0 a 9, y desplazamientos con respecto a clases individuales que se pueden seleccionar para cada tipo de desplazamiento.

60 La Figura 7 ilustra ejemplos de números QAOMU que se adjuntan a QAOMUs que se incluyen en una unidad objeto de procesamiento: la parte (a) ilustra un número QAOMU asignado a un QAOMU que tiene una profundidad de división de 0; la parte (b) ilustra números QAOMU asignados a QAOMUs que tienen una profundidad de división de uno; la parte (c) ilustra números QAOMU asignados a QAOMUs que tienen una profundidad de división de dos; la parte (d) ilustra números QAOMU asignados a QAOMUs que tienen una profundidad de división de tres; y la parte (e) ilustra números QAOMU asignados a QAOMUs que tienen una profundidad de división de cuatro.

65 La Figura 8 es un diagrama de flujo de un flujo de procesamiento realizado por un procesador de filtro de desplazamiento adaptativo de conformidad con la primera forma de realización de la presente invención.

La Figura 9 ilustra, junto con las profundidades de bits de pixels, ejemplos de profundidades de bits de

desplazamiento y valores de desplazamiento establecidos por una sección de configuración de atributo de desplazamiento, incluida en un filtro adaptativo, de conformidad con la primera forma de realización de la presente invención: las partes (a) a (d) ilustran, respectivamente, ejemplos asociados con los patrones S1 a S4; y la parte (e) muestra un caso en el que STEP=2 en la parte (d).

5 La Figura 10 ilustra el procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo de conformidad con la primera forma de realización de la presente invención: las partes (a) a (d) ilustran los pixels a los que se hace referencia cuando sao_type_idx=1 a 4, respectivamente.

10 La Figura 11 ilustra el procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo de conformidad con la primera forma de realización de la presente invención: la parte (a) muestra gráficos que indican las relaciones de magnitud entre un valor de píxel $\text{pic}[x]$, de un píxel objeto x , y el valor de píxel de un píxel, a o b , y muestra, además, los valores de una función Sign de conformidad con la relación de magnitud; la parte (b) muestra gráficos que indican las relaciones de magnitud entre el valor de píxel del píxel objeto x y cada uno de los valores de píxel del píxel a , y el píxel b , y muestra, además, los valores de EdgeType de conformidad con la relación de magnitud; la parte (c) muestra la asociación entre cada gráfico que se muestra en la parte (b) y class_idx; y la parte (d) ilustra una tabla de transformación que indica la transformación desde EdgeType a class_idx.

20 La Figura 12 ilustra un procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo, de conformidad con la primera forma de realización de la presente invención: la parte (a) muestra, de forma esquemática, la clasificación realizada cuando sao_type_idx=5; la parte (b) muestra, esquemáticamente, la clasificación realizada cuando sao_type_idx=6; y la parte (c) muestra una tabla que indica un ejemplo de clasificación realizada cuando se especifica un desplazamiento de banda.

25 La Figura 13 ilustra un procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo, de conformidad con la primera forma de realización de la presente invención, y muestra una tabla que indica otro ejemplo de clasificación que se realiza cuando se especifica un desplazamiento de banda.

30 La Figura 14 es un diagrama de bloques que ilustra la configuración de un dispositivo de codificación de vídeo de conformidad con la primera forma de realización de la presente invención.

La Figura 15 es un diagrama de bloques que ilustra la configuración de un filtro de desplazamiento adaptativo incluido en el dispositivo de codificación de vídeo, de conformidad con la primera forma de realización de la presente invención.

35 La Figura 16 es un diagrama de flujo de un flujo de procesamiento realizado por un calculador de desplazamiento del filtro de desplazamiento adaptativo incluido en el dispositivo de codificación de vídeo, de conformidad con la primera forma de realización de la presente invención.

40 La Figura 17 es un diagrama de flujo de un flujo de procesamiento realizado por un selector de información de desplazamiento del filtro de desplazamiento adaptativo incluido en el dispositivo de codificación de vídeo de conformidad con la primera forma de realización de la presente invención.

45 La Figura 18 ilustra un procesamiento realizado por el selector de información de desplazamiento del filtro de desplazamiento adaptativo incluido en el dispositivo de codificación de vídeo, de conformidad con la primera forma de realización de la presente invención: la parte (a) muestra un modo de división cuando la profundidad de división es 0 y 1; la parte (b) muestra un modo de división cuando la profundidad de división es 1; la parte (c) muestra un modo de división cuando la profundidad de división es 2; y la parte (d) muestra un ejemplo de modo de división determinado por el selector de información de desplazamiento.

50 La Figura 19 es un diagrama de bloques que ilustra la configuración de un filtro de desplazamiento adaptativo incluido en un dispositivo de decodificación de vídeo de conformidad con una segunda forma de realización de la presente invención.

55 La Figura 20 ilustra el filtro de desplazamiento adaptativo, de conformidad con la segunda forma de realización de la presente invención: la parte (a) ilustra un primer ejemplo específico de una función merge_tbl[sao_type_idx]; y la parte (b) ilustra un segundo ejemplo específico de la función merge_tbl[sao_type_idx].

60 La Figura 21 es un diagrama de bloques que ilustra la configuración de un filtro de desplazamiento adaptativo incluido en un dispositivo de codificación de vídeo, de conformidad con la segunda forma de realización de la presente invención.

La Figura 22 es un diagrama de bloques que ilustra la configuración de un filtro de desplazamiento adaptativo de conformidad con una tercera forma de realización de la presente invención.

65 La Figura 23 ilustra la sintaxis de información de desplazamiento OI de datos codificados de conformidad con la

tercera forma de realización.

La Figura 24 ilustra la tercera forma de realización: la parte (a) muestra QAOMUs que tienen una profundidad de división de tres y forman una unidad objeto de procesamiento y muestra, además, índices de QAOU asignados a QAOMUs; y la parte (b) muestra un tipo de desplazamiento asociado con cada uno de los índices QAOU de 0 a 9 y desplazamientos con respecto a clases individuales que se pueden seleccionar para cada tipo de desplazamiento.

La Figura 25 ilustra tablas de transformación utilizadas por la sección de decodificación de información de desplazamiento de conformidad con la tercera forma de realización.

La Figura 26 ilustra un procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo, de conformidad con la tercera forma de realización: la parte (a) muestra gráficos que indican las relaciones de magnitud entre el valor de píxel $\text{pix}[x]$ de un píxel objeto x , y el valor del píxel de un píxel a o b y muestra, además, los valores de una función Sign de conformidad con la relación de magnitud; la parte (b) muestra gráficos que indican las relaciones de magnitud entre el valor de píxel del píxel objeto x , y cada uno de los valores de píxel del píxel a , y el píxel b , y muestra, además, el valor de EdgeType de conformidad con la relación de magnitud; la parte (c) muestra la asociación entre cada gráfico que se ilustra en la parte (b) y `class_idx`; y las partes (d) a (f) ilustran tablas de transformación utilizadas para la transformación desde EdgeType a `class_idx`.

La Figura 27 ilustra un procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo de conformidad con la tercera forma de realización: la parte (a) muestra, de forma esquemática, la clasificación realizada cuando `"sao_type_idx=5"`; y la parte (b) muestra, esquemáticamente, la clasificación realizada cuando `"sao_type_idx=6"`.

La Figura 28 ilustra un procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo, de conformidad con la tercera forma de realización: la parte (a) muestra, esquemáticamente, la clasificación realizada cuando la profundidad jerárquica de un QAOU objeto es menor que un umbral; y la parte (b) muestra, de forma esquemática, la clasificación realizada cuando la profundidad jerárquica de un QAOU objeto es igual o mayor que el umbral.

La Figura 29 ilustra un ejemplo de clasificación realizada cuando se especifica un desplazamiento de banda: la parte (a) muestra un ejemplo de clasificación realizada cuando la profundidad jerárquica de un QAOU objeto es menor que un umbral; y la parte (b) muestra un ejemplo de clasificación realizada cuando la profundidad jerárquica de un QAOU objeto es igual o mayor que el umbral.

La Figura 30 es un diagrama de bloques que ilustra la configuración de un filtro de desplazamiento adaptativo 80, incluido en un dispositivo de codificación de vídeo, de conformidad con la tercera forma de realización.

La Figura 31 muestra una visión general de un caso en el que se calcula un error cuadrático con respecto a un QAOU que tiene un índice QAOU "x" para cada tipo de desplazamiento en la tercera forma de realización.

La Figura 32 ilustra la configuración en la que EO y BO se conmutan, de conformidad con el valor de píxel, en una cuarta forma de realización de la presente invención: la parte (a) muestra una visión general de la configuración en la que se conmutan EO y BO de conformidad con el valor de píxel; la parte (b) muestra valores específicos que han de conmutarse; y la parte (c) muestra el contenido de una memoria de lista que se memoriza en la sección de memorización de información de desplazamiento 621.

La Figura 33 ilustra un caso en el que el tipo EO está restringido a la dirección horizontal, en una quinta forma de realización de la presente invención: la parte (a) muestra una visión general de las ventajas; la parte (b) muestra las posiciones de píxeles en la dirección horizontal; la parte (c) muestra un estado diferente al estado en la parte (b); y la parte (d) muestra una visión general de las ventajas obtenidas por el estado en la parte (c).

La Figura 34 ilustra un caso en el que el tipo EO está restringido a la dirección horizontal, en la quinta forma de realización de la presente invención: las partes (a) y (b) muestran casos en los que los píxeles de referencia están situados de forma asimétrica; y la parte (c) muestra una visión general de bordes horizontales.

La Figura 35 muestra una visión general de un caso en el que se mejora la precisión de desplazamiento en una sexta forma de realización de la presente invención.

La Figura 36 ilustra una visión general de un caso en el que se realiza una clasificación más detallada en la sexta forma de realización: la parte (a) muestra una tabla de transformación; y las partes (b) a (d) muestran la clasificación.

La Figura 37 ilustra un caso en el que la clasificación se realiza de conformidad con la crominancia en la sexta forma de realización; la parte (a) muestra la clasificación realizada considerando el valor de píxel de un color acromático; la parte (b) muestra una clasificación en la que dos márgenes de valores están situados, de forma asimétrica, con respecto a los valores de píxeles de un color acromático; las partes (c) y (d) muestran un caso en el que la

clasificación se realiza de forma distinta, de conformidad con el canal de crominancia (Cr o Cb); y la parte (e) muestra un caso en el que la clasificación se realiza utilizando solamente un tipo BO.

5 La Figura 38 es un diagrama de bloques que ilustra la configuración de una sección de decodificación de información de desplazamiento de conformidad con una séptima forma de realización de la presente invención.

La Figura 39 ilustra una visión general de un caso en el que existen clases en las que los pixels no están clasificados, en la séptima forma de realización.

10 La Figura 40 ilustra la sintaxis que ha de utilizarse cuando se usa un indicador de candidato de predicción en la séptima forma de realización.

La Figura 41 es un diagrama de bloques que ilustra la configuración de una sección de decodificación de información de desplazamiento de conformidad con la quinta forma de realización.

15 La parte (a) de la Figura 42 es un diagrama de bloques que ilustra la configuración de un selector de tipo de desplazamiento de conformidad con la quinta forma de realización; y la parte (b) es un diagrama de bloques que ilustra la configuración de otro selector de tipo de desplazamiento.

20 La Figura 43 es un diagrama de bloques que ilustra la configuración de una sección de clasificación de conformidad con la quinta forma de realización.

25 La Figura 44 ilustra la sintaxis de información de desplazamiento, y la sintaxis de información de QAOU, de conformidad con la tercera forma de realización: la parte (a) ilustra la sintaxis de información de desplazamiento; la parte (b) ilustra la sintaxis de información de QAOU; y la parte (c) ilustra la sintaxis completa de un filtro de desplazamiento adaptativo que considera la sintaxis de la parte (a) y la sintaxis de la parte (b).

30 La Figura 45 ilustra un caso en el que un dispositivo de decodificación de vídeo y un dispositivo de codificación de vídeo se pueden utilizar para transmitir y recibir imágenes de vídeo; la parte (a) es un diagrama de bloques que ilustra la configuración de un aparato de transmisión en el que está montado el dispositivo de codificación de vídeo; y la parte (b) es un diagrama de bloques que ilustra la configuración de un aparato de recepción en el que está montado el dispositivo de decodificación de vídeo.

35 La Figura 46 ilustra un caso en el que un dispositivo de decodificación de vídeo, y un dispositivo de codificación de vídeo, se pueden utilizar para grabar y reproducir imágenes de vídeo; la parte (a) es un diagrama de bloques que ilustra la configuración de un aparato de registro en el que está montado el dispositivo de codificación de vídeo 2; y la parte (b) es un diagrama de bloques que ilustra la configuración de un aparato de reproducción en el que está montado el dispositivo de decodificación de vídeo.

40 DESCRIPCIÓN DE LAS FORMAS DE REALIZACIÓN

Primera forma de realización

(Datos codificados nº 1)

45 Antes de realizar una descripción detallada de un dispositivo de codificación de vídeo 2, y un dispositivo de decodificación de vídeo 1 de conformidad con una primera forma de realización, a continuación, se da a conocer la estructura de datos de datos codificados nº 1 que ha de generarse por el dispositivo de codificación de vídeo 2 y se decodificará por el dispositivo de decodificación de vídeo 1.

50 La Figura 2 ilustra la estructura de datos de los datos codificados nº 1. Los datos codificados nº 1 incluyen una secuencia y una pluralidad de imágenes que forman la secuencia a modo de ejemplo.

55 Las estructuras de niveles jerárquicos de una capa de imagen, o capas inferiores de los datos codificados nº 1, se muestran en la Figura 2. Las partes (a) a (d) de la Figura 2 ilustran, respectivamente, una capa de imagen que define una imagen PICT, una capa de corte que define un corte S, una capa de bloque de árbol, que define un bloque de árbol (Tree block) TBLK y, una capa de CU que define una unidad de codificación (Coding Unit; CU) incluida en el bloque de árbol TBLK.

60 (Capa de imagen)

65 En la capa de imagen, se define un conjunto de elementos de datos a los que se hará referencia mediante el dispositivo de decodificación de vídeo 1, con el fin de decodificar una imagen PICT que ha de procesarse (en lo sucesivo, también denominada como una "imagen objeto"). La imagen PICT incluye, tal como se muestra en la parte (a) de la Figura 2, una cabecera de imagen PH y cortes S₁ a S_{NS} (NS es el número total de cortes incluidos en la imagen PICT).

Si no es necesario distinguir los cortes individuales S_1 a S_{NS} entre sí, se pueden omitir los subíndices de los números de referencia. Otros elementos de datos provistos con subíndices, incluidos en los datos codificados nº 1, descritos a continuación, son gestionados de modo similar.

La cabecera de imagen PH incluye un conjunto de parámetros de codificación a los que hace referencia el dispositivo de decodificación de vídeo 1 con el fin de determinar un método de decodificación para una imagen objeto. A modo de ejemplo, información del modo de codificación (`entropy_coding_mode_flag`), que indica el modo de codificación de longitud variable, que se utiliza cuando el dispositivo de codificación de vídeo 2 ha realizado la codificación, es un ejemplo de parámetros de codificación incluidos en la cabecera de imagen PH.

Si el indicador `entropy_coding_mode_flag` es 0, la imagen PICT ha sido codificada por CAVLC (Codificación de Longitud Variable Adaptativa basada en Contexto). Si `entropy_coding_mode_flag` es 1, la imagen PICT ha sido codificada por CABAC (Codificación Aritmética Binaria Adaptativa basada en Contexto).

La cabecera de imagen PH se denomina, además, como un "conjunto de parámetros de imagen (PPS: Picture Parameter Set).

(Capa de Corte)

En la capa de corte, se define un conjunto de elementos de datos a los que se hace referencia por el dispositivo de decodificación de vídeo 1, con el fin de decodificar un corte S que ha de procesarse (en adelante denominado, además, como un "corte objeto"). El corte S incluye, según se muestra en la parte (b) de la Figura 2, una cabecera de corte SH y una secuencia de bloques de árbol $TBLK_1$ a $TBLK_{NC}$ (NC es el número total de bloques de árbol incluidos en el corte S).

La cabecera de corte SH incluye un conjunto de parámetros de codificación a los que debe hacer referencia el dispositivo de decodificación de vídeo 1 con el fin de determinar un método de decodificación para un corte objeto. La información de especificación de tipo de corte (`slice_type`), que especifica un tipo de corte, es un ejemplo de parámetros de codificación incluidos en la cabecera de corte SH.

Ejemplos de tipos de corte que se pueden especificar mediante la información de especificación de tipo de corte son (1) corte I, que utiliza solamente predicción intra cuando se realiza la codificación, (2) corte P, que utiliza predicción unidireccional, o predicción intra, cuando se realiza la codificación, y (3) corte B, que utiliza predicción unidireccional, predicción bidireccional, o predicción intra, cuando se realiza la codificación.

La cabecera de corte SH incluye, además, un parámetro de filtro FP al que se hace referencia mediante un filtro adaptativo, que se proporciona en el dispositivo de decodificación de vídeo 1. El parámetro de filtro FP puede estar incluido en la cabecera de imagen PH.

(Capa de bloque de árbol)

En la capa de bloque de árbol, se define un conjunto de elementos de datos a los que se hará referencia mediante el dispositivo de decodificación de vídeo 1, con el fin de decodificar un bloque de árbol TBLK que ha de procesarse (en adelante referido, además, como un "bloque de árbol objeto"). El bloque de árbol se puede denominar, además, como una unidad de codificación mayor (LCU: Largest Coding Unit).

El bloque de árbol TBLK incluye una cabecera de bloque de árbol TBLKH y elementos de información de unidad de codificación CU_1 a CU_{NL} (NL es el número total de elementos de información de unidad de codificación incluidos en el bloque de árbol TBLK). A continuación, se explicará, en primer lugar, la relación entre el bloque de árbol TBLK y la información de unidad de codificación CU.

El bloque de árbol TBLK se divide en particiones para especificar el tamaño de bloque que se utiliza para cada una de las operaciones de procesamiento, tal como predicción intra, predicción inter y transformación.

Las particiones del bloque de árbol TBLK se obtienen mediante división recursiva de árbol de cuadratura. En adelante, la estructura de árbol obtenida por esta división recursiva de árbol de cuadratura, se referirá como un "árbol de codificación".

En lo sucesivo, una partición correspondiente a una hoja del árbol, que es un nodo final de un árbol de codificación, se denominará como un nodo de codificación. El nodo de codificación es una unidad básica de procesamiento de codificación y, por lo tanto, será denominada, además, como una "unidad de codificación (CU)".

Es decir, cada uno de los elementos de información de unidad de codificación CU_1 a CU_{NL} (en lo sucesivo referida como "información de CU"), es información correspondiente a cada nodo de codificación (unidad de codificación) obtenida al realizar una división recursiva de árbol de cuadratura en el bloque de árbol TBLK.

La raíz del árbol de codificación está asociada con el bloque de árbol TBLK. Dicho de otro modo, el bloque de árbol TBLK está asociado con el nodo superior de una estructura de árbol de división de árbol de cuadratura, en donde se incluye, de forma recursiva, una pluralidad de nodos de codificación.

Los tamaños vertical y horizontal de cada nodo de codificación son la mitad de los tamaños vertical y horizontal de un nodo de codificación superior al que pertenece, directamente, el nodo de codificación (es decir, una partición de un nodo de un nivel superior al nodo de codificación).

Los posibles tamaños de cada nodo de codificación dependen de la información que especifica el tamaño y la profundidad jerárquica máxima del nodo de codificación incluido en un conjunto de parámetros de secuencia SPS de los datos codificados nº 1. A modo de ejemplo, si el tamaño de un bloque de árbol TBLK es 64×64 pixels, y la profundidad jerárquica máxima es tres, cada uno de los nodos de codificación del bloque de árbol TBLK, y niveles inferiores, pueden tomar uno de los tres tamaños, es decir, 64×64 pixels, 32×32 pixels y 16×16 pixels.

(Cabecera de bloque de árbol)

En la cabecera de bloque de árbol TBLKH, se incluyen los parámetros de codificación a los que se hace referencia por el dispositivo de decodificación de vídeo 1, con el fin de determinar un método de decodificación para un bloque de árbol objeto. Más concretamente, según se ilustra en la parte (c) de la Figura 2, la información de división de bloque de árbol SP_TBLK, que especifica un patrón de división para dividir un bloque de árbol objeto en CUs, y una diferencia de parámetro de cuantización Δqp (qp_delta), que especifica el tamaño de la etapa de cuantización están incluidos.

La información de división de bloque de árbol SP_TBLK es información que indica un árbol de codificación para dividir un bloque de árbol y, más específicamente, la configuración y el tamaño de cada una de las CUs incluidas en el bloque de árbol objeto, y la posición de cada CU dentro del bloque de árbol objeto.

La información de división de bloque de árbol SP_TBLK no tiene que incluir, de forma explícita, la configuración y el tamaño de cada CU. A modo de ejemplo, la información de división de bloque de árbol SP_TBLK puede ser un conjunto de indicadores (*split_coding_unit_flag*) que indican si el bloque de árbol objeto completo, o sub-zonas del bloque de árbol se someterá, o no, a la división de árbol de cuadratura. En este caso, la configuración y el tamaño del bloque de árbol se utilizan junto con el conjunto de indicadores, lo que hace posible especificar la configuración y el tamaño de cada CU.

La diferencia del parámetro de cuantización Δqp es una diferencia $qp - qp'$ entre el parámetro de cuantización qp , utilizado en el bloque de árbol objeto, y el parámetro de cuantización qp' , utilizado en un bloque de árbol que se ha codificado inmediatamente antes del bloque de árbol objeto.

(Capa de CU)

En la capa de CU, se define un conjunto de elementos de datos referidos por el dispositivo de decodificación de vídeo 1 para la decodificación de una CU que ha de procesarse (en lo sucesivo referida, además, como una "CU objeto").

Antes de una descripción específica de los elementos de datos incluidos en la CU de información de CU, la estructura de árbol de los elementos de datos incluidos en una CU se examina a continuación. El nodo de codificación es un nodo de una raíz de un árbol de predicción (PT), o un nodo de una raíz de un árbol de transformación (TT). El árbol de predicción y el árbol de transformación se examinan como sigue.

En el árbol de predicción, un nodo de codificación se divide en uno, o una pluralidad, de bloques de predicción y, se definen la posición y el tamaño de cada bloque de predicción. Dicho de otro modo, un bloque de predicción, o los bloques de predicción, son una o una pluralidad de zonas no superpuestas que forman un nodo de codificación. El árbol de predicción incluye uno, o una pluralidad, de bloques de predicción obtenidos mediante la división anteriormente descrita.

El procesamiento de predicción se realiza en cada bloque de predicción. El bloque de predicción, que es una unidad de predicción, se referirá, además, como una "unidad de predicción (PU)".

En términos generales, los métodos de división utilizados en un árbol de predicción se dividen en dos tipos, es decir, un tipo que utiliza una predicción intra y el otro tipo que utiliza una predicción inter.

En el tipo de predicción intra, el método de división incluye patrones de división de $2N \times 2N$ (el mismo tamaño que el de un nodo de codificación) y $N \times N$.

En el tipo de inter predicción, el método de división incluye patrones de división de $2N \times 2N$ (el mismo tamaño que el

del nodo de codificación), $2N \times N$, $N \times 2N$ y $N \times N$.

En el árbol de transformación, un nodo de codificación se divide en uno, o una pluralidad, de bloques de transformación y, se definen la posición y el tamaño de cada bloque de transformación. Dicho de otro modo, un bloque de transformación, o los bloques de transformación, son una o una pluralidad de zonas no superpuestas que forman un nodo de codificación. El árbol de transformación incluye uno, o una pluralidad, de bloques de transformación obtenidos mediante la división descrita con anterioridad.

Los métodos de división utilizados en el árbol de transformación se dividen en los siguientes tipos: asignación de una zona que tiene el mismo tamaño que el de un nodo de codificación, como el bloque de transformación; y la asignación de zonas obtenidas mediante la realización de la división recursiva de árbol de cuadratura, como en la división del bloque de árbol descrita anteriormente, como bloques de transformación.

El procesamiento de la transformación se realiza en cada bloque de transformación. El bloque de transformación, que es una unidad de transformación, se referirá, además, como una "unidad de transformación (TU)".

(Estructura de datos de información de CU)

Los datos incluidos en la CU información de CU se explicarán específicamente a continuación con referencia a la parte (d) de la Figura 2. Más concretamente, la información de CU incluye CU, tal como se ilustra en la parte (d) de la Figura 2, un indicador de salto operativo SKIP, PTI de información de PT, y TTI de información de TT.

El indicador de salto operativo SKIP es un indicador que indica si se aplica, o no, un modo de salto a una PU objeto. Si el valor del indicador de salto operativo SKIP es 1, es decir, si el modo de salto se aplica a una CU objeto, se omite PTI de información de PT y la TTI de información de TT de esta CU de información de CU. El indicador de salto operativo SKIP se omite en l cortes de imagen.

La PTI de información de PT es información relativa a un PT incluido en la CU. Dicho de otro modo, la PTI de información de PT es un conjunto de elementos de información relacionados con una, o una pluralidad, de PUs incluidas en el PT, y al que se hace referencia por el dispositivo de decodificación de vídeo 1 cuando se generan imágenes de predicción. La PTI de información de PT incluye, tal como se ilustra en la parte (d) de la Figura 2, información de tipo de predicción PType e información de predicción PInfo.

La información de tipo de predicción PType es información que especifica si se utilizará la predicción intra o la predicción inter como el método de generación de imágenes de predicción para la PU objeto.

La información de predicción PInfo está constituida por información de predicción intra, o información de predicción inter, dependiendo de si la información de tipo de predicción PType especifica predicción intra o predicción inter. A continuación, una PU a la que se aplica la predicción intra se puede denominar, además, como una "PU intra" y, una PU a la que se aplica la predicción inter se puede denominar, además, como una "PU inter".

La información de predicción PInfo incluye, además, información que especifica la configuración, tamaño y posición de la PU objeto. Tal como se indicó con anterioridad, la generación de una imagen de predicción se realiza utilizando una PU como una unidad. Los detalles de la información de predicción PInfo se examinarán más adelante.

La TTI de información de TT es información relativa a un TT incluido en la CU. Dicho de otro modo, la TTI de información TT es un conjunto de elementos de información relativos a una, o una pluralidad, de TUs incluidas en el TT y al que se hace referencia por el dispositivo de decodificación de vídeo 1 cuando se decodifican datos residuales. A continuación, una TU se puede denominar, además, como un "bloque".

La TTI de información de TT incluye, según se muestra en la parte (d) de la Figura 2, la información de división TT SP_TT que especifica un patrón de división utilizado para dividir la CU objeto en bloques de transformación y los residuos de predicción cuantizados QD_1 a QD_{NT} (NT es el número total de bloques incluidos en la CU objeto).

La información de división TT SP_TT es información para la determinación de la configuración y el tamaño de cada TU incluida en la CU objeto, y la posición de cada TU dentro de la CU objeto. A modo de ejemplo, la información de división TT SP_TT se puede poner en práctica mediante información (split_transform_unit_flag) que indica si un nodo objeto se dividirá, o no, e información (trafoDepth) que indica la profundidad de la división.

Si el tamaño de una CU es 64×64 , el tamaño de cada una de las Tus, obtenidas mediante la división de la CU, puede estar en un margen de 32×32 pixels a 2×2 pixels.

Cada residuo de predicción cuantizado QD son datos codificados generados como un resultado del dispositivo de codificación de vídeo 2, que ejecuta el Procesamiento 1 al Procesamiento 3 siguiente en un bloque objeto, que es un bloque que ha de procesarse.

Procesamiento 1: la realización de la transformación DCT (Transformada de Coseno Discreta) en un residuo de predicción, obtenido mediante la sustracción de una imagen de predicción desde una imagen que ha de codificarse;

Procesamiento 2: la cuantización de un coeficiente de transformación obtenido en el Procesamiento 1; y

Procesamiento 3: la realización de una codificación de longitud variable en el coeficiente de transformación cuantizado en el Procesamiento 2.

El parámetro de cuantización qp , anteriormente descrito, representa un tamaño de etapa de cuantización QP ($QP=2^{qp/6}$) utilizado cuando el dispositivo de codificación de vídeo 2 realiza una cuantización del coeficiente de transformación.

(Información de predicción PInfo)

Tal como se ha indicado anteriormente, la información de predicción PInfo es de dos tipos, es decir, información inter predicción e información intra predicción.

La información inter predicción incluye parámetros de codificación a los que hace referencia el dispositivo de decodificación de vídeo 1 cuando se genera una imagen inter predicción utilizando la inter predicción. Más concretamente, la información de predicción inter incluye información de división inter PU que especifica un patrón de división utilizado para dividir una CU objeto en inter PUs y parámetros de predicción inter correspondientes a inter PUs individuales.

Los parámetros inter-predicción incluyen un índice de imagen de referencia, un índice de vectores de movimiento de estimación y un residuo de vector de movimiento.

La información de intra-predicción incluye parámetros de codificación referidos por el dispositivo de decodificación de vídeo 1 cuando se genera una imagen de intra-predicción utilizando una intra predicción. Más concretamente, la información de intra-predicción incluye información de división intra PU que especifica un patrón de división utilizado para dividir una CU objeto en intra PUs, y parámetros intra-predicción referentes a intra PUs individuales. El parámetro intra-predicción es un parámetro para especificar un método de intra-predicción (modo de predicción) para cada intra PU.

(Unidad de desplazamiento)

En esta forma de realización, cada imagen o cada corte está dividido, de forma recursiva, en una pluralidad de unidades de desplazamiento (también referida como una "QAOU: Unidad de Desplazamiento Adaptativa de Cuadratura") de conformidad con una estructura de árbol de cuadratura. La QAOU es una unidad de procesamiento del procesamiento de filtrado de desplazamiento, realizado por un filtro de desplazamiento adaptativo de esta forma de realización.

Tal como se ilustra en las partes (e) y (f) de la Figura 2, la información de QAOU, que es información sobre cada QAOU, incluye `sao_split_flag` que indica si se dividirá, o no, la QAOU. Más concretamente, `sao_split_flag` se especifica mediante argumentos (`sao_curr_depth`, `ys`, `xs`), que se analizarán más adelante y, se representa, además, por `sao_split_flag[sao_curr_depth][ys][xs]`.

Si `sao_split_flag`, incluido en una QAOU, indica que la QAOU se dividirá aún más (es decir, si la QAOU no es una hoja del árbol), la información de QAOU relativa a la QAOU incluye, según se muestra en la parte (e) de la Figura 2, elementos de información de QAOU relativos a una pluralidad de QAOU incluidas en la QAOU.

Por otro lado, si `sao_split_flag`, incluido en una QAOU, indica que la QAOU no se dividirá más (es decir, si la QAOU es una hoja del árbol), la información de QAOU relativa a la QAOU incluye, tal como se ilustra en la parte (f) de la Figura 2, información de desplazamiento OI relativa a la QAOU. La información de desplazamiento OI incluye, según se muestra en la parte (f) de la Figura 2, información de especificación de tipo de desplazamiento OTI que especifica un tipo de desplazamiento y un grupo de desplazamiento que se determina de conformidad con el tipo de desplazamiento. Tal como se ilustra en la parte (f) de la Figura 2, el grupo de desplazamiento incluye una pluralidad de desplazamientos.

Un desplazamiento incluido en datos codificados es un valor cuantizado. Un desplazamiento incluido en datos codificados puede ser un residual de predicción obtenido al utilizar una cierta predicción, a modo de ejemplo, una predicción lineal. Si la profundidad de bits de píxel, que se explicará más adelante, se establece de forma variable para cada zona unitaria, la profundidad de bits de píxel para una zona que ha de procesarse puede incluirse en la información de especificación de tipo de desplazamiento OTI, puesto que la profundidad de bits de desplazamiento, el margen de valores de desplazamiento, y los valores de desplazamiento se cambian de conformidad con la profundidad de bits del píxel en esta forma de realización.

La información de desplazamiento OI se discutirá a continuación cambiando los dibujos a los que se hace referencia.

La Figura 3 ilustra cada sintaxis incluida en la información de desplazamiento OI (indicada por `sao_offset_param` () en la Figura 3).

Tal como se ilustra en la Figura 3, la información de desplazamiento OI incluye la sintaxis `sao_type_idx [sao_curr_depth] [ys] [xs]`. Si `sao_type_idx [sao_curr_depth] [ys] [xs]` no es 0, `sao_offset [sao_curr_depth] [ys] [xs] [i]` se incluye en la información de desplazamiento OI.

(`sao_curr_depth`, `ys`, `xs`)

Un argumento `sao_curr_depth`, que es un argumento de `sao_type_idx` y `sao_offset`, es un índice que indica la profundidad de división de una QAOU y, `ys` y `xs` son índices que indican, respectivamente, la posición en la dirección y, y la posición en la dirección x de una QAOU (o una QAOMU, que se examinará más adelante).

Las partes (a) a (e) de la Figura 4 muestran modos de división de una QAOU de conformidad con los valores de `sao_curr_depth`: la parte (a) muestra un modo de división cuando `sao_curr_depth=0`, la parte (b) muestra un modo de división cuando `sao_curr_depth=1`, la parte (c) muestra un modo de división cuando `sao_curr_depth=2`, la parte (d) muestra un modo de división cuando `sao_curr_depth=3` y, la parte (e) muestra un modo de división cuando `sao_curr_depth=4`. Tal como se ilustra en las partes (a) a (e) de la Figura 4, cada QAOU se especifica mediante `sao_curr_depth` y (`xs`, `ys`).

Tal como se ilustra en la parte (a) de la Figura 4, cuando `sao_curr_depth=0`, `xs` e `ys` son ambos 0. Según se muestra en la parte (b) de la Figura 4, cuando `sao_curr_depth=1`, `xs` e `ys` toman los valores 0 o 1. Tal como se ilustra en la parte (c) de la Figura 4, cuando `sao_curr_depth=2`, `xs` e `ys` toman uno de entre 0, 1, 2 y 3. Generalmente, en respuesta a un valor dado de `sao_curr_depth`, `xs` e `ys` toman valores de 0 a $2^{\text{sao_curr_depth}-1}$.

(`sao_type_idx`)

La sintaxis `sao_type_idx [sao_curr_depth] [ys] [xs]` corresponde a la información de especificación de tipo de desplazamiento OTI, anteriormente descrita, y es una sintaxis para especificar el tipo de desplazamiento de cada QAOU. En lo sucesivo, `sao_type_idx` puede referirse, simplemente, como un "tipo de desplazamiento".

En esta forma de realización, `sao_type_idx [sao_curr_depth] [ys] [xs]` toma números enteros de 0 a 6. Cuando `sao_type_idx [sao_curr_depth] [ys] [xs]=0`, indica que el procesamiento de filtrado de desplazamiento no se realiza en una imagen que aún no se ha sometido a filtrado de desplazamiento (a modo de ejemplo, una imagen decodificada desbloqueada `P_DB`, que se analizará más adelante) en una QAOU objeto. Cuando `sao_type_idx [sao_curr_depth] [ys] [xs]=1 a 4`, indica que el procesamiento de desplazamiento de borde se realiza en una imagen que aún no se ha sometido a filtrado de desplazamiento en una QAOU objeto. Cuando `sao_type_idx [sao_curr_depth] [ys] [xs]=5 o 6`, indica que el procesamiento de desplazamiento de banda se realiza en una imagen que aún no ha sido sometida a filtrado de desplazamiento en una QAOU objeto. El procesamiento de desplazamiento de borde, y el procesamiento de desplazamiento de banda, se tratarán, de forma específica, a continuación.

(`sao_offset`)

La sintaxis `sao_offset [sao_curr_depth] [ys] [xs] [i]` es una sintaxis que representa un valor específico de un desplazamiento que ha de añadirse a cada píxel incluido en una QAOU objeto en el proceso de filtrado de desplazamiento utilizando un filtro de desplazamiento adaptativo de esta forma de realización. En esta forma de realización, `sao_offset` también puede ser referido simplemente como un "desplazamiento".

La sintaxis `sao_offset [sao_curr_depth] [ys] [xs] [i]` se especifica por el índice `i`, así como los argumentos `sao_curr_depth`, `ys` y `xs`. El índice `i` es un índice para especificar una clase y, se indica, además, por `class_idx`. El índice `i` toma uno de los números enteros 0 a 4 cuando el valor de `sao_type_idx [sao_curr_depth] [ys] [xs]` es uno de entre 1 a 4 (es decir, en el caso de un desplazamiento de borde). El índice `i` toma uno de los números enteros 0 a 16 cuando el valor de `sao_type_idx [sao_curr_depth] [ys] [xs]` es 5 o 6 (es decir, en el caso de un desplazamiento de banda). En cualquier caso, cuando `i` es 0, ello indica que no se añade un desplazamiento, lo que se explicará más adelante.

El filtro de desplazamiento adaptativo de esta forma de realización clasifica un píxel objeto incluido en una QAOU objeto, como una de la pluralidad de clases descritas anteriormente y, añade un desplazamiento Offset asociado con la clase del píxel objeto, al píxel objeto. Lo anterior se examinará más adelante.

El descriptor `ue(v)`, ilustrado en la Figura 3, indica que la sintaxis asociada con este descriptor es un valor numérico sin signo y que la codificación de longitud variable se realizará en el valor numérico. El descriptor `se(v)`, mostrado en la Figura 3, indica que la sintaxis asociada con este descriptor es un valor numérico con signo y que la codificación

de longitud variable se realizará en un signo y un valor absoluto que forman el valor numérico.

(Dispositivo de decodificación de vídeo 1)

El dispositivo de decodificación de vídeo 1 de esta forma de realización se describirá, a continuación, haciendo referencia a las Figuras 1 y 5 a 13. El dispositivo de decodificación de vídeo 1 incluye, de forma parcial, un método definido en H. 264/MPEG-4. AVC, un método utilizado en el software KTA, que es un códec de desarrollo conjunto en VCEG (Grupo de Expertos de Codificación de Vídeo), un método utilizado en el software TMuC (Modelo de Prueba bajo Consideración), que es un códec sucesor del códec utilizado en el software KTA y, una tecnología utilizada en el software HM (HEVC TestModel).

La Figura 5 es un diagrama de bloques que ilustra la configuración del dispositivo de decodificación de vídeo 1. El dispositivo de decodificación de vídeo 1 incluye, tal como se ilustra en la Figura 5, un decodificador de código de longitud variable 13, una unidad de reconstrucción de vector de movimiento 14, una memoria intermedia 15, un generador de imagen inter-predicción 16, un generador de imagen intra-predicción 17, una unidad de determinación de método de predicción 18, una unidad de transformación y cuantización inversa 19, un sumador 20, un filtro de desbloqueo 41, un filtro adaptativo 50, y un filtro de desplazamiento adaptativo 60. El dispositivo de decodificación de vídeo 1 es un dispositivo para generar una imagen de vídeo nº 4 mediante la decodificación de datos codificados nº 1.

El decodificador de código de longitud variable 13 decodifica un parámetro de predicción PP relativo a cada partición desde de los datos codificados nº 1. Es decir, con respecto a una partición inter-predicción, el decodificador de código de longitud variable 13 decodifica un índice de imagen de referencia RI, un índice de vector de movimiento de estimación PMVI, y un MVD residual de vector de movimiento desde de los datos codificados nº 1 y, los proporciona a la unidad de reconstrucción de vector de movimiento 14. Por otro lado, con respecto a una partición intra-predicción, el decodificador de código de longitud variable 13 decodifica (1) información de especificación de tamaño que especifica el tamaño de la partición, y (2) información de especificación de índice de predicción que especifica un índice de predicción desde de los datos codificados nº 1 y, los proporciona al generador de imagen intra-predicción 17. El decodificador de código de longitud variable 13 decodifica, además, información de CU procedente de datos codificados, y la proporciona a la unidad de determinación del método de predicción 18 (no se ilustra el flujo de información de CU). El decodificador de código de longitud variable 13 decodifica, además, un QD residual de predicción cuantizado con respecto a cada bloque, y una diferencia de parámetro de cuantización Δqp , con respecto a un bloque de árbol que incluye este bloque, desde de los datos codificados nº 1, y los proporciona a la unidad de transformación inversa y cuantización inversa 19. El decodificador de código de longitud variable 13 extrae, además, información de QAOU desde de los datos codificados nº 1, y suministra la información QAOU extraída al filtro de desplazamiento adaptativo 60.

La unidad de reconstrucción de vector de movimiento 14 reconstruye un vector de movimiento mv con respecto a cada partición inter-predicción procedente de MVD residual de vector de movimiento concerniente a esta partición, y desde un vector de movimiento reconstruido mv', que concierne a otra partición. Más concretamente, la unidad de reconstrucción de vector de movimiento 14 (1) determina un vector de movimiento de estimación pmv desde del vector de movimiento reconstruido mv', de conformidad con un método de estimación especificado por el índice de vector de movimiento de estimación PMVI y, (2) obtiene el vector de movimiento mv añadiendo el MVD residual del vector de movimiento al vector de movimiento de estimación pmv. El vector de movimiento reconstruido mv' relacionado con otra partición, puede ser objeto de lectura desde de la memoria intermedia 15. La unidad de reconstrucción de vector de movimiento 14 proporciona el vector de movimiento reconstruido mv al generador de imagen inter-predicción 16, junto con el índice de imagen de referencia asociado RI. En el caso de una partición inter-predicción en la que se realizará una predicción bidireccional (predicción ponderada), la unidad de reconstrucción de vector de movimiento 14 proporciona dos vectores de movimiento reconstruidos mv1 y mv2 al generador de imagen inter-predicción 16, junto con índices de imagen de referencia asociados RI1 y RI2.

El generador de imagen inter-predicción 16 genera una imagen de movimiento compensado mc en relación con cada partición inter-predicción. Más concretamente, el generador de imagen inter-predicción 16 genera una imagen de movimiento compensado mc, desde de una imagen decodificada filtrada P_FL', que se especifica por el índice de imagen de referencia RI proporcionado desde la unidad de reconstrucción de vector de movimiento 14 utilizando el vector de movimiento mv, que se proporciona desde de la unidad de reconstrucción de vector de movimiento 14. La imagen decodificada filtrada P_FL' es una imagen obtenida al realizar un procesamiento de desbloqueo mediante el uso del filtro de desbloqueo 41, el procesamiento de filtrado de desplazamiento mediante el uso del filtro de desplazamiento adaptativo 60, y el procesamiento de filtrado adaptativo utilizando el filtro adaptativo 50 en una imagen decodificada P. El generador de imagen inter-predicción 16 puede leer valores de píxel de píxeles individuales que forman la imagen decodificada filtrada P_FL', desde de la memoria intermedia 15. La imagen de movimiento compensado mc, generada por el generador de imagen inter-predicción 16, se proporciona a la unidad de determinación del método de predicción 18 como una imagen inter-predicción Pred_Inter. En el caso de una partición inter-predicción en la que se realizará una predicción bidireccional (predicción ponderada), el generador de imagen inter-predicción 16 realiza: (1) la generación de una imagen de movimiento compensado mc1 desde de una imagen decodificada filtrada P_FL1', especificada por el índice de imagen de referencia RI1 utilizando un vector de

movimiento mv1; (2) la generación de una imagen de movimiento compensado mc2 desde de una imagen decodificada filtrada P_FL2', que se especifica por el índice de imagen de referencia R12 utilizando un vector de movimiento mv2; y (3) la generación de una imagen inter-predicción Pred_Inter mediante la adición de un valor de desplazamiento a una media ponderada de la imagen de movimiento compensado mc1, y la imagen de movimiento compensado mc2.

El generador de imagen intra-predicción 17 genera una imagen de predicción Pred_Intra con respecto a cada partición intra-predicción. Más concretamente, el generador de imagen intra-predicción 17 se refiere a un modo de predicción decodificado desde de los datos codificados nº 1, y asigna el modo de predicción a una partición objeto, a modo de ejemplo, en un orden de barrido más rápido. A continuación, el generador de imagen intra-predicción 17 genera una imagen de predicción Pred_Intra desde de la imagen decodificada P de conformidad con el método de predicción representado por el modo de predicción. La imagen intra-predicción Pred_Intra, generada por el generador de imagen intra-predicción 17 se proporciona a la unidad de determinación del método de predicción 18.

El generador de imagen intra-predicción 17 proporciona, además, el tamaño de la partición objeto y la información del modo de codificación intra IEM, que es información que indica el modo de predicción asignado a la partición objeto, al filtro adaptativo 50.

La unidad de determinación del método de predicción 18 determina, sobre la base de la información de CU, si cada partición es una partición inter-predicción que se someterá a una predicción inter, o una partición intra-predicción que se someterá a una predicción intra. Si la partición objeto es una partición inter-predicción, la unidad de determinación del método de predicción 18 proporciona la imagen inter-predicción Pred_Inter, generada por el generador de imagen inter-predicción 16 para el sumador 20 como la imagen de predicción Pred. Si la partición objeto es una partición intra-predicción, la unidad de determinación del método de predicción 18 proporciona la imagen intra-predicción Pred_Intra, generada por el generador de imagen intra-predicción 17 al sumador 20 como la imagen de predicción Pred.

La unidad de transformación inversa de cuantización inversa 19 realiza: (1) la cuantización inversa de la predicción cuantizada residual QD; (2) transformación DCT inversa (Transformada de Coseno Discreta), en un coeficiente DCT obtenido al realizar la cuantización inversa; y (3) el suministro de una predicción residual D obtenida al realizar la transformación DCT inversa al sumador 20. Cuando se realiza la cuantización inversa en la predicción cuantizada residual QD, la unidad de transformación inversa y cuantización inversa 19 determina una etapa de cuantización QP desde de la diferencia del parámetro de cuantización Δq_p , que se proporciona desde el decodificador de código de longitud variable 13. El parámetro de cuantización q_p se puede determinar mediante la adición de la diferencia del parámetro de cuantización Δq_p a un parámetro de cuantización q_p' , relativo a un bloque de árbol que ha sido sometido a una transformación DCT inversa y cuantización inversa inmediatamente antes del bloque de árbol objeto, y la etapa de cuantización QP se puede determinar calculando $QP=2^{pq/6}$ desde de la etapa de cuantización q_p . La generación de una predicción residual D, por la unidad de transformación inversa y cuantización inversa 19 se realiza utilizando un bloque (unidad de transformación) como una unidad.

El sumador 20 genera una imagen decodificada P añadiendo la predicción residual D, que se proporciona por la unidad de transformación inversa y cuantización inversa 19 a la imagen de predicción Pred, suministrada por la unidad de determinación del método de predicción 18.

Cuando la diferencia entre los valores de píxel, de los píxeles adyacentes entre sí, con un límite de bloque o un límite de CU entre ellos en la imagen decodificada P, es menor que un umbral predeterminado, el filtro de desbloqueo 41 realiza el procesamiento de desbloqueo en el límite de bloque, o el límite de CU en la imagen decodificada P, suavizando así una imagen en la proximidad del límite de bloque, o el límite de CU. La imagen sometida al procesamiento de desbloqueo por el filtro de desbloqueo 41 se envía al filtro de desplazamiento adaptativo 60 como una imagen decodificada desbloqueada P_DB.

El filtro de desplazamiento adaptativo 60 realiza el procesamiento de filtrado de desplazamiento utilizando un desplazamiento decodificado desde de los datos codificados nº 1, en la imagen decodificada desbloqueada P_DB, que se proporciona desde el filtro de desbloqueo 41, utilizando una QAOU como una unidad de procesamiento, generando así una imagen decodificada de desplazamiento filtrado P_OF. La imagen decodificada de desplazamiento filtrado P_OF generada, se proporciona al filtro de desplazamiento adaptativo 60. La configuración específica del filtro adaptativo 60 se describirá a continuación y, por lo tanto, se omite aquí una explicación de la misma.

El filtro adaptativo 50 realiza el procesamiento de filtrado utilizando un parámetro de filtro FP decodificado desde los datos codificados nº 1, en la imagen decodificada de desplazamiento filtrado P_OF, proporcionada desde el filtro de desplazamiento adaptativo 60, con lo que se genera una imagen decodificada filtrada P_FL. La imagen sometida al proceso de filtrado por el filtro adaptativo 50 se envía al exterior como la imagen decodificada filtrada P_FL y, además, se memoriza en la memoria intermedia 15 en asociación con información de especificación de POC decodificada desde de los datos codificados por el decodificador de código de longitud variable 13.

(Filtro de desplazamiento adaptativo 60)

La Figura 1 es un diagrama de bloques que ilustra la configuración del filtro de desplazamiento adaptativo 60. El filtro de desplazamiento adaptativo 60 incluye, según se ilustra en la Figura 1, un decodificador de información de filtro de desplazamiento adaptativo 61, y un procesador de filtro de desplazamiento adaptativo 62.

El decodificador de información de filtro de desplazamiento adaptativo 61 incluye, tal como se ilustra en la Figura 1, una sección de decodificación de información de desplazamiento 611, una sección de decodificación de estructura de QAOU 612 y una sección de establecimiento de atributos de desplazamiento 613.

La sección de decodificación de información de desplazamiento 611 se refiere a información de QAOU incluida en los datos codificados nº 1, y la información de desplazamiento decodificada OI, que se incluye en la información de QAOU. Los valores `sao_type_idx` [`sao_curr_depth`] [`ys`] [`xs`] y `sao_offset` [`sao_curr_depth`] [`ys`] [`xs`] [`i`], obtenidos mediante decodificación de la información de desplazamiento OI, se proporcionan a una sección de memorización de información de desplazamiento 621 en asociación con los argumentos (`sao_curr_depth`, `ys`, `xs`) y los argumentos (`sao_curr_depth`, `ys`, `xs`, `i`), respectivamente.

La sección de decodificación de estructura de QAOU 612 determina una estructura de división de QAOU mediante la decodificación de `sao_split_flag` [`sao_curr_depth`] [`ys`] [`xs`], incluida en la información de QAOU y proporciona la información de estructura de QAOU que indica la estructura de división de QAOU determinada, a la sección de memorización de información de desplazamiento 621.

La sección de establecimiento de atributos de desplazamiento 613 determina la profundidad del bit de desplazamiento (también referida como "SAO_DEPTH"), y el margen de valores de desplazamiento. En este caso, la profundidad del bit de desplazamiento se determina desde de la profundidad de bits de píxel (también referida como "PIC_DEPTH") (no ilustrada) introducida en la sección de establecimiento de atributos de desplazamiento 613. La profundidad de bits de píxel representa un margen de valores de píxel que forman una entrada de imagen en el filtro de desplazamiento adaptativo 60 como la anchura de bits. Cuando la profundidad de bits de píxel es N bits, los valores de píxel tienen un margen de 0 a 2^N-1 . A modo de ejemplo, cuando la profundidad de bits de píxel es 8 bits, los valores de píxel pueden tener un margen de 0 a 255. Si la entrada de imagen en el filtro de desplazamiento adaptativo 60 es una imagen decodificada, obtenida mediante el dispositivo de decodificación de vídeo 1, o una imagen decodificada, de forma local, obtenida por el dispositivo de codificación de vídeo 2, se utiliza la profundidad de bits de la imagen decodificada o la imagen decodificada localmente. La precisión de una entrada de imagen en el filtro de desplazamiento adaptativo 60, y la precisión de una salida de imagen procedente del filtro de desplazamiento adaptativo 60, se representa por la profundidad de bits de píxel.

Al igual que la profundidad de bits del píxel, se puede decodificar y utilizar un valor determinado mediante el uso de la sintaxis en los datos codificados nº 1. A modo de ejemplo, en el caso de H. 264/AVC, la profundidad de bits de píxel se puede determinar utilizando `bit_depth_luma_minus8` en un conjunto de parámetros de secuencia. En otra forma operativa, la profundidad de bits de píxel se puede determinar para cada zona unitaria que ha de procesarse, en cuyo caso, la profundidad de bits de píxel se decodifica desde de la cabecera de cada zona unitaria. La profundidad de bits de píxel, en cada zona unitaria, se puede incluir en la información de parámetro, o la cabecera en los datos codificados nº 1, que se utiliza para decodificar una entrada de imagen en el filtro de desplazamiento adaptativo 60, o se puede incluir como parte de la información de QAOU. Si la profundidad de bits del píxel está incluida en la información de parámetro o en la cabecera, que se utiliza para decodificar una imagen de entrada, puede incluirse en una cabecera de parámetro de imagen, una cabecera de corte, una LCU o una CU. Si la profundidad de bits de píxel se incluye en parte de la información de QAOU, puede incluirse en una hoja del árbol de QAOU, o en información de QAOU de un nivel predeterminado (a modo de ejemplo, el mayor nivel de QAOU o el primer nivel QAOU). Con respecto a la codificación de la profundidad de bits de píxel, es preferible que se codifique como un valor de diferencia desde de ocho bits. Si se utiliza parte de la información de QAOU, en lugar de utilizar la profundidad de bits de píxel, es adecuado, además, que se codifique el propio valor de desplazamiento. En este caso, el valor de desplazamiento puede codificarse, de forma adecuada, solamente cuando excede los ocho bits. El valor de desplazamiento es una cantidad de desplazamiento binario que es necesaria para realizar la cuantización inversa. Como un parámetro utilizado para la cuantización inversa, se puede utilizar un valor de etapa en lugar del valor de desplazamiento. En este caso, la cuantización inversa de un desplazamiento se realiza multiplicando el desplazamiento por un valor de etapa y, la cuantización de un desplazamiento se realiza dividiendo el desplazamiento por un valor de etapa.

La profundidad de bits de desplazamiento es un valor que indica la precisión de un desplazamiento codificado en los datos codificados nº 1. El desplazamiento incluido en los datos codificados nº 1 es un valor cuantizado, y este desplazamiento cuantizado se decodifica y, a continuación, se somete a una cuantización inversa y se transforma en la profundidad de bits correspondiente a la profundidad de bits del píxel por una unidad de determinación de desplazamiento 625, que se discutirá más adelante. A continuación, el desplazamiento cuantizado inverso se añade mediante un sumador de desplazamiento 626, que se explicará más adelante. La profundidad de bits de desplazamiento tiene un valor igual, o menor, que la profundidad de bits de píxel. Si la profundidad de bits de desplazamiento es menor que la profundidad de bits de píxel por k bits, esto significa que el valor de un

desplazamiento codificado es un valor cuantizado por una etapa de cuantización de desplazamiento de 2^k . La transformación de la profundidad de bits decodificada, en la profundidad de bits de píxel, se realiza mediante la unidad de determinación de desplazamiento 625, que se describirá a continuación.

La sección de establecimiento de atributos de desplazamiento 613 establece un margen de valores de desplazamiento utilizando la profundidad de bits de desplazamiento determinada. La sección de establecimiento de atributos de desplazamiento 613 establece, además, un valor de desplazamiento desde la profundidad de bits del píxel de desplazamiento y el margen de valores de desplazamiento. El margen de valores de desplazamiento establecido, y el valor de desplazamiento establecido, se proporcionan a la sección de memorización de información de desplazamiento 621 y a la sección de decodificación de información de desplazamiento 611. La determinación de la profundidad de bits de desplazamiento, y la configuración del valor de desplazamiento se realizan utilizando uno de los patrones S1 a S6, que se tratará más adelante. La configuración del margen de valores de desplazamiento se realiza utilizando uno de los patrones C1 y C2, que se explicará a continuación. El margen de valores de desplazamiento y el valor de desplazamiento, que se utilizan como atributos de un desplazamiento, se denominan como "atributos de desplazamiento". El procesador de filtro de desplazamiento adaptativo 62 incluye, según se ilustra en la Figura 1, la sección de memorización de información de desplazamiento 621, un controlador de QAOU 622, una sección de determinación de tipo de desplazamiento 623, una sección de clasificación 624, la sección de determinación de desplazamiento 625 y un sumador de desplazamiento 626.

La sección de memorización de información de desplazamiento 621 gestiona y memoriza el tipo de desplazamiento especificado para cada QAOU, y los valores específicos de desplazamientos con respecto a clases individuales que se pueden seleccionar para el tipo de desplazamiento especificado, sobre la base de la información de estructura de QAOU, `sao_type_idx [sao_curr_depth] [ys] [xs]` y, `sao_offset [sao_curr_depth] [ys] [xs] [i]`. La sección de memorización de información de desplazamiento 621 tiene una memoria de mapa y una memoria de lista.

En la memoria de mapa, se memoriza un índice QAOU asignado a cada unidad mínima de desplazamiento (también referida como "QAOMU: Unidad Mínima de Desplazamiento Adaptativo de Cuadratura"), que se determina por la profundidad de división. El índice QAOU será examinado más adelante. La memoria de mapa se describe, a continuación, con referencia a la parte (a) de la Figura 6. La parte (a) de la Figura 6 ilustra ejemplos de índices QAOU memorizados en la memoria de mapa y, muestra QAOMUs con una profundidad de división de tres y que forman la unidad de procesamiento (a modo de ejemplo, una LCU) e índices QAOU asignados a QAOMUs. En la parte (a) de la Figura 6, los índices 0 a 9 especifican QAOUs de una manera sencilla sin tener en cuenta la profundidad de división de QAOU. En el ejemplo ilustrado en la parte (a) de la Figura 6, una QAOU especificada por el índice QAOU=*i* se indica por QAOU*i*. En la parte (a) de la Figura 6, las líneas finas indican límites QAOMU, mientras que las líneas gruesas indican límites QAOU.

Tal como se muestra en la parte (a) de la Figura 6, QAOU0 está constituido por cuatro QAOMUs y 0 se asigna a estos cuatro QAOMU como el índice QAOU. QAOU3 está constituido por un QAOMU, y 3 se asigna a este QAOMU como el índice QAOU. De este modo, en la memoria de mapa, se memorizan índices QAOU asignados a las QAOMUs individuales.

En la memoria de lista, cada índice QAOU, el tipo de desplazamiento asociado con cada índice QAOU, y valores específicos de desplazamientos con respecto a clases, que se pueden seleccionar para este tipo de desplazamiento, se memorizan asociados sí. En la sección de memorización de información de desplazamiento 621, se memorizan desplazamientos que tienen un margen de valores restringido, establecido por la sección de establecimiento de atributos de desplazamiento 613.

La parte (b) de la Figura 6 ilustra un ejemplo de información memorizada en la memoria de lista, y muestra un tipo de desplazamiento asociado con cada uno de los índices QAOU 0 a 9, y desplazamientos con respecto a clases individuales que se pueden seleccionar para cada tipo de desplazamiento. En la parte (b) de la Figura 6, "xxx" indica valores numéricos específicos que pueden ser diferentes entre sí.

En la parte (b) de la Figura 6, "BO_1" indica un tipo de desplazamiento especificado por `sao_type_idx=5`. "EO_1" indica un tipo de desplazamiento especificado por `sao_type_idx=1`. De este modo, desplazamientos de borde, que son de un tipo de desplazamiento especificado por `sao_type_idx=1, 2, 3, 4`, se indican, además, por EO_1, 2, 3, 4, respectivamente, y desplazamientos de banda, que son de un tipo de desplazamiento especificado por `sao_type_idx=5, 6`, se indican, además, por BO_1, 2, respectivamente.

Tal como se ilustra en la parte (b) de la Figura 6, cuando el tipo de desplazamiento es un desplazamiento de banda, un total de dieciséis desplazamientos, es decir, desde el desplazamiento 1 hasta el desplazamiento 16, se memorizan en la memoria de lista para este tipo de desplazamiento. El desplazamiento 1 hasta el desplazamiento 16 son valores especificados por `sao_offset [sao_curr_depth] [ys] [xs] [1]` hasta `sao_offset [sao_curr_depth] [ys] [xs] [16]`, respectivamente, cuando el valor de `sao_type_idx [sao_curr_depth] [ys] [xs]` es 5 o 6. Por otro lado, cuando el tipo de desplazamiento es un desplazamiento de borde, un total de cuatro desplazamientos, es decir, desde el desplazamiento 1 hasta el desplazamiento 4, se memorizan en la memoria de lista para este tipo de desplazamiento. El desplazamiento 1 hasta el desplazamiento 4 son valores especificados por `sao_offset [sao_curr_depth] [ys] [xs]`

[1] hasta `sao_offset[sao_curr_depth][ys][xs]` [4], respectivamente, cuando el valor de `sao_type_idx[sao_curr_depth][ys][xs]` es uno de entre 1, 2, 3 y 4. No se memoriza ningún valor en el desplazamiento 5 al desplazamiento 16.

5 El tamaño de la memoria de cada desplazamiento memorizado en la memoria de lista se determina por el margen de valores de desplazamiento, que se proporciona por la sección de establecimiento de atributos de desplazamiento 613. Cuando el margen de valores de desplazamiento es de -2^4 a 2^4-1 , cada desplazamiento se puede representar por cinco bits y, por lo tanto, se requiere un tamaño de memoria que tiene cinco bits.

10 Se añade un número QAOMU a cada QAOMU y, QAOMUs se pueden distinguir entre sí por los números QAOMU. En adelante, QAOMU que tenga un número QAOMU N_Q se referirá, además, como "QAOMUN $_Q$ ".

Los índices QAOU son números de bloque QAOU que se especifican, en serie, cuando la profundidad de división es desde 0 al máximo. Si la profundidad de división máxima es cuatro, los valores de 0 a 340 se especifican para todos los bloques ($1 + 4 + 16 + 64 + 256 = 341$) de la totalidad de profundidades de división por los índices QAOU, tal como se muestra en la Figura 7.

Puesto que no es necesario memorizar todos los desplazamientos de QAOU de la totalidad de profundidades de división en la sección de memorización de información de desplazamiento 621, que se utilizan en el dispositivo de decodificación de vídeo 1, no existe la necesidad de garantizar 341 zonas de memoria y, en su lugar, es suficiente si se proporciona el mismo número de zonas de memoria que las de QAOUs que se utilizan realmente en la estructura especificada por la información de QAOU de entrada. Cuando la profundidad de división máxima es cuatro, el número de bloques es 256 o menor y, por lo tanto, la provisión de 256 memorias de mapa, y 256 memorias de lista, resulta suficiente. En este caso, puesto que los índices QAOU utilizados en la sección de memorización de información de desplazamiento 621, se utilizan índices únicos para identificar QAOU de hoja del árbol, a modo de ejemplo, índices de 0 a 255, o más pequeños, que se incrementan en uno cada vez que se decodifica un QAOU de hoja del árbol. Es posible, además, memorizar en la sección de memorización de información de desplazamiento 621, una lista de mapa utilizando información de QAOU correspondiente a la profundidad de división máxima como una unidad. En este caso, se utilizan 0 a 255 como los números de índice QAOU, usados en la sección de memorización de información de desplazamiento 621 y que corresponden a números de QAOMU 85 a 340 en la Figura 7.

Las partes (a) a (e) de la Figura 7 ilustran ejemplos de números de QAOMU que se adjuntan a QAOMUs incluidas en la unidad de procesamiento (a modo de ejemplo, una LCU). La parte (a) de la Figura 7 muestra un número QAOMU asignado a una QAOMU que tiene una profundidad de división de 0. La parte (b) de la Figura 7 lustra números de QAOMU asignados a QAOMUs que tienen una profundidad de división de uno. La parte (c) de la Figura 7 muestra números de QAOMU asignados a QAOMUs que tienen una profundidad de división de dos. La parte (d) de la Figura 7 ilustra números de QAOMU asignados a QAOMUs que tienen una profundidad de división de tres. La parte (e) de la Figura 7 muestra números de QAOMU asignados a QAOMUs que tienen una profundidad de división de cuatro.

El controlador de QAOU 622 controla los componentes individuales incluidos en el procesador de filtro de desplazamiento adaptativo 62. El controlador de QAOU 622 se refiere, además, a la información de estructura de QAOU y a continuación, divide la imagen decodificada desbloqueada `P_DB` en uno, o una pluralidad de QAOUs, y escanea los QAOUs individuales en un orden predeterminado. El controlador de QAOU 622 proporciona, además, un número QAOMU que representa una QAOMU objeto, para la sección de determinación de tipo de desplazamiento 623.

La sección de determinación de tipo de desplazamiento 623 se refiere a la memoria de mapa y la memoria de lista de la sección de memorización de información de desplazamiento 621 y determina el tipo de desplazamiento especificado por el número QAOMU que se proporciona desde el controlador de QAOU 622. La sección de determinación de tipo de desplazamiento 623 proporciona, además, el tipo de desplazamiento determinado a la sección de clasificación 624.

La sección de clasificación 624 clasifica cada píxel incluido en el QAOU objeto como una de las clases que se pueden seleccionar para el tipo de desplazamiento suministrado procedente de la sección de determinación de tipo de desplazamiento 623. La sección de clasificación 624 proporciona, además, el tipo de desplazamiento y el índice de clase que indica la clase de cada píxel a la sección de determinación de desplazamiento 625. Un procesamiento de clasificación específico, realizado por la sección de clasificación 624, se examinará más adelante y, por lo tanto, se omite aquí una explicación del mismo.

La sección de determinación de desplazamiento 625 se refiere a la memoria de lista de la sección de memorización de información de desplazamiento 621 y determina, para cada píxel incluido en el QAOU objeto, un desplazamiento especificado por el tipo de desplazamiento y el índice de clase que se proporciona por la sección de clasificación 624. La sección de determinación de desplazamiento 625 incluye un desplazador inverso de desplazamiento (no ilustrado) que realiza un desplazamiento a la izquierda a nivel de bits en un desplazamiento en una cantidad igual a

un valor de desplazamiento establecido por la sección de establecimiento de atributos de desplazamiento 613. El desplazador de desplazamiento opuesto realiza una cuantización inversa en el desplazamiento, de modo que la profundidad de bits de desplazamiento puede coincidir con la profundidad de bits de píxel. Realizando dicha cuantización inversa, además del procesamiento realizado por el sumador de desplazamiento 626, que se explicará más adelante, se puede añadir el desplazamiento al valor de píxel utilizando la misma profundidad de bit. El desplazamiento cuantizado inversamente para cada píxel se proporciona al sumador de desplazamiento 626.

El sumador de desplazamiento 626 añade el desplazamiento suministrado desde la sección de determinación de desplazamiento 625 al píxel asociado que forma el QAOU objeto de la imagen decodificada desbloqueada P_DB. El sumador de desplazamiento 626 proporciona, a la salida, una imagen obtenida realizando el procesamiento en todos los QAOUs incluidos en la imagen decodificada desbloqueada P_DB como una imagen decodificada de desplazamiento filtrado P_OF.

La Figura 8 es un diagrama de flujo de un flujo de procesamiento realizado por el procesador de filtro de desplazamiento adaptativo 62.

(Etapa S101)

En primer lugar, el controlador de QAOU 622 obtiene información de estructura QAOU procedente de la sección de memorización de información de desplazamiento 621.

(Etapa S102)

A continuación, el controlador de QAOU 622 inicia un primer bucle utilizando el número QAOMU de una QAOMU objeto, como una variable de bucle.

(Etapa S103)

El controlador de QAOU 622 proporciona el número QAOMU a la sección de determinación de tipo de desplazamiento 623. Bajo el control del controlador de QAOU 622, la sección de determinación de tipo de desplazamiento 623 realiza la lectura de un tipo de desplazamiento especificado por el número QAOMU que se proporciona desde el controlador de QAOU 622, desde la memoria de mapa y la memoria de lista, de la sección de memorización de información de desplazamiento 621. La sección de determinación de tipo de desplazamiento 623 proporciona, además, la lectura del tipo de desplazamiento a la sección de clasificación 624.

(Etapa S104)

A continuación, el controlador de QAOU 622 inicia un segundo bucle utilizando el número de píxel de cada píxel incluido en la QAOMU objeto como una variable de bucle. Los números de píxel se utilizan para distinguir, entre sí, los píxeles incluidos en la QAOMU y se pueden utilizar, a modo de ejemplo, números asignados a píxeles incluidos en la QAOMU objeto en un orden de exploración predeterminado. En lugar de utilizar los números de píxel, las coordenadas x e y de cada píxel incluido en la QAOMU objeto, se pueden usar como variables de bucle.

(Etapa S105)

Luego, bajo el control del controlador de QAOU 622, la sección de clasificación 624 clasifica el píxel objeto como una de entre una pluralidad de clases que se pueden seleccionar para el tipo de desplazamiento que se proporciona desde la sección de determinación de tipo de desplazamiento 623. La sección de clasificación 624 proporciona, además, el tipo de desplazamiento y el índice de clase que indica la clase del píxel objeto a la sección de determinación de desplazamiento 625.

(Etapa S106)

A continuación, bajo el control del controlador de QAOU 622, la sección de determinación de desplazamiento 625 efectúa la lectura de un desplazamiento que ha de añadirse al píxel objeto procedente de la sección de memorización de información de desplazamiento 621. Es decir, la sección de determinación de desplazamiento 625 realiza la lectura del desplazamiento especificado por el tipo de desplazamiento y el índice de clase que se proporcionan desde la sección de clasificación 624. La sección de determinación de desplazamiento 625 realiza una cuantización de manera inversa, además, el desplazamiento ejecutando un desplazamiento a la izquierda a nivel de bits en el desplazamiento determinado para el píxel objeto en una cantidad igual a un valor de desplazamiento suministrado desde la sección de establecimiento de atributos de desplazamiento 613. La sección de determinación de desplazamiento 625 proporciona, entonces, el desplazamiento cuantizado inversamente al sumador de desplazamiento 626.

(Etapa S107)

Luego, bajo el control del controlador de QAOU 622, el sumador de desplazamiento 626 añade el desplazamiento que se proporciona desde de la sección de determinación de desplazamiento 625 al valor de píxel del píxel objeto que forma la imagen decodificada desbloqueada P_DB.

5 (Etapa S108)

Esta etapa es la terminación del segundo bucle.

(Etapa S109)

10

Esta etapa es la terminación del primer bucle.

Si en la etapa S103 se encuentra que el desplazamiento leído por la sección de determinación de tipo de desplazamiento 623 es 0 (tipo de desplazamiento=0), el controlador de QAOU 622 controla el sumador de desplazamiento 626 de modo que el sumador de desplazamiento 626 no añada un desplazamiento a cualquiera de los píxeles que forman la QAOMU objeto.

Si se encuentra en la etapa S105 que el píxel objeto se clasifica como clase 0 (índice de clase=0), el controlador de QAOU 622 controla el sumador de desplazamiento 626 de modo que el sumador de desplazamiento 626 no añada un desplazamiento al píxel objeto.

20 (El número de bits necesarios para memorizar el desplazamiento)

La cantidad de bits necesarios para memorizar los desplazamientos (sao_offset) se explicará a continuación. Cuando la profundidad de bits del píxel es de diez bits y la profundidad de bits de desplazamiento es de nueve bits, el desplazamiento puede adquirir un margen de valores de -2^9 a 2^9-1 y, el número de bits por desplazamiento es tan grande como diez bits. Si se decodifica un desplazamiento que tiene un número tan grande de bits, es necesario que la sección de memorización de información de desplazamiento 621 tenga el siguiente tamaño máximo de memoria para memorizar desplazamientos para cada imagen:

30 (el número total de QAOMUs por imagen) x (el número de clases) x (el número de bits por desplazamiento) = $256 \times 16 \times 10$ (bits) = 40960 (bits).

Aunque el número total de QAOMUs por imagen es de 256 en el dispositivo de decodificación, se utilizan 341 QAOMUs en un dispositivo de codificación, lo que se analizará más adelante y, por lo tanto, se requiere una memoria aún más grande. De este modo, si el margen de valor que se puede tomar con la profundidad de bits de desplazamiento se utiliza sin restringir el margen de valores de desplazamiento, se requiere un gran tamaño de memoria para memorizar desplazamientos, puesto que cada desplazamiento tiene un gran número de bits.

40 (Relación entre SAO_DEPTH y Eficiencia de Codificación)

SAO_DEPTH y PIC_DEPTH están estrechamente relacionados entre sí en términos de errores de cuantización. La profundidad de bits de una salida de imagen procedente del filtro de desplazamiento adaptativo 60 se representa mediante la profundidad de bits de píxel PIC_DEPTH y, SAO_DEPTH representa la profundidad de bits de un desplazamiento que se añade a un píxel. En consecuencia, incluso si se utiliza un desplazamiento que tiene una más alta precisión que la de la profundidad de bits de píxel, se descarta en el proceso de salida. Por lo tanto, es preferible que SAO_DEPTH, que indica la precisión del desplazamiento, se establezca para que sea igual, o menor, a PIC_DEPTH. Por otro lado, si SAO_DEPTH es menor que PIC_DEPTH, una imagen de entrada se corrige simplemente con una precisión menor que la precisión (PIC_DEPTH) con la que la imagen de entrada podría corregirse utilizando un filtro, disminuyendo así el efecto de filtrado.

Por el contrario, si la precisión de desplazamiento SAO_DEPTH es más alta, aumenta la cantidad de datos necesarios para codificar los desplazamientos. En general, como puede deducirse desde del hecho de que la eficiencia de la codificación se optimiza al minimizar el coste de distorsión de tasa $D + \lambda R$, representado utilizando la cantidad de datos (tasa) R requerida para codificación, la distorsión D de una imagen de entrada, y la ponderación λ , la precisión de desplazamiento disminuye la distorsión D y aumenta la tasa R. Por lo tanto, en relación con el nivel de precisión, existe una compensación entre una disminución en la distorsión D y un aumento en la tasa R en la que se proporciona un valor óptimo específico de la precisión.

Además, en esta forma de realización, al restringir un desplazamiento cuantizado a un margen de valores de desplazamiento que se puede representar por una determinada anchura de bits, es posible reducir la anchura de bit para memorizar un desplazamiento cuantizado en la sección de memorización de información de desplazamiento 621. Con esta disposición, se puede obtener el efecto de reducir el tamaño de la memoria, en comparación con la restricción no impuesta descrita con anterioridad. Sin embargo, si el margen de valores de desplazamiento está excesivamente restringido, disminuye el efecto de corregir la distorsión de una imagen decodificada mediante el uso de un desplazamiento, y no es posible eliminar la distorsión de una imagen decodificada en el procesamiento de

adición de desplazamiento, lo que puede disminuir la eficiencia de codificación. Por lo tanto, es preferible que el margen de valores de desplazamiento se establezca en un margen óptimo en un grado tal que no disminuya la eficiencia de codificación.

- 5 La sección de establecimiento de atributos de desplazamiento 613 establece la profundidad de bits de desplazamiento y el valor de desplazamiento de desplazamiento utilizando uno de los siguientes patrones S1 a S6, y establece el margen de valores de desplazamiento utilizando uno de los patrones siguientes C1 a C3. Solo los patrones en Z S2 y C1 afectan a la presente invención.

10 (Patrón S1)

En el patrón S1, tal como se muestra en la parte (a) de la Figura 9, la profundidad de bits de desplazamiento SAO_DEPTH se establece para que sea igual a la profundidad de bits de píxel PIC_DEPTH. Puesto que el valor máximo de la precisión de desplazamiento es igual a la profundidad de bits de píxel, se codifica un desplazamiento con la máxima precisión en el patrón S1.

(Patrón S2)

En el patrón S2, según se ilustra en la parte (b) de la Figura 9, cuando PIC_DEPTH es diez bits o más pequeño, SAO_DEPTH se establece para ser igual a PIC_DEPTH y, cuando PIC_DEPTH es once bits o mayor, SAO_DEPTH se establece para ser diez bits. En el patrón S2, el límite superior de la profundidad de bits de desplazamiento se establece en diez bits. Los inventores han encontrado que, cuando el etapa de cuantización QP de una imagen decodificada es pequeño (cuando la tasa binaria es alta), la eficiencia de codificación se vuelve mayor si la profundidad de bits de desplazamiento es igual a la profundidad de bits de píxel, que en un caso en el que la profundidad de bits de desplazamiento es menor que la profundidad de bits de píxel y que, de forma inversa, cuando el etapa de cuantización QP es grande, la eficiencia de codificación es mayor si la profundidad de bits de desplazamiento es menor que la profundidad de bits de píxel, que en un caso en el que la profundidad de bits de desplazamiento es igual a la profundidad de bits de píxel. Se ha encontrado a través de los experimentos de los inventores que, cuando el parámetro de cuantización qp oscila desde 12 a 27, se mejora la eficiencia de codificación si las profundidades de bits de desplazamiento se determinaron como en el patrón S2, en comparación con un caso en el que la profundidad de bits de desplazamiento fue de ocho bits cuando la profundidad de bits de píxel era de nueve bits o más pequeña, y la profundidad de bits de desplazamiento era de nueve bits cuando la profundidad de bits de píxel era de diez bits o mayor. En consecuencia, en el patrón S2, al cambiar la dependencia de la profundidad de bits de desplazamiento en la profundidad de bits de píxel, en el límite de diez bits, se puede disminuir la cantidad de datos necesaria para codificar desplazamientos, en comparación con un caso en el que la profundidad de bits de desplazamiento es igual a la profundidad de bits del píxel, como en el patrón S1, con lo que se consigue una alta eficiencia de codificación.

(Patrón S3)

En el patrón S3, tal como se muestra en la parte (c) de la Figura 9, cuando PIC_DEPTH es de nueve bits o más pequeño, SAO_DEPTH se establece para ser igual a PIC_DEPTH y, cuando PIC_DEPTH es de diez bits o mayor, SAO_DEPTH se establece para ser de nueve bits. En el patrón S3, el límite superior de la profundidad de bits de desplazamiento se establece en nueve bits. Tanto en el patrón S3 como en el patrón S2, se puede reducir la cantidad de datos necesarios para codificar los desplazamientos, logrando así una alta eficiencia de codificación.

(Patrón S4)

En el patrón S4, según se ilustra en la parte (d) de la Figura 9, cuando PIC_DEPTH es de diez bits o más pequeño, SAO_DEPTH se establece para ser igual a PIC_DEPTH y, cuando PIC_DEPTH es de once bits o mayor, SAO_DEPTH se establece para ser 10 niveles $((PIC_DEPTH-10)/STEP)$. La función de nivel $\text{floor}(x)$ es una función que proporciona el mayor número entero no superior a x. En el patrón S4, cuando la profundidad de bits de píxel es de diez bits o mayor, cada vez que el bit de píxel aumenta (disminuye) en STEP bits, la profundidad de bits de desplazamiento aumenta (disminuye) en un bit. La parte (e) de la Figura 9 ilustra un caso en el que STEP es dos en el patrón S4 y, cada vez que la profundidad de bits de píxel aumenta en dos bits, la profundidad de bits de desplazamiento aumenta en un bit. Con esta configuración, es posible responder de manera más flexible al nivel de la tasa binaria que en el patrón S2 o S3, al mismo tiempo que se considera el nivel de la profundidad de bits de píxel.

Para todos patrones entre el patrón S1 al patrón S4, el valor de desplazamiento está representado por el valor de diferencia $PIC_DEPTH - SAO_DEPTH$ entre PIC_DEPTH y SAO_DEPTH. Mediante el cambio de la profundidad de bits de desplazamiento y el valor de desplazamiento como en el patrón S1 a S4 anteriormente descritos, la profundidad de bits de desplazamiento y el valor de desplazamiento se pueden establecer sin aumentar el tamaño de la memoria, o la cantidad de procesamiento, lo que hace posible mejorar la eficiencia de codificación.

(Patrón S5)

En el patrón S5, la profundidad de bits de desplazamiento se codifica, de forma explícita. Más concretamente, se codifica la diferencia entre la profundidad de bits de desplazamiento y un valor predeterminado. Es conveniente que el valor predeterminado sea ocho o la profundidad de bits de píxel. Si el valor predeterminado es ocho, se codifica $SAO_DEPTH - 8$. Si el valor predeterminado es la profundidad de bits de píxel, se codifica $PIC_DEPTH - SAO_DEPTH$. La profundidad de bits de desplazamiento se puede codificar, de forma segura, como varios elementos de información de parámetro, o la cabecera de datos codificados, o como parte de información de QAOU. Si la profundidad de bits se codifica como parte de información de QAOU, se puede incluir, de forma segura, en una hoja del árbol QAOU o en información de QAOU, de un nivel predeterminado (a modo de ejemplo, el nivel más alto de QAOU o el primer nivel de QAOU). Al codificar la profundidad de bits como parte de los datos codificados, la profundidad de bits se puede establecer para que sea un valor óptimo en el dispositivo de decodificación, y en el dispositivo de codificación, con lo que se obtiene el efecto de maximizar la eficiencia de codificación. Si la profundidad de bits se codifica como información de QAOU, la profundidad de bits se modifica de conformidad con la profundidad de QAOU sao_curr_depth , lo que permite reducir el tamaño de la memoria para la memorización de desplazamientos. Es probable que aparezcan numerosos desplazamientos cuando sao_curr_depth sea grande. En consecuencia, al configurar la profundidad de bits para que sea pequeña cuando sao_curr_depth es grande, y al establecer la profundidad de bits para ser grande cuando sao_curr_depth es pequeña, es posible reducir el tamaño de memoria requerido. A modo de ejemplo, es conveniente que, cuando $sao_curr_depth=0$ a 1, la profundidad de bits de desplazamiento se establezca para que sea igual a la profundidad de bits de píxel (patrón S1) y cuando $sao_curr_depth=2$ a 4, se establece el límite superior para la profundidad de bits de desplazamiento (patrón S2). La anchura de bit se codifica de este modo. Es adecuado, además, que un indicador que indica si la anchura del bit de desplazamiento se codificará en respuesta a la profundidad de QAOU, o solamente se utilizará una anchura de bits de desplazamiento sin tener en cuenta si la profundidad de QAOU se puede codificar, cambiando así entre un caso en el que la anchura de bits de desplazamiento se codifica en respuesta a la profundidad de QAOU, y un caso en el que solamente se utiliza una anchura de bits de desplazamiento independientemente de la profundidad de QAOU.

(Patrón S6)

Sin realizar la codificación, de forma explícita, de la profundidad de bit, la profundidad de bits se determina de conformidad con sao_curr_depth . A modo de ejemplo, es conveniente que, cuando $sao_curr_depth=0$ a 1, la profundidad de bits de desplazamiento se establece para ser igual a la profundidad de bits de píxel (patrón S1), y cuando $sao_curr_depth=2$ a 4, se establece el límite superior para la profundidad de bits de desplazamiento (patrón S2).

(Patrón C1)

En el patrón C1, el margen de valores de desplazamiento se establece de conformidad con SAO_DEPTH . En adelante, la longitud binaria máxima que representa el margen de valores de desplazamiento se indica mediante $CLIP_BIT$. Más concretamente, al calcular $CLIP_BIT = SAO_DEPTH - K$, el margen de valores de desplazamiento está determinado para ser -2^{CLIP_BIT-1} a $2^{CLIP_BIT-1}-1$. Se ha encontrado a través de los experimentos de los inventores que es adecuado $K=4$. Es decir, se ha validado que, en el caso de $K=4$, la eficiencia de la codificación no disminuyó, incluso si el margen de desplazamiento estaba restringido por el margen de valores de desplazamiento. $K=4$ es adecuado cuando la profundidad de bits de píxel es ocho, lo que se utiliza más comúnmente. Cuando la profundidad de bits de píxel es ocho, la profundidad de bits de desplazamiento SAO_DEPTH también es ocho y, por lo tanto, $CLIP_BIT=8-K=4$. Si se puede memorizar un desplazamiento utilizando cuatro bits, a modo de ejemplo, en un software que gestiona un byte constituido por ocho bits como la unidad, un desplazamiento se puede empaquetar y memorizarse en un byte, reduciendo así el tamaño de la memoria de forma fácil.

(Patrón C2)

En el patrón C2, el margen de valores de desplazamiento se establece independientemente de SAO_DEPTH . Más específicamente, $CLIP_BIT$ se establece para ser ocho y, el margen de valores de desplazamiento se establece para ser de -2^7 a 2^7-1 .

Como alternativa, en condiciones normales, se puede utilizar una constante N que es independiente de SAO_DEPTH y, $CLIP_BIT$ se puede establecer para que sea N . Cuando el margen de valores de desplazamiento se establece, con independencia de SAO_DEPTH , es preferible que el margen de valores de desplazamiento se establezca para ser más pequeño que la profundidad de bits de desplazamiento con el fin de conseguir el efecto de reducir el tamaño de la memoria.

(Patrón C3)

En el patrón C3, el margen de valores de desplazamiento se determina de conformidad con la profundidad jerárquica de QAOU. Es conveniente que, cuando sao_curr_depth sea pequeño (a modo de ejemplo, 0 a 1), el margen de valores de desplazamiento se determine independientemente de la profundidad de bits de desplazamiento, y cuando sao_curr_depth sea grande (a modo de ejemplo, 2 a 4), el margen de valores de desplazamiento se determine de

conformidad con la profundidad de bits de desplazamiento. Si el margen de valores de desplazamiento se determina con independencia de la profundidad de bits de desplazamiento, CLIP_BIT se puede establecer para que sea ocho (patrón C2). Si el margen de valores de desplazamiento se determina de conformidad con la profundidad de bits de desplazamiento, CLIP_BIT puede establecerse para ser SAO_DEPTH-K bits (patrón C1). Si la longitud binaria de desplazamiento se determina de conformidad con la profundidad jerárquica de QAOU, se puede utilizar, de forma segura, un número de bit fijo para CLIP_BIT. A modo de ejemplo, resulta adecuado CLIP_BIT=4.

(Primer ejemplo específico del número de bits de desplazamiento)

A continuación, se describe un primer ejemplo específico del número de bits de un desplazamiento (sao_offset) de conformidad con esta forma de realización. En este ejemplo, se analizará un caso en el que la profundidad de bits de píxel se establece en diez bits y, el valor de desplazamiento se establece utilizando el patrón S2, y el margen de valores de desplazamiento se establece utilizando el patrón C1. En el patrón S2, la profundidad de bits de desplazamiento se establece para ser diez bits y, en el patrón C1, el margen de valores de desplazamiento se establece para ser $10-4=6$ bits. Si el número de bits por desplazamiento es seis bits, significa que los valores de desplazamiento están restringidos desde -32 a 31. En este ejemplo, como el tamaño de memoria para la memorización de desplazamientos, en la sección de memorización de información de desplazamiento 621, es suficiente el siguiente tamaño máximo para cada la imagen:

(el número total de QAOMUs por imagen) x (el número de clases) x (el número de bits por desplazamiento) = $256 \times 16 \times 6$ (bits) = 24576 (bits).

Por lo tanto, con la configuración en la que el margen de valores de desplazamiento se establece como en este ejemplo, el tamaño de memoria necesario para la sección de memorización de información de desplazamiento 621 se puede reducir a prácticamente 3/5 del tamaño de memoria de un ejemplo de la técnica relacionada.

Puesto que se puede reducir la cantidad de datos requeridos para codificar desplazamientos incluidos en los datos codificados nº 1, es posible mejorar la eficiencia de codificación. Además, puesto que no se añaden excesivos desplazamientos, se puede garantizar un nivel adecuado de calidad de imagen.

(Segundo ejemplo específico del número de bits de desplazamiento)

Un segundo ejemplo específico del número de bits de un desplazamiento (sao_offset), de conformidad con esta forma de realización, se explicará a continuación. En este ejemplo, se analiza un caso en el que la profundidad de bits de píxel se establece para ser diez bits y, el valor de desplazamiento se establece utilizando el patrón S2, y el margen de valores de desplazamiento se establece utilizando el patrón C2. En el patrón S2, la profundidad de bits de desplazamiento se establece en diez bits y, en el patrón C2, el margen de valores de desplazamiento se establece para ser ocho bits. Si el número de bits por desplazamiento es ocho bits, ello significa que los valores de desplazamiento están restringidos desde -128 a 127. En este ejemplo, puesto que el tamaño de memoria para memorizar desplazamientos en la sección de memorización de información de desplazamiento 621, el siguiente tamaño máximo para cada la imagen es suficiente:

(el número total de QAOMUs por imagen) x (el número de clases) x (el número de bits por desplazamiento) = $256 \times 16 \times 8$ (bits) = 32768 (bits).

En consecuencia, con la configuración en la que el número de bits de un desplazamiento se establece como en este ejemplo, el tamaño de memoria necesario para la sección de memorización de información de desplazamiento 621 se puede reducir a prácticamente un 4/5 del tamaño de memoria de un ejemplo de la técnica relacionada.

Puesto que se puede reducir la cantidad de datos requeridos para codificar desplazamientos que se incluyen en los datos codificados nº 1, es posible mejorar la eficiencia de codificación. Además, puesto que no se añaden excesivos desplazamientos, se puede garantizar un nivel adecuado de calidad de imagen.

Si el número de bits de desplazamiento está restringido, se puede reducir la cantidad de datos requeridos para codificar la información de desplazamiento incluida en los datos codificados nº 1. Sin embargo, si el número de bits de desplazamiento está excesivamente restringido, disminuye el efecto de proporcionar un filtro de desplazamiento adaptativo, lo que aumenta la cantidad de datos necesarios para codificar datos residuales (el valor de píxel de una imagen residual) que se incluyen en los datos codificados.

(Tercer ejemplo específico del número de bits de desplazamiento)

Se describe, a continuación, un tercer ejemplo específico del número de bits de un desplazamiento (sao_offset). En cualquiera de los patrones C1 y C2 de este ejemplo, el valor que ha de establecerse para el número de bits de un desplazamiento se cambia dependiendo de si el tipo de desplazamiento es un desplazamiento de borde (el tipo de desplazamiento es uno de 1 a 4), o un desplazamiento de banda (el tipo de desplazamiento es 5 o 6). En este ejemplo, se da a conocer un método para establecer el valor de desplazamiento utilizando el patrón S2, y para

establecer el margen de valores de desplazamiento de un desplazamiento de borde utilizando el patrón C2, y el margen de valores de desplazamiento de un desplazamiento de banda utilizando el patrón C1, cuando se establece la profundidad de bits de píxel para ser diez bits.

5 Cuando la profundidad de bits del píxel es de diez bits, en el patrón S2, la profundidad de bits de desplazamiento se establece en diez bits. El margen de valor de un desplazamiento que pertenece al tipo de desplazamiento de borde (en lo sucesivo denominado como un "desplazamiento de borde") se establece en ocho bits en el patrón C2. El
10 margen de valor de un desplazamiento que pertenece al tipo de desplazamiento de banda (en lo sucesivo denominado como un "desplazamiento de banda") se establece para ser seis bits en el patrón C1. Más en general, cuando el número de bits de un desplazamiento de borde es N bits, y el número de bits de un desplazamiento de banda es M bits, los números de bits de desplazamientos se determinan de modo que se cumpla la condición $N \geq M$.

15 El tamaño de memoria requerido para ser asegurado en la sección de memorización de información de desplazamiento 621 es el número de bits representado por (el número de clases de un tipo de desplazamiento $\times$ el número de bits de desplazamiento) para cada QAOU. En consecuencia, el número de bits de un desplazamiento de banda, que tiene un mayor número de clases que un desplazamiento de borde, se establece para que sea más pequeño, lo que hace posible utilizar, de forma eficaz, la zona de memoria para memorizar desplazamientos en la sección de memorización de información de desplazamiento 621.

20 De este modo, al variar el número de bits de un desplazamiento en función del tipo de desplazamiento, se puede mejorar la eficiencia de codificación sin requerir un tamaño de memoria excesivo de la sección de memorización de información de desplazamiento 621. Además, la zona de memoria se puede utilizar más eficazmente

25 Cuando el umbral th para valores de restricción que se pueden tomar como un desplazamiento es mayor que 2^{m-1} , pero no mayor que 2^m , se puede utilizar el método de codificación/decodificación de longitud fija de m-bits como método de codificación para la codificación del desplazamiento. Se puede utilizar el método de codificación/decodificación de longitud variable, tal como la codificación unitaria truncada o la codificación de corte truncado que tiene th como el valor máximo. El valor máximo th descrito anteriormente se determina por el margen de valores de desplazamiento que se proporciona por la sección de configuración de atributo de desplazamiento
30 613. El dispositivo de decodificación de vídeo 1 es capaz de decodificar desplazamientos que han sido codificados según se describió anteriormente.

35 Con la configuración descrita anteriormente, en la sección de configuración de atributo de desplazamiento 613, se establece la profundidad de bits de desplazamiento, el margen de valores de desplazamiento y el valor de desplazamiento. En el decodificador de información del filtro de desplazamiento adaptativo 61, se decodifica un desplazamiento cuantizado que tiene un valor dentro del margen de valores de desplazamiento, y se memoriza en la sección de memorización de información de desplazamiento 621, en la que se garantiza para cada desplazamiento una zona de memorización que tiene una anchura de bits igual o mayor que la gama de valores de desplazamiento. Esta forma de realización, se caracteriza por cuanto que el margen de valores de desplazamiento se determina de
40 conformidad con la profundidad de bits de desplazamiento. La profundidad de bits de desplazamiento se determina de conformidad con la profundidad de bits de píxels. Por consiguiente, esta forma de realización está caracterizada, además, porque la profundidad de bits de desplazamiento se determina en función de la profundidad de bits de píxel.

45 El decodificador de información de filtro de desplazamiento adaptativo 61, de la primera forma de realización, puede incluir una sección de memorización para memorizar desplazamientos decodificados en la misma, y una sección de cuantización inversa que realiza una cuantización, de forma inversa, desplazamientos obtenidos desde de la sección de memorización. El proceso de cuantización inversa, realizado por la sección de determinación de desplazamiento 625 puede omitirse. Este caso, se caracteriza por cuanto que la sección de memorización memoriza un
50 desplazamiento que está restringido al margen de valores de desplazamiento establecido por la sección de configuración de atributo de desplazamiento 613 y por cuanto que la sección de cuantización inversa realiza el procesamiento de cuantización inversa efectuando un desplazamiento a la izquierda a nivel de bits, en un desplazamiento de conformidad con el valor de desplazamiento establecido por la sección de establecimiento de atributos de desplazamiento 613.

55 La sección de decodificación de información de desplazamiento 611 decodifica, desde de los datos codificados $n^{\circ} 1$, cada desplazamiento al que debe hacer referencia el sumador de desplazamiento 626, que añade un desplazamiento al valor de píxel de cada píxel que forma una imagen de entrada constituida por una pluralidad de zonas unitarias. Dicho de otro modo, la sección de decodificación de información de desplazamiento 611 incluye un
60 medio de decodificación de desplazamiento (no ilustrado), para decodificar un desplazamiento que tiene un margen de valores de desplazamiento y un valor de desplazamiento que se establecen de conformidad con la profundidad de bits de píxel y que está restringido al margen de valores de desplazamiento.

65 El filtro de desplazamiento adaptativo 60, de esta forma de realización, es un dispositivo de filtrado de imagen que añade un desplazamiento al valor de píxel de cada píxel que forma una imagen de entrada constituida por una pluralidad de zonas unitarias. En otras palabras, el filtro de desplazamiento adaptativo 60 puede ser un dispositivo

de filtrado de imagen que incluye la sección de establecimiento de atributos de desplazamiento 613 que se refiere a la información de especificación de tipo de desplazamiento incluida en datos codificados, y establece atributos de desplazamiento en una zona unitaria objeto, la sección de decodificación de información de desplazamiento 611 que decodifica un desplazamiento que tiene una anchura de bit correspondiente a un margen de valores de desplazamiento incluido en los atributos de desplazamiento establecidos tal como se describió con anterioridad, y el sumador de desplazamiento 626 que añade el desplazamiento al valor de píxel de cada píxel que forma la imagen de entrada.

Además del medio de decodificación de desplazamiento descrito anteriormente, la sección de decodificación de información de desplazamiento 611 puede incluir un medio de determinación para determinar el tipo de desplazamiento al que pertenece una zona unitaria objeto, entre una pluralidad de tipos de desplazamiento, y un medio de decodificación de desplazamiento para decodificar desplazamientos que tienen diferentes anchuras de bits dependiendo del tipo de desplazamiento determinado por el medio de determinación.

La información de especificación de tipo de desplazamiento anteriormente descrita incluye información relativa a la profundidad de bits de valores de píxel en cada zona unitaria de la imagen de entrada, y la sección de decodificación de información de desplazamiento 611 puede decodificar un desplazamiento que tenga una anchura de bit correspondiente a la profundidad de bits de los valores de píxel.

A continuación, se proporciona una descripción de ejemplos específicos de procesamiento de clasificación realizado por la sección de clasificación 624. Es preferible que la sección de clasificación 624 realice, entre los siguientes ejemplos de procesamiento de clasificación, el procesamiento de clasificación que corresponde al procesamiento de clasificación realizado por el dispositivo de codificación de vídeo que ha generado los datos codificados nº 1.

(Primer ejemplo de procesamiento de clasificación de la sección de clasificación 624)

A continuación, se describe un primer ejemplo de procesamiento de clasificación realizado por la sección de clasificación 624, haciendo referencia a las partes (a) a (d) de la Figura 10 a la Figura 12.

(Cuando el tipo de desplazamiento es uno de entre 1 a 4 (desplazamiento de borde))

Cuando el tipo de desplazamiento proporcionado desde de la sección de determinación de tipo de desplazamiento 623 es uno de entre 1 a 4, la sección de clasificación 624 determina si existe, o no, un borde próximo a un píxel objeto y, si existe un borde, determina el tipo de borde. La sección de clasificación 624 clasifica, entonces, el píxel objeto como una de entre una pluralidad de clases de conformidad con los resultados de determinación.

Más concretamente, la sección de clasificación 624 determina, en primer lugar, el signo de la diferencia entre el valor de píxel $pic[x]$ de un píxel x objeto, y cada uno de los valores de píxel $pic[a]$ y $pic[b]$ de dos píxeles, a y b , que son adyacentes al píxel objeto, o que tienen el mismo vértice que el píxel objeto. Es decir, la sección de clasificación 624 calcula:

$$Sign(pic[x] - pic[a])$$

y

$$Sign(pic[x] - pic[b]):$$

en donde $Sign(z)$ es una función que toma los siguientes valores:

$$Sign(z) = +1 \quad (\text{cuando } z > 0)$$

$$Sign(z) = 0 \quad (\text{cuando } z = 0)$$

$$Sign(z) = -1 \quad (\text{cuando } z < 0).$$

Más concretamente, una determinación de qué píxeles se utilizan como el píxel a y el píxel b depende del tipo de desplazamiento y se realiza como sigue.

- Cuando el tipo de desplazamiento es 1 ($sao_type_idx=1$)

Tal como se ilustra en la parte (a) de la Figura 10, el píxel situado inmediatamente en el lado izquierdo del píxel objeto x , se establece para ser el píxel a , y el píxel situado inmediatamente en el lado derecho del píxel objeto x , se establece para ser el píxel b .

- Cuando el tipo de desplazamiento es 2 ($sao_type_idx=2$)

Según se ilustra en la parte (b) de la Figura 10, el píxel situado inmediatamente en la parte superior del píxel objeto x, se establece para ser el píxel a, y el píxel situado inmediatamente en la parte inferior del píxel objeto x, se establece para ser el píxel b.

• Cuando el tipo de desplazamiento es 3 (sao_type_idx=3)

Tal como se muestra en la parte (c) de la Figura 10, el píxel que tiene el mismo vértice que el del lado superior izquierdo del píxel objeto x, se establece para ser el píxel a, y el píxel que tiene el mismo vértice que el del lado inferior derecho del píxel objeto x, se establece para ser el píxel b.

• Cuando el tipo de desplazamiento es 4 (sao_type_idx=4)

Según se muestra en la parte (d) de la Figura 10, el píxel que tiene el mismo vértice que el lado inferior izquierdo del píxel objeto x, se establece para ser el píxel a, y el píxel que tiene el mismo vértice que el del lado superior derecho del píxel objeto x, se establece para ser el píxel b.

La parte (a) de la Figura 11 ilustra gráficos que indican las relaciones de magnitud entre el valor de píxel pic[x] del píxel objeto x, y el valor de píxel del píxel a o b y, además, muestra los valores de la función Sign de conformidad con la relación de magnitud. En los gráficos que se muestran en la parte (a) de la Figura 11, el círculo negro con pic[x] indica el valor de píxel del píxel objeto x, y el círculo negro sin pic[x] indica el valor de píxel del píxel objeto a o b. La dirección vertical en los gráficos que se ilustran en la parte (a) de la Figura 11 indica la magnitud del valor de píxel.

A continuación, la sección de clasificación 624 encuentra EdgeType de conformidad con la siguiente ecuación matemática (1-1) sobre la base de Sign(pic[x]-pic[a]) y Sign(pic[x]-pic[b]).

$$\text{EdgeType} = \text{Sign}(\text{pic}[x] - \text{pic}[a]) + \text{Sign}(\text{pic}[x] - \text{pic}[b]) + 2 \dots (1-1)$$

La parte (b) de la Figura 11 ilustra gráficos que indican las relaciones de magnitud entre el valor de píxel del píxel objeto x, y cada uno de los valores de píxel del píxel a y el píxel b y muestra, además, los valores de EdgeType de conformidad con la relación de magnitud. En la parte (b) de la Figura 11, el círculo negro central en cada gráfico indica el valor de píxel del píxel objeto x, y los círculos negros en ambos lados indican los valores del píxel a y el píxel b. La dirección vertical en los gráficos ilustrados en la parte (b) de la Figura 11 indica la magnitud del valor de píxel.

A continuación, la sección de clasificación 624 determina lo que sigue, sobre la base del EdgeType determinado, el índice de clase (class_idx) de la clase a la que pertenece el píxel objeto x.

$$\text{class_idx} = \text{EoTbl}[\text{EdgeType}]$$

en donde EoTbl [EdgeType] es una tabla de transformación utilizada para determinar el class_idx desde EdgeType. Un ejemplo específico de la tabla de transformación EoTbl se muestra en la parte (d) de la Figura 11.

Tal como se muestra en la parte (d) de la Figura 11, cuando no existe un borde en una zona constituida por el píxel objeto x, el píxel a y, el píxel b (en adelante, este caso se referirá, además, como un caso en el que una zona es plana), el píxel objeto x se clasifica como clase 0 (class_idx=0). La parte (c) de la Figura 11 muestra la asociación entre cada gráfico que se ilustra en la parte (b) de la Figura 11 y class_idx.

(Cuando el tipo de desplazamiento es 5 o 6 (desplazamiento de banda))

Cuando el tipo de desplazamiento que se proporciona desde de la sección de determinación de tipo de desplazamiento 623 es 5 o 6, la sección de clasificación 624 clasifica el valor de píxel del píxel objeto x como una de entre una pluralidad de clases de conformidad con el valor de píxel pic[x] del píxel objeto x.

• Cuando el tipo de desplazamiento es 5 (sao_type_idx=5)

Cuando el valor de píxel pic[x], del píxel objeto x, satisface: $(\text{max} \times 1/4) \leq \text{pic}[x] \leq (\text{max} \times 3/4)$, la sección de clasificación 624 clasifica el píxel objeto como una clase distinta de 0. Es decir, cuando el valor de píxel del píxel objeto está incluido en un margen indicado por la parte sombreada en la parte (a) de la Figura 12, la sección de clasificación 624 clasifica el píxel objeto como una clase distinta de 0. En la condición descrita anteriormente, max indica el valor máximo que puede tomar el píxel objeto x y, a modo de ejemplo, max=255. Cuando max=255, la condición anteriormente descrita puede representarse por $8 \leq \text{pic}[x]/8 \leq 23$.

- Cuando el tipo de desplazamiento es 6 (sao_type_idx=6)

Cuando el valor de píxel $\text{pic}[x]$ del píxel objeto x , satisface: $\text{pic}[x] \leq (\text{max} \times 1/4)$ o $(\text{max} \times 3/4) \leq \text{pic}[x]$, la sección de clasificación 624 clasifica el píxel objeto como una clase distinta de 0. Es decir, cuando el valor de píxel del píxel objeto está dentro de un margen indicado por la parte sombreada en la parte (b) de la Figura 12, la sección de clasificación 624 clasifica el píxel objeto como clase distinta de 0. En la condición descrita con anterioridad, max indica el valor máximo que puede tomar el píxel objeto x y, a modo de ejemplo, $\text{max}=255$. Cuando $\text{max}=255$, la condición descrita anteriormente se puede representar por $(\text{pic}[x]/8) \leq 7$ o $24 \leq (\text{pic}[x]/8)$.

El procesamiento de clasificación realizado por la sección de clasificación 624 se describirá, a continuación, más concretamente.

Cuando el tipo de desplazamiento es 5 o 6, la sección de clasificación 624 determina el índice de clase (class_idx) de la clase a la que pertenece el píxel objeto x , como sigue.

```
class_idx = EoTbl[sao_type_idx][pic[x]/8]
```

en donde $\text{EoTbl}[\text{sao_type_idx}][\text{pic}[x]/8]$ es una tabla de transformación utilizada para la determinación de class_idx desde el valor de píxel $\text{pic}[x]$ del píxel objeto x , y sao_type_idx . Un ejemplo específico de la tabla de transformación EoTbl se ilustra en la Figura 12. En la parte (c) de la Figura 12, "BO_1" indica que $\text{sao_type_index}=5$, y "BO_2" indica que $\text{sao_type_index}=6$.

Según se muestra en la parte (c) de la Figura 12, cuando $\text{sao_type_index}=5$, la sección de clasificación 624 clasifica el píxel objeto x como uno de los índices de clase 1 a 16 de conformidad con la magnitud de $\text{pic}[x]$ si el píxel el valor $\text{pic}[x]$ del píxel de objeto x cumple la condición $8 \leq (\text{pic}[x]/8) \leq 23$.

Cuando $\text{sao_type_index}=6$, la sección de clasificación 624 clasifica el píxel objeto x como uno de los índices de clase 1 a 16 de conformidad con la magnitud de $\text{pic}[x]$ si el valor de píxel $\text{pic}[x]$ del píxel objeto x , satisface la condición $(\text{pic}[x]/8) \leq 7$ o $24 \leq (\text{pic}[x]/8)$.

En general, cuando la profundidad de bits de una imagen es PIC_DEPTH , el valor máximo del píxel objeto se puede representar por $\text{max}=2^{\text{PIC_DEPTH}-1}$ y, por lo tanto, la clasificación se realiza utilizando $\text{pic}/2^{(\text{PIC_DEPTH}-5)}$ en lugar de $\text{pic}/8$ en la parte (c) de la Figura 12.

(Segundo ejemplo de procesamiento de clasificación de la sección de clasificación 624)

A continuación, se describirá un segundo ejemplo de procesamiento de clasificación realizado por la sección de clasificación 624.

En este ejemplo de procesamiento, la sección de clasificación 624 determina el EdgeType utilizando la siguiente expresión matemática (1-2) en lugar de la expresión matemática (1-1). Las otras características son similares a las del primer ejemplo del procesamiento de clasificación.

```
EdgeType = Sign((pic[x]>>shift) - (pic[a]>>shift))
+ Sign((pic[x]>>shift) - (pic[b]>>shift)) + 2
... (1-2)
```

en donde ">>" indica un desplazamiento a la derecha a nivel de bits y, "desplazamiento" indica la magnitud del desplazamiento binario. El valor específico de "desplazamiento" se puede determinar de modo que tenga una correlación positiva con la profundidad de bits del valor de píxel.

En el procesamiento de clasificación del primer ejemplo, incluso cuando el gradiente del valor de píxel es muy pequeño, si no es 0, el valor de Sign no es 0. En consecuencia, en el procesamiento de clasificación del primer ejemplo, el valor de EdgeType es vulnerable a la influencia del ruido.

En este ejemplo del procesamiento, después de realizar un desplazamiento a la derecha a nivel de bits en el valor de píxel, se calcula la diferencia entre los valores de píxel. Por lo tanto, el valor de EdgeType es menos vulnerable a la influencia del ruido, logrando así el efecto de mejorar la eficiencia de codificación.

En este ejemplo del procesamiento, se puede utilizar la siguiente expresión matemática (1-3) en lugar de la expresión matemática (1-2).

```

EdgeType = Sign((pic[x]-pic[a])>>shift) + Sign((pic[x]-
pic[b])>>shift)) + 2
... (1-3)

```

Es decir, una vez calculada la diferencia entre los valores de píxel, se puede realizar el desplazamiento a la derecha a nivel de bits. Al usar la expresión matemática (1-3), se pueden conseguir ventajas similares a las obtenidas al utilizar la expresión matemática (1-2).

(Tercer ejemplo de procesamiento de clasificación de la sección de clasificación 624)

A continuación, se describe un tercer ejemplo de procesamiento de clasificación realizado por la sección de clasificación 624.

En este ejemplo del procesamiento, la definición de la función Sign, descrita en el primer ejemplo del procesamiento de clasificación calculado por la sección de clasificación 624, se cambia del modo siguiente. Las otras características son similares a las del primer ejemplo del procesamiento de clasificación.

```

Sign(z) = +1 (cuando z > th)

Sign(z) = 0 (cuando -th ≤ z ≤ th)

Sign(z) = -1 (cuando z < -th)

```

en donde th indica un umbral que tiene un valor predeterminado. El valor específico del umbral th se puede determinar de tal modo que el valor absoluto del umbral th tenga una correlación positiva con la profundidad de bits del valor de píxel.

En este ejemplo de procesamiento, también, el valor de EdgeType es menos vulnerable a la influencia del ruido, con lo que se consigue una alta eficiencia de codificación.

(Cuarto ejemplo de procesamiento de clasificación de la sección de clasificación 624)

A continuación, se describirá un cuarto ejemplo de procesamiento de clasificación realizado por la sección de clasificación 624.

En este ejemplo del procesamiento, la sección de clasificación 624 utiliza EoTbl[sao_type_idx][pic[x]/8] que se ilustra en la Figura 13 en lugar de EoTbl[sao_type_idx][pic[x]/8] que se muestra en la parte (c) de la Figura 12.

Tal como se ilustra en la Figura 13, en este ejemplo del procesamiento, cuando el valor de pic[x]/8 es 8 o 9, el píxel objeto x se clasifica como una clase que tiene un índice de clase distinto de 0, independientemente de si sao_type_index=5 o sao_type_index=6. Además, cuando el valor de pic[x]/8 es 22 o 23, el píxel objeto x se clasifica como una clase que tiene un índice de clase distinto de 0, con independencia de si sao_type_index=5 o sao_type_index=6.

En este ejemplo del procesamiento, cuando el valor de píxel de un píxel objeto es 15 o menor (cuando pic[x]/8 es 0 o 1), el píxel objeto se recorta a MIN. Además, cuando el valor de píxel de un píxel objeto es 240 o mayor (cuando pic[x]/8 es 30 o 31), el píxel objeto se recorta a MAX. Como MIN y MAX, se utiliza, preferiblemente, una de las siguientes combinaciones.

MIN=15, MAX=240

MIN=16, MAX=239

MIN=16, MAX=235

En el primer ejemplo del procesamiento de clasificación, un píxel objeto que se clasifica como de clase 0 cuando sao_type_index=5, se clasifica como una clase distinta de 0 cuando sao_type_index=6. Además, un píxel objeto que se clasifica como de clase 0 cuando sao_type_index=6, se clasifica como una clase distinta de 0 cuando sao_type_index=5.

En consecuencia, en el primer ejemplo del procesamiento de clasificación, el valor de píxel al que se añade un desplazamiento puede ser muy diferente dependiendo de si sao_type_index=5 o sao_type_index=6, lo que puede dar lugar a un problema en el que la eficiencia de codificación no mejora de forma suficiente según estaba previsto.

Un problema de este tipo puede ser evidente cuando $\text{pic}[x]/8$ del valor de píxel de una imagen que aún no ha sido sometida a filtrado de desplazamiento es 8 o 9, o 22, o 23.

En este ejemplo de procesamiento, cuando $\text{pic}[x]/8$ es 8 o 9, el píxel objeto x se clasifica como una clase distinta de 0, independientemente de si $\text{sao_type_index}=5$ o $\text{sao_type_index}=6$. Además, cuando $\text{pic}[x]/8$ es 22 o 23, el píxel objeto x se clasifica como una clase distinta de 0, independientemente de si $\text{sao_type_index}=5$ o $\text{sao_type_index}=6$. En consecuencia, es poco probable que suceda el problema descrito con anterioridad. Por lo tanto, al realizar el procesamiento de este ejemplo, se puede mejorar la eficiencia de codificación.

En este ejemplo del procesamiento, en un caso en el que el valor de $\text{pic}[x]/8$ es 8 o 9, y en un caso en el que el valor de $\text{pic}[x]/8$ es 22 o 23, el píxel objeto x se clasifica como una clase distinta de 0, con independencia de si $\text{sao_type_index}=5$ o $\text{sao_type_index}=6$. Sin embargo, lo que antecede no es una limitación para la puesta en práctica de este ejemplo del procesamiento. En resumen, el procesamiento se realiza de modo que, cuando el valor de $\text{pic}[x]/8$ está dentro de un margen predeterminado, el píxel objeto x se clasifica como una clase distinta de 0, independientemente de si $\text{sao_type_index}=5$ o $\text{sao_type_index}=6$.

En general, cuando la profundidad de bits de píxel de una imagen es PIC_DEPTH , el valor máximo del píxel objeto se puede representar por $\text{max}=2^{\text{PIC_DEPTH}-1}$ y, por lo tanto, la clasificación se realiza utilizando $\text{pic}/2^{(\text{PIC_DEPTH}-5)}$ en lugar de $\text{pic}/8$ en la Figura 13.

De este modo, cuando el valor de píxel, del píxel objeto descrito anteriormente, está dentro de un margen predeterminado, la sección de clasificación 624 de este ejemplo puede clasificar el píxel objeto como una clase de desplazamiento en la que se añadirá un desplazamiento, independientemente de si el tipo de desplazamiento de una zona unitaria, que contiene el píxel objeto, es el primer tipo de desplazamiento descrito con anterioridad, o el segundo tipo de desplazamiento descrito anteriormente.

(Dispositivo de codificación de vídeo 2)

A continuación, se proporciona una descripción, haciendo referencia a las Figuras 14, a las partes (a) a (d) de la Figura 18, del dispositivo de codificación de vídeo 2 que genera datos codificados nº 1 mediante la codificación de una imagen objeto. El dispositivo de codificación de vídeo 2 incluye, de forma parcial, un método definido en H. 264/MPEG-4. AVC, un método utilizado en el software KTA, que es un códec de desarrollo conjunto en VCEG (Grupo de Expertos de Codificación de Vídeo), un método utilizado en el software TMuC (Modelo de Prueba bajo Consideración), que es un códec sucesor del códec utilizado en el software KTA y, una tecnología utilizada en el software HM (HEVC TestModel).

La Figura 14 es un diagrama de bloques que ilustra la configuración del dispositivo de codificación de vídeo 2 de esta forma de realización. El dispositivo de codificación de vídeo 2 incluye, según se ilustra en la Figura 14, una unidad de transformación y cuantización 21, un codificador de código de longitud variable 22, una unidad de transformación inversa y cuantización inversa 23, una memoria intermedia 24, una generador de imagen intra-predicción 25, un generador de imagen inter-predicción 26, un detector de vector de movimiento 27, un controlador de método de predicción 28, una unidad de eliminación de redundancia de vector de movimiento 29, un sumador 31, un subtrahedor 32, un filtro de desbloqueo 33, un filtro adaptativo 70 y un filtro de desplazamiento adaptativo 80. El dispositivo de codificación de vídeo 2 codifica una imagen de vídeo nº 10 (imagen objeto) con el fin de generar los datos codificados nº 1.

La unidad de transformación y cuantización 21 realiza: (1) Transformación de DCT (Transformada de Coseno Discreta) en cada bloque de una predicción residual D , obtenida mediante la sustracción de una imagen de predicción Pred de una imagen objeto, (2) realizar una cuantización de un coeficiente DCT obtenido realizando una transformación de DCT, y (3) proporcionando una predicción residual cuantizada QD obtenida mediante la realización de la cuantización por el codificador de código de longitud variable 22 y la unidad de transformación inversa y cuantización inversa 23. La unidad de transformación y cuantización 21 realiza, además: (1) la selección de una etapa de cuantización QP que se utilizará para la cuantización para cada bloque de árbol; (2) proporcionar una diferencia de parámetro de cuantización Δqp que representa el nivel de la etapa de cuantización QP seleccionada, al codificador de código de longitud variable 22; y (3) proporcionar la etapa de cuantización QP seleccionada a la unidad de transformación inversa y cuantización inversa 23. La diferencia de parámetro de cuantización Δqp es un valor de diferencia obtenido al restar el valor del parámetro de cuantización qp' , con respecto a un bloque de árbol que se ha sometido a la transformación de DCT y cuantización, inmediatamente antes de un bloque de árbol objeto, procedente del parámetro de cuantización qp ($QP=2^{qp/6}$) en relación con el bloque del árbol objeto que se someterá a la transformación de DCT y cuantización.

El codificador de código de longitud variable 22 genera datos codificados nº 1 realizando una codificación de longitud variable en: (1) la predicción cuantizada QD residual y Δqp , que se proporcionan desde de la unidad de transformación y cuantización 21; (2) un parámetro de cuantización PP suministrado por el controlador de método de predicción 28, que se explicará más adelante; y (3) un número de conjunto de filtros, un grupo de coeficientes de filtro, una información de especificación de zona e información de ON/OFF (activación/desactivación) proporcionada

por el filtro adaptativo 70, que se examina más adelante. El codificador de código de longitud variable 22 codifica, además, la información de QAOU suministrada por el filtro de desplazamiento adaptativo 80 e inserta la información de QAOU codificada en los datos codificados nº 3.

5 La unidad de transformación inversa y cuantización inversa 23 realiza: (1) la cuantización inversa del residual de predicción QD cuantizado, (2) transformación Inversa de DCT (Transformada de Coseno Discreta) en un coeficiente de DCT obtenido al realizar la cuantización inversa y, (3) el suministro de una predicción residual D obtenida mediante la ejecución de una DCT inversa, al sumador 31. Cuando se realiza una cuantización inversa la predicción cuantizada QD residual, se utiliza la etapa de cuantización QP que se proporciona desde la unidad de transformación y cuantización 21. La salida de residuo de predicción D, desde la unidad de transformación inversa y cuantización inversa 23 es la entrada de residuo de predicción D en la unidad de transformación y cuantización 21, a la que se añade un error de cuantización, sin embargo, para una descripción simple, se utilizará la misma denominación.

15 El generador de imagen intra-predicción 25 genera una imagen de predicción Pred_Intra con respecto a cada partición. Más concretamente, el generador de imagen intra-predicción 25, realiza: (1) a selección de un modo de predicción en relación con cada partición utilizada para predicción intra; y (2) la generación de la imagen de predicción Pred_Intra desde de la imagen decodificada P utilizando el modo de predicción seleccionado. El generador de imagen intra-predicción 25 proporciona la imagen de predicción generada Pred_Intra al controlador de método de predicción 28.

El generador de imagen intra-predicción 25 especifica, además, un índice de predicción PI relativo a cada partición desde un modo de predicción seleccionado, que se selecciona para la partición, y desde el tamaño de la partición, y a continuación, proporciona el índice de predicción PI al controlador de método de predicción 28.

25 El generador de imagen intra-predicción 25 suministra, además, el tamaño de una partición objeto e información de modo intra-predicción IEM, que es información que indica un modo de predicción asignado a la partición objeto, al filtro adaptativo 70.

30 El detector de vector de movimiento 27 detecta un vector de movimiento mv relacionado con a cada partición. Más concretamente, el detector de vector de movimiento 27 detecta un vector de movimiento mv relacionado con una partición objeto, mediante: (1) la selección de una imagen decodificada filtrada P_FL' para su uso como una imagen de referencia; y (2) la búsqueda de una zona que mejor se aproxime a la partición objeto en la imagen decodificada filtrada seleccionada P_FL'. La imagen decodificada filtrada P_FL' es una imagen obtenida al realizar el procesamiento de desbloqueo utilizando el filtro de desbloqueo 33, un procesamiento de desplazamiento adaptativo utilizando el filtro de desplazamiento adaptativo 80, y un procesamiento de filtrado adaptativo utilizando el filtro adaptativo 70, en una imagen decodificada. El detector de vector de movimiento 27 puede efectuar la lectura de los valores de píxel de pixels individuales que forman la imagen decodificada filtrada P_FL' desde la memoria intermedia 24. El detector de vector de movimiento 27 proporciona el vector de movimiento detectado mv al generador de imagen inter-predicción 26, y la unidad de eliminación de redundancia de vector de movimiento 29, junto con un índice de imagen de referencia RI, que especifica la imagen decodificada filtrada P_FL' que se utiliza como imagen de referencia. Con respecto a una partición en la que se realizará una predicción bidireccional (predicción ponderada), el detector de vector de movimiento 27 selecciona dos imágenes decodificadas filtradas P_FL1' y P_FL2' como imágenes de referencia, y proporciona vectores de movimiento mv1 y mv2 e índices de imagen de referencia RI1 y RI2, asociados con las dos imágenes decodificadas filtradas P_FL1' y P_FL2' al generador de imagen inter-predicción 26 y a la unidad de eliminación de redundancia de vector de movimiento 29.

El generador de imagen inter-predicción 26 genera una imagen de movimiento compensado mc en relación con cada partición inter-predicción. Más concretamente, el generador de imagen inter-predicción 26 genera una imagen de movimiento compensado mc desde de la imagen decodificada filtrada P_FL', especificada por el índice de imagen de referencia RI, que se proporciona desde del detector de vector de movimiento 27 mediante el uso del vector de movimiento mv suministrado desde el detector de vector de movimiento 27. Al igual que en el detector de vector de movimiento 27, el generador de imagen inter-predicción 26 puede realizar la lectura de valores de píxel de pixels individuales que forman la imagen decodificada filtrada P_FL' desde la memoria intermedia 24. El generador de imagen inter-predicción 26 proporciona la imagen de movimiento compensado mc generada (imagen inter-predicción Pred_Inter) al controlador de método de predicción 28, junto con el índice de imagen de referencia RI suministrado desde el detector de vector de movimiento 27. Con respecto a una partición en la que se realizará la predicción bidireccional (predicción ponderada), el generador de imagen inter-predicción 26, realiza: (1) la generación de una imagen de movimiento compensado mc1 desde de una imagen decodificada filtrada P_FL1', especificada por el índice de imagen de referencia RI1 utilizando un vector de movimiento mv1; (2) la generación de una imagen de movimiento compensado mc2 desde de una imagen decodificada filtrada P_FL2', que se especifica por el índice de imagen de referencia RI2 utilizando un vector de movimiento mv2; y (3) la generación de una imagen de inter-predicción Pred_Inter mediante la adición de un valor de desplazamiento a un promedio ponderado de la imagen de movimiento compensado mc1 y la imagen de movimiento compensado mc2.

El controlador de método de predicción 28 compara cada una de entre la imagen de intra-predicción Pred_Intra y la

imagen de inter-predicción Pred_Inter con una imagen objeto que se va a codificar y, determina si se debe realizar una predicción intra o predicción inter. Si se selecciona la predicción intra, el controlador de método de predicción 28 proporciona la imagen de intra-predicción Pred_Intra al sumador 31, y al substractor 32, como la imagen de predicción Pred y proporciona, además, el índice de predicción PI suministrado desde el generador de imagen intra-predicción 25 al codificador de código de longitud variable 22 como el parámetro de predicción PP. Por el contrario, si se selecciona la predicción inter, el controlador de método de predicción 28 proporciona la imagen de inter-predicción Pred_Inter al sumador 31, y al substractor 32, como la imagen de predicción Pred y proporciona, además, el índice de imagen de referencia RI, suministrado desde el generador de imagen de inter-predicción 26 y un índice de vector de movimiento de estimación PMVI y MVD residual de vector de movimiento, que se proporcionan desde la unidad de eliminación de redundancia de vector de movimiento 29 (que se examina más adelante) al codificador de código de longitud variable como los parámetros de predicción PP.

Mediante la sustracción de la imagen de predicción Pred, seleccionada en el controlador de método de predicción 28, de la imagen objeto que ha de codificarse, se genera un residuo de predicción D en el substractor 32. El residuo de predicción D, generado en el substractor 32 se somete a una transformación de DCT y cuantización por la unidad de transformación y cuantización 21, tal como se indicó anteriormente. Asimismo, al añadir la imagen de predicción Pred seleccionada en el controlador de método de predicción 28, al residuo de predicción D, generado en la unidad de transformación inversa y cuantización inversa 23, se genera una imagen decodificada, localmente, P en el sumador 31. La imagen decodificada localmente P, generada en el sumador 31, pasa a través del filtro de desbloqueo 33, el filtro de desplazamiento adaptativo 80 y el filtro adaptativo 70. A continuación, la imagen decodificada localmente P se memoriza en la memoria intermedia 24 como una imagen decodificada filtrada P_FL y se utiliza como una imagen de referencia para realizar una predicción inter.

La unidad de eliminación de redundancia de vector de movimiento 29 elimina una redundancia en un vector de movimiento mv, detectado por el detector de movimiento de vector 27. Más concretamente, la unidad de eliminación de redundancia de vector de movimiento 29, realiza: (1) la selección de un método de estimación utilizado para la estimación del vector de movimiento mv; (2) la determinación de un vector de movimiento de estimación pmv de conformidad con el método de estimación seleccionado; y (3) la generación de un MVD residual de vector de movimiento mediante la resta del vector de movimiento de estimación pmv del vector de movimiento mv. La unidad de eliminación de redundancia de vector de movimiento 29 proporciona un MVD residual de vector de movimiento al controlador de método de predicción 28, junto con el índice de vector de movimiento de estimación PMVI que indica el método de estimación seleccionado.

Cuando la diferencia entre los valores de píxel de los pixels adyacentes entre sí con un límite de bloque, o un límite de CU entre sí, en la imagen decodificada P es menor que un umbral predeterminado, el filtro de desbloqueo 33 realiza el procesamiento de desbloqueo en el límite de bloque, o el límite de CU, en la imagen decodificada P, suavizando así una imagen en la proximidad del límite de bloque o el límite de CU. La imagen sometida al procesamiento de desbloqueo por el filtro de desbloqueo 33 se envía al filtro de desplazamiento adaptativo 80 como una imagen decodificada desbloqueada P_DB.

El filtro de desplazamiento adaptativo 80 realiza un proceso de filtrado de desplazamiento adaptativo en la imagen decodificada desbloqueada P_DB, proporciona desde el filtro de desbloqueo 33, generando así una imagen decodificada de desplazamiento filtrado P_OF. La imagen decodificada de desplazamiento filtrado P_OF generada, se suministra al filtro adaptativo 70. La configuración específica del filtro de desplazamiento adaptativo 80 se explicará a continuación y, por lo tanto, se omite aquí su explicación.

El filtro adaptativo 70 realiza el procesamiento de filtrado adaptativo en la imagen decodificada de desplazamiento filtrado P_OF, que se proporciona desde el filtro de desplazamiento adaptativo 80, con lo que se genera una imagen decodificada filtrada P_FL. La imagen decodificada filtrada P_FL, sometida al proceso de filtrado por el filtro adaptativo 70, se memoriza en la memoria intermedia 24. Se determina un coeficiente de filtro utilizado por el filtro adaptativo 70, de modo que el error entre la imagen decodificada filtrada P_FL y la imagen objeto n° 10 se pueda minimizar, y el coeficiente de filtro determinado de este modo se codifica como un parámetro de filtro FP y se transmite al dispositivo de decodificación de vídeo 1.

(Filtro de desplazamiento adaptativo 80)

La Figura 15 es un diagrama de bloques que ilustra la configuración del filtro de desplazamiento adaptativo 80. El filtro de desplazamiento adaptativo 80 incluye, tal como se muestra en la Figura 15, una unidad de establecimiento de información de filtro de desplazamiento adaptativo 81, y un procesador de filtro de desplazamiento adaptativo 82.

La unidad de establecimiento de información de filtro de desplazamiento adaptativo 81 incluye, según se ilustra en la Figura 15, un calculador de desplazamiento 811, un desplazador de desplazamiento 816, una sección de recorte de desplazamiento 812, un selector de información de desplazamiento 813 y una sección de establecimiento de atributos de desplazamiento 815.

(Calculador de desplazamiento 811)

El calculador de desplazamiento 811 calcula desplazamientos con respecto a todos los tipos de desplazamiento y todas las clases para la totalidad de QAOMUs hasta una profundidad de división predeterminada, incluida en la unidad de procesamiento (a modo de ejemplo, una LCU). En este caso, los tipos de desplazamiento y las clases, son los mismos que los examinados en una descripción del dispositivo de decodificación de vídeo 1.

La Figura 16 es un diagrama de flujo de un flujo de procesamiento realizado por el calculador de desplazamiento 811.

(Etapa S201)

En primer lugar, el calculador de desplazamiento 811 inicia un primer bucle utilizando el número de QAOMU de una QAOMU objeto, como una variable de bucle. A modo de ejemplo, en las partes (a) a (e) de la Figura 7 se ilustra un ejemplo en el que el primer bucle es un bucle desde el número de QAOMU=0 hasta el número de QAOMU=340.

(Etapa S202)

A continuación, el calculador de desplazamiento 811 inicia un segundo bucle utilizando el tipo de desplazamiento que se puede seleccionar para la QAOMU objeto como una variable de bucle. El segundo bucle es un bucle desde el tipo de desplazamiento 1 al tipo de desplazamiento 6.

(Etapa S203)

Después, el calculador de desplazamiento 811 inicia un tercer bucle utilizando un píxel incluido en la QAOMU objeto como una unidad.

(Etapa S204)

A continuación, el calculador de desplazamiento 811 clasifica un píxel objeto en una de entre una pluralidad de clases. Más concretamente, cuando el tipo de desplazamiento, que es la variable de bucle del segundo bucle, es uno de entre 1 a 4, el calculador de desplazamiento 811 clasifica el píxel objeto en una de entre la clase 1 a la clase 4. El procesamiento de clasificación, en esta etapa, es el mismo procesamiento que el procesamiento de clasificación en uno de entre el primero al cuarto ejemplos, realizado por la sección de clasificación 624 del filtro de desplazamiento adaptativo 60 del dispositivo de decodificación de vídeo 1.

En lo que respecta a la QAOMU objeto, el calculador de desplazamiento 811 calcula, además, el conteo del número de clasificación [part_idx] [sao_type_idx] [class_idx], que es el número de veces que se clasifican los píxels para cada clase. En este número de clasificación, part_idx indica el número de QAOMU.

(Etapa S205)

Luego, el calculador de desplazamiento 811 calcula un valor de píxel de diferencia para el píxel objeto mediante el cálculo de la diferencia entre el valor de píxel del píxel objeto en la imagen decodificada P_DB, y el valor de píxel del píxel objeto en la imagen objeto nº 10, que ha de codificarse. Más concretamente, cuando la posición del píxel objeto está representada por (x, y), el calculador de desplazamiento 811 calcula P_DB (x, y) - Org (x, y). En esta expresión, P_DB (x, y) indica el valor de píxel del píxel objeto en la imagen decodificada desbloqueada P_DB y, Org (x, y) indica el valor de píxel del píxel objeto en la imagen objeto nº 10.

(Etapa S206)

Esta etapa es la terminación del tercer bucle. En el momento en que se completa esta etapa, se han calculado valores de píxel de diferencia para todos los píxels incluidos en la QAOMU objeto.

(Etapa S207)

A continuación, el calculador de desplazamiento 811 calcula un desplazamiento al dividir la suma de los valores de píxel de diferencia, de los píxels incluidos en la QAOMU objeto, para cada clase por el número de clasificación descrito anteriormente con respecto a esta clase. Más concretamente, el calculador de desplazamiento 811 calcula un desplazamiento, offset [part_idx] [sao_type_idx] [class_idx] para la QAOMU objeto, el tipo de desplazamiento objeto, y la clase objeto utilizando la siguiente ecuación.

$$\text{offset}[\text{part_idx}][\text{sao_type_idx}][\text{class_idx}] = \Sigma(\text{P_DB}(x, y) - \text{Org}(x, y)) / \text{count}[\text{part_idx}][\text{sao_type_idx}][\text{class_idx}],$$

en donde el símbolo Σ indica la suma de los píxels clasificados como la clase objeto especificada por class_idx en la

QAOMU objeto, que se especifica por part_idx, y el tipo de desplazamiento objeto, especificado por sao_type_idx.

(Etapa S208)

5 Esta etapa es la terminación del primer bucle.

(Etapa S209)

10 Esta etapa es la terminación del primer bucle.

Al realizar el procesamiento descrito con anterioridad, el calculador de desplazamiento 811 calcula desplazamientos con respecto a todos los tipos de desplazamiento, y todas las clases, para la totalidad de QAOMUs hasta una profundidad de división predeterminada incluida en una LCU objeto. A modo de ejemplo, en el caso del ejemplo ilustrado en las partes (a) a (e) de la Figura 7, el calculador de desplazamiento 811 calcula un total de 16368 desplazamientos, tal como se representa a continuación:

15 $((\text{el número total de QAOMUs en la profundidad de división 0}) + \dots + (\text{el número total de QAOMUs en la profundidad de división 4})) \times (\text{el número de tipos de desplazamiento de EO}) \times (\text{el número de clases de EO}) + (\text{el número de tipos de desplazamiento de BO}) \times (\text{el número de clases de BO}) = (1+4+16+64+256) \times ((4 \times 4) + (2 \times 16)) = 16368$. El número de bits de cada desplazamiento es, a modo de ejemplo, diez bits.

El calculador de desplazamiento 811 proporciona información de desplazamiento que indica los desplazamientos calculados por el procesamiento anteriormente descrito, tipos de desplazamiento, clases y la información de estructura de QAOU que indica la estructura de división de QAOU, al desplazador de desplazamiento 816.

25 En la etapa S204, el dispositivo de codificación de vídeo 2 puede codificar un indicador que indica qué tipo de procesamiento de clasificación se ha realizado y, luego, el filtro de desplazamiento adaptativo 60, del dispositivo de decodificación de vídeo 1, se puede referir a este indicador y puede realizar el mismo procesamiento de clasificación que el indicado por el indicador. Como alternativa, sin utilizar dicho indicador, se puede realizar, por anticipado, el mismo proceso de clasificación que se ha determinado en el dispositivo de codificación de vídeo 2, y el dispositivo de decodificación de vídeo 1.

(Desplazador de desplazamiento 816)

35 El desplazador de desplazamiento 816 realiza una cuantización en cada uno de los desplazamientos incluidos en la información de desplazamiento que se proporciona desde el calculador de desplazamiento 811. El desplazador de desplazamiento 816 realiza una cuantización en los desplazamientos realizando un desplazamiento a la derecha a nivel de bits en los desplazamientos, de modo que la precisión de los desplazamientos se puede transformar desde la profundidad de bits de píxel, a la profundidad de bits de desplazamiento. La magnitud en la que se desplaza un desplazamiento en el procesamiento correspondiente, se determina por el valor de desplazamiento que se proporciona desde la sección de establecimiento de atributos de desplazamiento 815, que se explicará más adelante.

(Sección de recorte de desplazamiento 812)

45 Con el fin de restringir los desplazamientos al margen de valores de desplazamiento suministrado desde la sección de establecimiento de atributos de desplazamiento 815, que se examinará más adelante, la sección de recorte de desplazamiento 812 realiza el procesamiento de recorte utilizando uno de entre el primer procesamiento de recorte y el segundo procesamiento de recorte, en los desplazamientos que se proporcionan desde el desplazador de desplazamiento 816.

(Primer procesamiento de recorte)

55 La sección de recorte de desplazamiento 812 realiza el procesamiento de recorte en cada uno de los desplazamientos incluidos en la información de desplazamiento que se proporciona desde el desplazador de desplazamiento 816. La sección de recorte de desplazamiento 812 recorta cada desplazamiento suministrado desde el desplazador de desplazamiento 816 a, por ejemplo, valores desde -8 a 7, expresando así cada desplazamiento por cuatro bits. Los desplazamientos recortados se proporcionan al selector de información de desplazamiento 813. La anchura de bit utilizado para el recorte se establece de conformidad con la profundidad de bits de la imagen y la profundidad de bits de desplazamiento, como en el dispositivo de decodificación de vídeo 1.

Al recortar cada desplazamiento de este modo, se puede reducir el tamaño de memoria de una memoria (no ilustrada) para memorizar cada desplazamiento. Además, se puede reducir la cantidad de datos requeridos para desplazamientos codificados, que se incluyen en los datos codificados n° 1, lo que hace posible mejorar la eficiencia de codificación. Además, puesto que no se añaden desplazamientos excesivos, se puede garantizar un nivel adecuado de calidad de imagen.

(Segundo procesamiento de recorte)

5 La sección de recorte de desplazamiento 812 puede cambiar el valor para el margen de recorte de un desplazamiento suministrado desde el desplazador de desplazamiento 816, en función del tipo de desplazamiento.

10 A modo de ejemplo, si el tipo de desplazamiento es un desplazamiento de borde, el número de bits de un desplazamiento se establece en ocho bits y, si el tipo de desplazamiento es un desplazamiento de banda, el número de bits de un desplazamiento se establece en cuatro bits. Más en general, cuando el número de bits de un desplazamiento de borde es N bits, y el número de bits de un desplazamiento de banda es M bits, los números de bits de desplazamientos se determinan de modo que se satisfaga la condición $N > M$.

15 De este modo, al variar el número de bits de un desplazamiento, en función del tipo de desplazamiento, puede mejorar la eficiencia de codificación sin requerir un tamaño de memoria excesivo de una memoria para almacenar cada desplazamiento.

20 Cuando el umbral th para valores de restricción que pueden tomarse como un desplazamiento, es mayor que 2^{m-1} , pero no mayor que 2^m , se puede utilizar el método de codificación de longitud fija de m-bit como el método de codificación para codificar el desplazamiento. Más concretamente, se puede utilizar la codificación unitaria truncada o la codificación de corte truncado teniendo th como el valor máximo. El valor máximo th , descrito anteriormente, está determinado por el margen de valores de desplazamiento suministrado desde de la sección de establecimiento de atributos de desplazamiento 815.

25 El procesamiento de recorte, realizado mediante una combinación del primer procesamiento de recorte y el segundo procesamiento de recorte, incluidos con anterioridad, también se incluye en esta forma de realización. El filtro de desplazamiento adaptativo 80 puede no incluir la sección de recorte de desplazamiento 812.

(Selector de información de desplazamiento 813)

30 El selector de información de desplazamiento 813 determina una combinación de un tipo de desplazamiento, una clase y un desplazamiento que minimizan el coste RD (costo de Distorsión-Tasa), y la estructura de división de QAOU asociada, y proporciona información de QAOU que indica los tipos de desplazamiento determinados, clases, y desplazamientos, y la estructura de división de QAOMU asociada, al codificador de código de longitud variable 22. El selector de información de desplazamiento 813 proporciona, además, los desplazamientos determinados para cada QAOU, o cada QAOMU, al procesador de filtro de desplazamiento adaptativo 82.

40 El procesamiento realizado por el selector de información de desplazamiento 813 se examinará de forma más específica con referencia a las Figuras 17 y 18. La Figura 17 es un diagrama de flujo de un flujo de procesamiento realizado por el selector de información de desplazamiento 813.

(Etapa S301)

45 El selector de información de desplazamiento 813 inicia un primer bucle utilizando el número de QAOMU de una QAOMU objeto como variable de bucle.

(Etapa S302)

50 A continuación, el selector de información de desplazamiento 813 inicia un segundo bucle utilizando el tipo de desplazamiento que se puede seleccionar para la QAOMU objeto como una variable de bucle. El segundo bucle es un bucle desde el tipo de desplazamiento 1 al tipo de desplazamiento 6.

(Etapa S303)

55 Luego, el selector de información de desplazamiento 813 calcula, para un tipo de desplazamiento objeto, el error cuadrático entre la imagen decodificada de desplazamiento filtrado P_OF , y la imagen codificada objeto n^o 10 con respecto a la QAOMU objeto.

(Etapa S304)

60 Esta etapa es la terminación del segundo bucle.

(Etapa S305)

65 Esta etapa es la terminación del primer bucle. En el momento en el que se completaron el primer y el segundo bucles, se calcularon los errores cuadrados relativos a cada QAOMU para todos los tipos de desplazamiento.

(Etapa S306)

A continuación, el selector de información de desplazamiento 813 determina, entre estructuras de división de QAOU que se pueden obtener al dividir la unidad de procesamiento objeto (a modo de ejemplo, una LCU) en QAOUs, la estructura de división de QAOU que minimiza el coste de RD.

Ejemplos específicos del procesamiento realizado por el selector de información de desplazamiento 813 en esta etapa, se describirán a continuación con referencia a las partes (a) a (d) de la Figura 18.

En primer lugar, el selector de información de desplazamiento 813 calcula el coste de RD cuando la profundidad de división es 0, y el coste de RD cuando la profundidad de división es 1 (parte (a) de la Figura 18). En la parte (a) de la Figura 18, se supone que el coste de RD de la profundidad de división 1 es menor que el de la profundidad de división 0 (parte (b) de la Figura 18).

A continuación, el selector de información de desplazamiento 813 calcula el coste de RD cuando la profundidad de división es 2 (parte (c) de la Figura 18).

Luego, el selector de información de desplazamiento 813 compara el coste de RD de una QAOMU de la profundidad de división 1, con el de QAOMUs de la profundidad de división 2, incluida en la QAOMU de la profundidad de división 1. Si el coste de RD de las QAOMUs de la profundidad de división 2 es menor que el de la QAOMU de la profundidad de división 1, el selector de información de desplazamiento 813 actualiza la QAOMU de la profundidad de división 1 a las QAOMUs de la profundidad de división 2 (parte (d) de la Figura 18). El selector de información de desplazamiento 813 repite este procesamiento hasta la mayor profundidad de división. De este modo, se puede determinar la estructura de división de QAOU que minimiza el coste de RD.

(Sección de establecimiento de atributos de desplazamiento 815)

La sección de establecimiento de atributos de desplazamiento 815 recibe la profundidad de bits de píxel (no ilustrada) y determina la profundidad de bits de desplazamiento. La sección de establecimiento de atributos de desplazamiento 815 establece, a continuación, el margen de valores de desplazamiento y el valor de desplazamiento, utilizando la profundidad de bits de desplazamiento determinada. El margen de valores de desplazamiento se suministra al procesador de filtro de desplazamiento adaptativo 82, y el valor de desplazamiento se proporciona al desplazador de desplazamiento 816. El establecimiento del margen de valores de desplazamiento, y el valor de desplazamiento, son el mismo procesamiento que el realizado por la sección de establecimiento de atributos de desplazamiento 613, descrito con anterioridad y, por lo tanto, se omite aquí su explicación-

(Procesador de filtro de desplazamiento adaptativo 82)

El procesador de filtro de desplazamiento adaptativo 82 añade un desplazamiento suministrado desde el selector de información de desplazamiento 813 a cada píxel de una QAOU objeto en la imagen decodificada desbloqueada P_DB. El procesador de filtro de desplazamiento adaptativo 82 genera, como una imagen decodificada de desplazamiento filtrado P_OF, una imagen obtenida mediante la realización del procesamiento en la totalidad de QAOUs incluidas en la imagen decodificada desbloqueada P_DB. La configuración del procesador de filtro de desplazamiento adaptativo 82 es la misma que la del procesador de filtro de desplazamiento adaptativo 62 y, por lo tanto, se omite aquí una explicación de la misma. Cada desplazamiento memorizado en una sección de memorización de información de desplazamiento (no ilustrada), incluida en el procesador de filtro de desplazamiento adaptativo 82 se limita al margen de valores de desplazamiento establecido por la sección de establecimiento de atributos de desplazamiento 815.

Segunda forma de realización

En la primera forma de realización, `sao_offset [sao_curr_depth] [ys] [xs] [i]` incluido en los datos codificados nº 1, es una sintaxis que representa un valor específico de un desplazamiento que ha de añadirse a cada píxel incluido en una QAOU objeto, en el proceso de filtrado de desplazamiento realizado utilizando el filtro de desplazamiento adaptativo.

Asimismo, los presentes inventores han encontrado que la cantidad de datos requeridos para codificar datos se puede reducir, todavía más, realizando una codificación predictiva en un valor de desplazamiento utilizado para el procesamiento de filtrado de desplazamiento, es decir, mediante la codificación de un residuo de desplazamiento que se calcula utilizando un valor de desplazamiento y un valor de predicción del valor de desplazamiento.

En una segunda forma de realización, se proporciona una descripción haciendo referencia a las Figuras 19 a 21 de un dispositivo de decodificación de vídeo que decodifica un desplazamiento que se ha sometido a codificación predictiva, y que realiza el procesamiento de filtrado de desplazamiento, y de un dispositivo de codificación de vídeo que realiza una codificación predictiva en un desplazamiento utilizado para el procesamiento de filtrado de desplazamiento. Las partes examinadas en la primera forma de realización no serán descritas.

(Datos codificados)

Los datos codificados de esta forma de realización incluyen un residuo de desplazamiento `sao_offset_residual[sao_curr_depth][ys][xs][i]` en lugar de `sao_offset[sao_curr_depth][ys][xs][i]` incluido en los datos codificados nº 1 de la primera forma de realización. Las otras partes de la configuración de los datos codificados de esta forma de realización son similares a las de los datos codificados nº 1 de la primera forma de realización. En lo sucesivo, los datos codificados de esta forma de realización se pueden indicar, además, como datos codificados nº 3.

(`sao_offset_residual`)

El residuo de desplazamiento `sao_offset_residual[sao_curr_depth][ys][xs][i]` representa un valor de diferencia ponderada entre un valor de desplazamiento que ha de añadirse a cada píxel incluido en una QAOU en el procesamiento de filtrado de desplazamiento, que se realiza por un filtro de desplazamiento adaptativo de esta forma de realización, y un valor de predicción del valor de desplazamiento y, además, se indica mediante `sao_offset_residual[sao_type_idx][class_idx]`.

Si un desplazamiento a añadirse a un píxel objeto incluido en una QAOU objeto, se indica mediante `Offset[sao_type_idx][class_idx]`, el residuo de desplazamiento `sao_offset_residual[sao_type_idx][class_idx]` se representa como sigue:

$$\begin{aligned} & \text{sao_offset_residual[sao_type_idx][class_idx]} \\ &= \text{Offset[sao_type_idx][class_idx]} \\ &- a * \text{pred_offset[merge_tbl[sao_type_idx]][class_idx]} \end{aligned}$$

en donde a es un coeficiente de ponderación que se multiplica por el valor de predicción `pred_offset` y `merge_tbl` es una función que utiliza `sao_type_idx` como un argumento. Ejemplos específicos de a y `merge_tbl` se examinarán más adelante y, por lo tanto, se omite aquí su explicación.

(Dispositivo de decodificación de vídeo)

El dispositivo de decodificación de vídeo de esta forma de realización incluye un filtro de desplazamiento adaptativo 60' en lugar del filtro de desplazamiento adaptativo 60 del dispositivo de decodificación de vídeo 1 de la primera forma de realización. Los otros elementos de la configuración del dispositivo de decodificación de vídeo, de esta forma de realización, son similares a los del dispositivo de decodificación de vídeo 1, de la primera forma de realización.

La Figura 19 es un diagrama de bloques que ilustra la configuración del filtro de desplazamiento adaptativo 60' de esta forma de realización. Tal como se ilustra en la Figura 19, el filtro de desplazamiento adaptativo 60' incluye una sección de decodificación de información de desplazamiento 611' en lugar de la sección de decodificación de información de desplazamiento 611 del filtro de desplazamiento adaptativo 60.

(Sección de decodificación de información de desplazamiento 611')

La sección de decodificación de información de desplazamiento 611' se refiere a información de QAOU incluida en los datos codificados nº 3, y decodifica información de desplazamiento `OI` incluida en la información de QAOU. Utilizando un residuo de desplazamiento `sao_offset_residual[sao_type_idx][class_idx]`, y un valor de predicción `pred_offset[merge_tbl[sao_type_idx]][class_idx]`, obtenido mediante la decodificación de la información de desplazamiento `OI`, la sección de decodificación de información de desplazamiento 611' calcula un desplazamiento `Offset[sao_type_idx][class_idx]` que ha de utilizarse para el procesamiento de filtrado de desplazamiento adaptativo, por la expresión siguiente:

$$\begin{aligned} & \text{Offset[sao_type_idx][class_idx]} \\ &= a * \text{pred_offset[merge_tbl[sao_type_idx]][class_idx]} \\ &+ \text{sao_offset_residual[sao_type_idx][class_idx]}. \end{aligned}$$

La sección de decodificación de información de desplazamiento 611' memoriza, entonces, el desplazamiento calculado `[sao_type_idx][class_idx]` en la sección de memorización de información de desplazamiento 621. En la expresión descrita anteriormente, `pred_offset[merge_tbl[sao_type_idx]][class_idx]` es un valor de predicción de `Offset[sao_type_idx][class_idx]` y, `merge_tbl[sao_type_idx]` representa una tabla para proporcionar un índice a `sao_type_idx=1` a 6, y uno o más elementos de `sao_type_idx` se pueden considerar como el mismo grupo.

(Primer ejemplo específico de pred_offset)

Un primer ejemplo específico de pred_offset [merge_tbl [sao_type_idx]] [class_idx] se describirá a continuación. En este ejemplo, el valor de predicción pred_offset [merge_tbl [sao_type_idx]] [class_idx] está determinado por:

$$\begin{aligned} & \text{pred_offset}[\text{merge_tbl}[\text{sao_type_idx}]][\text{class_idx}] \\ &= \text{Offset}'[\text{sao_type_idx}][\text{class_idx}] \end{aligned}$$

en donde Offset' [sao_type_idx] [class_idx] es un desplazamiento decodificado y representa un desplazamiento asociado con el índice de tipo de desplazamiento sao_type_idx, y el índice de clase class_idx.

De este modo, en este ejemplo, como el valor de predicción de Offset [sao_type_idx] [class_idx], se utiliza un desplazamiento Offset' [sao_type_idx] [class_idx], que es un desplazamiento decodificado asociado con el índice de tipo de desplazamiento sao_type_idx y el índice de clase class_idx.

(Segundo ejemplo específico de pred_offset)

A continuación, se describe un segundo ejemplo específico de pred_offset [merge_tbl [sao_type_idx]] [class_idx]. En este ejemplo, el valor de predicción pred_offset [merge_tbl [sao_type_idx]] [class_idx] se determina por:

$$\begin{aligned} & \text{pred_offset}[\text{merge_tbl}[\text{sao_type_idx}]][\text{class_idx}] \\ &= (\text{pred_offset}'[\text{merge_tbl}[\text{sao_type_idx}]][\text{class_idx}] * W1 + \\ & \text{Offset}'[\text{sao_type_idx}][\text{class_idx}] * W2 \gg \log_2(W1+W2)) \end{aligned}$$

en donde pred_offset' [merge_tbl [sao_type_idx]] [class_idx] indica un valor de predicción utilizado para calcular un desplazamiento decodificado Offset' [sao_type_idx] [class_idx], "*" indica un símbolo de operación que representa la multiplicación y, ">>" indica un desplazamiento a la derecha a nivel de bits. W1 y W2 indican los coeficientes ponderados y pueden tomar valores tales como W1=3 y W2=1. Se pueden determinar valores específicos de W1 y W2 de modo que se pueda aumentar la eficiencia de codificación.

Tal como se observa desde de la expresión descrita anteriormente, en este ejemplo, se hace referencia a valores de predicción decodificados y desplazamientos, de forma recursiva, del mismo modo que haciendo referencia a pred_offset' y Offset' con el fin de determinar pred_offset, se hace referencia a pred_offset'' y Offset'' de modo que se determine pred_offset', etc. En consecuencia, una pluralidad de desplazamientos decodificados contribuye a pred_offset, suprimiendo así las fluctuaciones excesivas en un valor de predicción. Con esta disposición, a modo de ejemplo, incluso si se ha calculado un valor de predicción inadecuado debido a la influencia del ruido, se puede suprimir la influencia de dicho valor de predicción inadecuado, haciendo posible, de este modo, la mejora de eficiencia de codificación.

(Tercer ejemplo específico de pred_offset)

Un tercer ejemplo específico de pred_offset [merge_tbl [sao_type_idx]] [class_idx] se examinará a continuación. En este ejemplo, el valor de predicción pred_offset [merge_tbl [sao_type_idx]] [class_idx] está determinado por:

$$\begin{aligned} & \text{pred_offset}[\text{merge_tbl}[\text{sao_type_idx}]][\text{class_idx}] \\ &= \text{clip3}(-\text{th}, \text{th}, \\ & \text{pred_offset}[\text{merge_tbl}[\text{sao_type_idx}]][\text{class_idx}]) \end{aligned}$$

en donde clip3 (A, B, C) indica que el valor C está recortado por el valor límite inferior A y el valor límite superior B. El argumento pred_offset [merge_tbl [sao_type_idx]] de clip3 se determina, a modo de ejemplo, como en el primero o segundo ejemplos específicos anteriormente descritos. El umbral th se determina como sigue dependiendo de la profundidad de bits bit_depth del valor de píxel.

$$\text{th} = 4 \quad (\text{bit_depth} = 8)$$

$$\text{th} = 8 \quad (\text{bit_depth} > 8)$$

De este modo, en este ejemplo, al utilizar un valor de predicción recortado por el valor límite superior y el valor límite inferior, no se generan valores de predicción excesivamente grandes y valores de predicción excesivamente pequeños, lo que hace posible mejorar la eficiencia de codificación. Los valores absolutos del valor límite superior y del valor límite inferior, se establecen de modo que se hacen grandes cuando el bit de un valor de píxel es grande.

Por lo tanto, el procesamiento de recorte adecuado se realiza de conformidad con la profundidad de bits de un valor de píxel, con lo que se evita la degradación de la calidad de imagen.

(Primer ejemplo específico de merge_tbl)

La parte (a) de la Figura 20 es una tabla que ilustra un primer ejemplo específico de la función merge_tbl [sao_type_idx]. Según se ilustra en la parte (a) de la Figura 20, cuando sao_type_idx=0, merge_tbl [sao_type_idx] de este ejemplo no toma ningún valor, y cuando sao_type_idx=1 a 6, merge_tbl [sao_type_idx] de este ejemplo toma un valor de 0 a 5, respectivamente. En consecuencia, merge_tbl [sao_type_idx] de este ejemplo se puede representar, además, como:

$$\text{merge_tbl}[\text{sao_type_idx}] = \text{sao_type_idx} - 1.$$

Utilizando merge_tbl [sao_type_idx] de este ejemplo, la sección de decodificación de información de desplazamiento 611' determina, de forma individual, un valor de predicción pred_offset para cada sao_type_idx, y para cada class_idx. Por lo tanto, es posible reducir la cantidad de datos requeridos para codificar residuos de desplazamiento sao_offset_residual.

(Segundo ejemplo específico de merge_tbl)

La parte (b) de la Figura 20 es una tabla que ilustra un segundo ejemplo específico de la función merge_tbl [sao_type_idx]. Tal como se muestra en la parte (b) de la Figura 20, en el caso de un desplazamiento de borde (sao_type_idx=1 a 4), merge_tbl [sao_type_idx] de este ejemplo toma el valor 0 y, en el caso de un desplazamiento de banda (sao_type_idx=5 o 6), merge_tbl [sao_type_idx] de este ejemplo toma 1 o 2, respectivamente.

A modo de ejemplo, si se han especificado sao_type_idx=1 (merge_tbl [sao_type_idx=1]=0) y class_idx=1, cuando se calcula el desplazamiento anterior, y si se han especificado sao_type_idx=2 (merge_tbl [sao_type_idx=2]=0) y class_idx=1 cuando se calcula el desplazamiento posterior, el valor de predicción que se utilizará para calcular el desplazamiento posterior es el mismo que el utilizado para calcular el desplazamiento anterior.

Mediante el uso de merge_tbl [sao_type_idx] de este ejemplo, la sección de decodificación de información de desplazamiento 611' realiza el siguiente procesamiento dependiendo de si se especifica un desplazamiento de borde o si se especifica un desplazamiento de banda.

- Cuando se especifica el desplazamiento del borde

El valor de predicción de un desplazamiento que ha de codificarse se calcula desde de un desplazamiento descodificado, clasificado como la misma clase que el desplazamiento a decodificar. En este caso, el tipo de desplazamiento del desplazamiento a decodificar, y el del valor de predicción, pueden ser diferentes siempre que la clase sea la misma. En consecuencia, un valor de predicción que se establece para calcular un desplazamiento de un determinado tipo de desplazamiento se puede utilizar para calcular un desplazamiento de un tipo de desplazamiento distinto de este determinado tipo de desplazamiento. Como resultado, se puede reducir la cantidad de procesamiento para el establecimiento de valores de predicción.

- Cuando se especifica el desplazamiento de banda

El valor de predicción de un desplazamiento que ha de codificarse se calcula desde de un desplazamiento que tiene el mismo tipo de desplazamiento, y se clasifica como la misma clase que las del desplazamiento a decodificar.

Al usar merge_tbl [sao_type_idx] de este ejemplo, se puede calcular un valor de predicción adecuado mientras se reduce la cantidad de procesamiento.

(Ejemplo Específico de Coeficiente a)

El coeficiente de ponderación a, que se multiplica por el valor de predicción pred_offset, puede ser 1 independientemente del tipo de desplazamiento, o puede variar en función del tipo de desplazamiento.

A modo de ejemplo, el coeficiente de ponderación a, puede variar como:

a=1 (en el caso de un desplazamiento de borde)

a=0.5 (en el caso de un desplazamiento de banda).

Más generalmente, si el coeficiente a a utilizarse cuando se especifica un desplazamiento de borde, se indica por a(borde), y si el coeficiente a, a utilizarse cuando se especifica un desplazamiento de banda, se indica por a(banda), se utiliza el coeficiente a, que satisface la siguiente condición:

a (borde)> a (banda).

Los presentes inventores han encontrado que la correlación entre un desplazamiento decodificado y un desplazamiento que se decodifica cuando se especifica un desplazamiento de borde, es mayor que cuando se especifica un desplazamiento de banda. En el ejemplo específico anteriormente descrito, la influencia de la correlación entre un desplazamiento decodificado y un desplazamiento que ha de decodificarse, puede reflejarse de forma adecuada, con lo que se reduce la cantidad de datos para codificar los residuos de desplazamiento.

(Dispositivo de codificación de vídeo)

El dispositivo de codificación de vídeo, de esta forma de realización, incluye un filtro de desplazamiento adaptativo 80' en lugar del filtro de desplazamiento adaptativo 80 del dispositivo de codificación de vídeo 2, de la primera forma de realización. Los otros elementos de la configuración del dispositivo de codificación de vídeo, de esta forma de realización, son similares a los del dispositivo de codificación de vídeo 2 de la primera forma de realización.

La Figura 21 es un diagrama de bloques que ilustra la configuración del filtro de desplazamiento adaptativo 80' de esta forma de realización. Tal como se ilustra en la Figura 21, el filtro de desplazamiento adaptativo 80' incluye una sección de determinación de residuo de desplazamiento 814 además de los componentes del filtro de desplazamiento adaptativo 60.

(Sección de determinación de residuo de desplazamiento 814)

La sección de determinación de residuo de desplazamiento 814 calcula un residuo de desplazamiento tomando la diferencia entre un desplazamiento proporcionado desde el selector de información de desplazamiento 813 y un valor de predicción del desplazamiento. El residuo de desplazamiento se codifica mediante el codificador de código de longitud variable 22 como parte de información de QAOU.

El valor de predicción establecido por la sección de determinación de residuo de desplazamiento 814 es similar al establecido por la sección de decodificación de información de desplazamiento 611' del dispositivo de decodificación de vídeo de esta forma de realización y, por lo tanto, se omite aquí una explicación de la misma.

(Apéndice 1)

La presente invención se puede describir como sigue.

Un dispositivo de filtrado de imagen, de conformidad con la presente invención, es un dispositivo de filtrado de imagen para la adición de un desplazamiento a un valor de píxel de cada píxel que forma una imagen de entrada, que está constituida por una pluralidad de zonas unitarias. El dispositivo de filtrado de imagen incluye: un medio de establecimiento de atributos de desplazamiento, para el establecimiento de atributos de desplazamiento para una zona unitaria objeto, haciendo referencia a la información de especificación de tipo de desplazamiento incluida en datos codificados; un medio de decodificación de desplazamiento para decodificar un desplazamiento que tiene una anchura de bit que corresponde a un margen de valores de desplazamiento incluido en los atributos de desplazamiento establecidos; y un medio de filtrado para añadir el desplazamiento al valor de píxel de cada píxel que forma la imagen de entrada.

Con el dispositivo de filtrado de imagen configurado tal como se describió con anterioridad, el medio de establecimiento de atributos de desplazamiento se refiere a la información de especificación de tipo de desplazamiento incluida en los datos codificados, y establece atributos de desplazamiento para la zona unitaria objeto, y el medio de decodificación de desplazamiento decodifica un desplazamiento que tiene una anchura de bit correspondiente a un margen de valores de desplazamiento incluido en los atributos de desplazamiento establecidos. Por lo tanto, es posible reducir, de forma efectiva, el tamaño de memoria de una memoria para almacenar desplazamientos.

En consecuencia, con la configuración descrita anteriormente, es posible realizar un procesamiento de filtrado de desplazamiento adecuado al mismo tiempo que se reduce el tamaño de memoria de una memoria para almacenar desplazamientos.

La información de especificación de tipo de desplazamiento se puede determinar para cada una de las imágenes de entrada, o para cada una de las zonas unitarias. Como alternativa, la información de especificación de tipo de desplazamiento se puede determinar para cada conjunto predeterminado de imágenes de entrada, o para cada conjunto predeterminado de las zonas unitarias.

La información de especificación de tipo de desplazamiento puede incluir, preferentemente, información relativa a una profundidad de bits del valor de píxel en cada zona unitaria que forma la imagen de entrada, y el medio de decodificación de desplazamiento puede decodificar, preferentemente, un desplazamiento que tenga una anchura

de bits correspondiente a la profundidad de bits del valor de píxel.

Con la configuración descrita anteriormente, puesto que el medio de decodificación de desplazamiento decodifica un desplazamiento que tiene una anchura de bit que corresponde a la profundidad de bits del valor de píxel, el tamaño de memoria de una memoria para almacenar desplazamientos se puede reducir de forma efectiva.

La información de especificación de tipo de desplazamiento puede incluir información relativa a una profundidad de bits del valor de píxel en cada zona unitaria que forma la imagen de entrada, y el medio de decodificación de desplazamiento puede decodificar un desplazamiento que tiene una anchura de bit que puede expresar un margen de valores correspondiente a la profundidad de bit.

Con la configuración anteriormente descrita, puesto que el medio de decodificación de desplazamiento decodifica un desplazamiento que tiene una anchura de bit que puede expresar un margen de valor que corresponde al valor de píxel en cada zona unitaria que forma la imagen de entrada, se puede reducir, de forma eficaz, el tamaño de memoria de una memoria para almacenar los desplazamientos.

La anchura de bit que puede expresar un margen de valor correspondiente a la profundidad de bit, es una anchura de bit cuando los valores incluidos en el margen de valor se representan en notación binaria. A modo de ejemplo, si el margen de valor es de -2^3 a 2^3-1 , la anchura de bit que puede expresar el margen de valor es de cuatro bits.

El desplazamiento decodificado descrito anteriormente puede ser, preferentemente, un valor cuantizado, y el medio de filtrado puede añadir, de forma preferible, un valor obtenido mediante cuantización inversa del desplazamiento utilizando un parámetro incluido en los atributos de desplazamiento al valor de píxel de cada píxel.

Con la configuración descrita con anterioridad, el desplazamiento decodificado es un valor cuantizado y, el medio de filtrado añade un valor obtenido mediante cuantización inversa del desplazamiento utilizando un parámetro incluido en los atributos de desplazamiento al valor de píxel de cada píxel. En consecuencia, se añade a cada valor de imagen un desplazamiento correspondiente a un parámetro incluido en los atributos de desplazamiento.

Por consiguiente, con la configuración descrita anteriormente, es posible mejorar la eficiencia de codificación mientras se reduce el tamaño de memoria de una memoria para almacenar desplazamientos.

La información que especifica el tipo de desplazamiento puede incluir información sobre un valor de desplazamiento del valor de píxel, y el medio de filtrado puede añadir un valor obtenido mediante cuantización inversa del desplazamiento utilizando el valor de desplazamiento en lugar de añadir el desplazamiento descrito anteriormente.

Con la configuración anteriormente descrita, la información de especificación de tipo de desplazamiento incluye información sobre un valor de desplazamiento del valor de píxel, y el medio de filtrado añade un valor obtenido mediante cuantización inversa del desplazamiento utilizando el valor de desplazamiento en lugar de añadir el desplazamiento descrito anteriormente, lo que permite obtener un desplazamiento correspondiente al valor de desplazamiento del valor de píxel. Por lo tanto, es posible mejorar la eficiencia de codificación al mismo tiempo que se reduce el tamaño de memoria de una memoria para almacenar desplazamientos.

El valor de desplazamiento del valor de píxel indica un valor de diferencia entre la profundidad de bits de píxel, y la profundidad de bits de desplazamiento y la cuantización inversa del desplazamiento mediante el uso del valor de desplazamiento, significa que el desplazamiento se somete a un desplazamiento a la izquierda a nivel de bits en una cantidad igual al valor de desplazamiento, por lo que se realiza la transformación de la profundidad de bits de desplazamiento a la profundidad de bits de píxel.

La información de especificación de tipo de desplazamiento se puede determinar, preferentemente, para la imagen de entrada.

Con la configuración descrita anteriormente, puesto que la información de especificación de tipo de desplazamiento se determina para la imagen de entrada, pudiendo realizar el dispositivo de filtrado de imagen el procesamiento de desplazamiento adecuado para la imagen de entrada.

Un dispositivo de decodificación de desplazamiento, de conformidad con la presente invención, es un dispositivo de decodificación de desplazamiento para decodificar cada desplazamiento al que se hace referencia mediante un filtro de imagen para añadir un desplazamiento a un valor de píxel de cada píxel que forma una imagen de entrada. El dispositivo de decodificación de desplazamiento incluye: un medio de decodificación de residuo de desplazamiento para decodificar cada residuo de desplazamiento desde de datos codificados; un medio de determinación del valor de predicción para determinar un valor de predicción de cada desplazamiento desde de un desplazamiento decodificado; y un medio de cálculo de desplazamiento para calcular cada desplazamiento desde de un valor de predicción determinado por el medio de determinación del valor de predicción, y un residuo de desplazamiento decodificado por el medio de decodificación de residuo de desplazamiento.

En el dispositivo de decodificación de desplazamiento, configurado según se describió con anterioridad, se proporciona el medio de decodificación de residuo de desplazamiento para la decodificación de cada residuo de desplazamiento desde de datos codificados, el medio de determinación del valor de predicción para determinar un valor de predicción de cada desplazamiento desde de un desplazamiento decodificado y el medio de cálculo de desplazamiento para calcular cada desplazamiento desde de un valor de predicción determinado por el medio de determinación de valor de predicción, y un residuo de desplazamiento, decodificado por el medio de decodificación de residuo de desviación. En consecuencia, se puede decodificar, de forma adecuada, un desplazamiento desde de datos codificados que tienen una menor cantidad de datos, en comparación con un caso en el que cada desplazamiento está codificado.

La imagen de entrada puede estar constituida, preferiblemente, por una pluralidad de zonas unitarias. El medio de decodificación de residuo de desplazamiento puede decodificar, preferentemente, cada residuo de desplazamiento en asociación con un tipo de desplazamiento, que se determina para cada zona unitaria, y una clase de desplazamiento, que se determina para cada píxel. El medio de determinación del valor de predicción puede determinar, preferentemente, un valor de predicción de cada desplazamiento desde de un desplazamiento decodificado asociado con el mismo tipo de desplazamiento y la misma clase de desplazamiento que los del desplazamiento para el que se determinará un valor de predicción.

Con la configuración anteriormente descrita, el valor de predicción de cada desplazamiento se determina desde de un desplazamiento decodificado asociado con el mismo tipo de desplazamiento y la misma clase de desplazamiento que los del desplazamiento para el cual se determinará un valor de predicción, por lo que es posible mejorar la precisión de predicción. Por consiguiente, con la configuración descrita anteriormente, es posible decodificar, de forma adecuada, un desplazamiento desde de datos codificados que tienen una menor cantidad de datos.

La imagen de entrada puede estar constituida preferentemente, por una pluralidad de zonas unitarias. El medio de decodificación de residuo de desplazamiento puede decodificar, de forma preferible, cada residuo de desplazamiento en asociación con un tipo de desplazamiento, que se determina para cada zona unitaria, y una clase de desplazamiento, que se determina para cada píxel. El medio de determinación del valor de predicción puede determinar preferiblemente un valor de predicción de cada desplazamiento desde de un desplazamiento decodificado, asociado con el mismo primer grupo de tipo de desplazamiento y la misma clase de desplazamiento que los del desplazamiento para el que se determinará un valor de predicción, en un caso en el que el tipo de desplazamiento asociado con el desplazamiento, pertenece a un primer grupo de tipo de desplazamiento, y el medio de determinación del valor de predicción pueden determinar, preferentemente, un valor de predicción de cada desplazamiento desde de un desplazamiento decodificado asociado con el mismo tipo de desplazamiento y la misma clase de desplazamiento que los del desplazamiento para el que se determinará el valor de predicción en un caso en el que el tipo de desplazamiento, asociado con el desplazamiento pertenece a un segundo grupo de tipo de desplazamiento.

Con la configuración descrita anteriormente, el valor de predicción de cada desplazamiento se determina desde de un desplazamiento decodificado asociado con el mismo tipo de desplazamiento que el desplazamiento para el cual se determinará un valor de predicción en un caso en el que el tipo de desplazamiento, asociado con el desplazamiento, pertenece a un primer grupo de tipo de desplazamiento, y el valor de predicción de cada desplazamiento se determina desde de un desplazamiento decodificado asociado con el mismo tipo de desplazamiento, y la misma clase de desplazamiento que los del desplazamiento para el que se determinará un valor de predicción en un caso en el que el tipo de desplazamiento asociado con el desplazamiento pertenece a un segundo grupo de tipo de desplazamiento, lo que hace posible mejorar la precisión de predicción al mismo tiempo que se reduce la cantidad de procesamiento. Por lo tanto, con la configuración descrita con anterioridad, es posible decodificar, de forma adecuada, un desplazamiento desde de datos codificados que tienen una menor cantidad de datos mientras se reduce la cantidad de procesamiento.

El primer tipo de desplazamiento es un tipo en el que cada píxel, en una zona unitaria asociada con el primer tipo de desplazamiento, se clasifica como una de entre una pluralidad de clases de conformidad con, a modo de ejemplo, el modo de un borde en la proximidad del píxel. El segundo tipo de desplazamiento es un tipo en el que cada píxel, en una zona unitaria asociada con el segundo tipo de desplazamiento, se clasifica como una de entre una pluralidad de clases de conformidad con, a modo de ejemplo, el valor de píxel del píxel.

El medio de cálculo de desplazamiento puede calcular, preferentemente, cada desplazamiento como una función lineal de un valor de predicción determinado por el medio de determinación de valor de predicción, y un residuo de desplazamiento decodificado por el medio de decodificación de residuo de desplazamiento. Un coeficiente que se multiplica por el valor de predicción puede diferir, de forma preferente, dependiendo de si el tipo de desplazamiento asociado con el desplazamiento pertenece al primer grupo de tipo de desplazamiento, o al segundo grupo de tipo de desplazamiento.

Con la configuración anteriormente descrita, el coeficiente que se multiplica por el valor de predicción difiere dependiendo de si el tipo de desplazamiento asociado con el desplazamiento pertenece al primer grupo de tipo de desplazamiento, o al segundo grupo de tipo de desplazamiento. De este modo, es posible calcular el

desplazamiento utilizando un coeficiente más adecuado en función del tipo de desplazamiento, lo que permite mejorar la eficiencia de codificación.

5 El medio de determinación del valor de predicción puede determinar, de forma preferida, un valor de predicción de cada desplazamiento calculando un promedio ponderado de un desplazamiento decodificado y un valor de predicción del desplazamiento decodificado.

10 Con la configuración descrita anteriormente, se determina un valor de predicción de cada desplazamiento calculando un promedio ponderado de un desplazamiento decodificado, y un valor de predicción, del desplazamiento decodificado. Por consiguiente, una pluralidad de desplazamientos decodificados contribuye a un valor de predicción de cada desplazamiento, suprimiendo así las fluctuaciones excesivas en el valor de predicción. Con esta disposición, a modo de ejemplo, incluso si se ha calculado un valor de predicción inadecuado, debido a la influencia del ruido, se puede suprimir la influencia de dicho valor de predicción inadecuado, haciendo posible, de este modo, la mejora de la eficiencia de codificación.

15 El medio de determinación del valor de predicción puede incluir, de forma preferida, un medio de recorte para recortar cada uno de los valores de predicción determinados utilizando un valor límite superior y un valor límite inferior que corresponden a una profundidad de bits del valor de píxel de cada píxel que forma la imagen de entrada.

20 Con la configuración descrita anteriormente, puesto que cada uno de los valores de predicción determinados se recorta utilizando un valor límite superior, y un valor límite inferior, que corresponde a una profundidad de bits del valor de píxel de cada píxel que forma la imagen de entrada, no se generan valores de predicción excesivamente grandes, y valores de predicción excesivamente pequeños, lo que hace posible mejorar la eficiencia de codificación.

25 Un dispositivo de filtrado de imagen, de conformidad con la presente invención, es un dispositivo de filtrado de imagen que opera sobre una imagen de entrada. El dispositivo de filtrado de imagen incluye: un medio de cálculo para calcular un valor de diferencia entre un valor de píxel de un píxel objeto que forma una imagen de entrada, y un valor de píxel de un píxel situado alrededor del píxel objeto; un desplazamiento binario significa realizar un desplazamiento a la derecha a nivel de bits en un valor de píxel referido por el medio de cálculo, o el valor de diferencia calculado por el medio de cálculo en una cantidad igual a un valor de desplazamiento predeterminado; un medio de clasificación para clasificar el píxel objeto como una de entre una pluralidad de clases de desplazamiento de conformidad con una relación de magnitud entre el valor de diferencia sometido a desplazamiento a la derecha a nivel de bits por el medio de desplazamiento binario y 0; y un medio de desplazamiento para añadir un desplazamiento asociado con la clase de desplazamiento del píxel objeto clasificado por el medio de clasificación, al valor de píxel del píxel objeto.

30 En el dispositivo de filtrado de imagen configurado tal como se describió anteriormente, el píxel objeto se clasifica como una de entre una pluralidad de clases de desplazamiento de conformidad con una relación de magnitud entre el valor de diferencia sometido a desplazamiento a la derecha a nivel de bits por el medio de desplazamiento binario y 0, y se añade un desplazamiento asociado con la clase de desplazamiento del píxel objeto clasificado por el medio de clasificación, al valor de píxel del píxel objeto. Por lo tanto, el procesamiento de clasificación es menos vulnerable a la influencia del ruido, lo que hace posible mejorar la eficiencia de codificación.

45 El valor de desplazamiento predeterminado puede tener, de forma preferible, una correlación positiva con una profundidad de bits del valor de píxel del píxel objeto.

50 Con la configuración anteriormente descrita, puesto que el valor de desplazamiento predeterminado puede tener, preferentemente, una correlación positiva con una profundidad de bits del valor de píxel del píxel objeto, se puede mejorar, de forma más eficaz, la eficiencia de codificación.

55 Un dispositivo de filtrado de imagen, de conformidad con la presente invención, es un dispositivo de filtrado de imagen que opera sobre una imagen de entrada. El dispositivo de filtrado de imagen incluye: un medio de cálculo para calcular un valor de diferencia entre un valor de píxel, de un píxel objeto, que forma una imagen de entrada y un valor de píxel de un píxel situado alrededor del píxel objeto; un medio de clasificación para clasificar el píxel objeto como una de entre una pluralidad de clases de desplazamiento de conformidad con una relación de magnitud entre el valor de diferencia, calculado por el medio de cálculo, y cada uno de los primero y segundo umbrales predeterminados; y un medio de desplazamiento para añadir un desplazamiento asociado con la clase de desplazamiento del píxel objeto clasificado por el medio de clasificación, al valor de píxel del píxel objeto.

60 En el dispositivo de filtrado de imagen configurado según se describió con anterioridad, el píxel objeto se clasifica como una de entre una pluralidad de clases de desplazamiento de conformidad con una relación de magnitud entre el valor de diferencia calculado por el medio de cálculo, y cada uno de los primero y segundo umbrales predeterminados, y se añade un desplazamiento asociado con la clase de desplazamiento del píxel objeto clasificado por el medio de clasificación al valor de píxel del píxel objeto. En consecuencia, el procesamiento de clasificación es menos vulnerable a la influencia del ruido, lo que hace posible mejorar la eficiencia de codificación.

65

Los valores absolutos de los primero y segundo umbrales pueden tener, preferentemente, una correlación positiva con una profundidad de bits del valor de píxel del píxel objeto.

5 Con la configuración descrita anteriormente, puesto que los valores absolutos de los primero y segundo umbrales, tienen una correlación positiva con la profundidad de bits del valor de píxel del píxel objeto, se puede mejorar la eficiencia de codificación de forma más eficaz.

10 Un dispositivo de filtrado de imagen, de conformidad con la presente invención, es un dispositivo de filtrado de imagen que funciona sobre una imagen de entrada constituida por una pluralidad de zonas unitarias. El dispositivo de filtrado de imagen incluye: un medio de determinación para determinar, entre los primero y segundo tipos de desplazamiento, un tipo de desplazamiento al que pertenece una zona unitaria objeto que incluye un píxel objeto que forma la imagen de entrada; un medio de clasificación para clasificar el píxel objeto como una de entre una clase de desplazamiento en la que no se añade un desplazamiento, y una pluralidad de clases de desplazamiento en la que se añade un desplazamiento de conformidad con el tipo de desplazamiento al que pertenece la zona unitaria objeto, y un valor de píxel del píxel objeto; y un medio de desplazamiento para añadir un desplazamiento asociado con el tipo de desplazamiento al que pertenece la zona unitaria objeto, y la clase de desplazamiento del píxel objeto clasificado por el medio de clasificación, al valor de píxel del píxel objeto. En un caso en el que el valor de píxel del píxel objeto está dentro de un margen predeterminado, el medio de clasificación clasifica el píxel objeto como una clase de desplazamiento en la que se añade un desplazamiento, sin tener en cuenta si el tipo de desplazamiento al que pertenece la zona unitaria, que incluye el píxel objeto, es el primer tipo de desplazamiento, o el segundo tipo de desplazamiento.

25 En el dispositivo de filtrado de imagen, configurado según se describió con anterioridad, en un caso en el que el valor de píxel del píxel objeto está dentro de un margen predeterminado, el píxel objeto se clasifica como una clase de desplazamiento en la que se añade un desplazamiento, independientemente de si el tipo de desplazamiento al que pertenece la zona unitaria que incluye el píxel objeto, es el primer tipo de desplazamiento, o el segundo tipo de desplazamiento, lo que hace posible eliminar, de forma eficaz, el ruido de bloque. En consecuencia, con la configuración descrita anteriormente, se puede mejorar la eficiencia de codificación.

30 Un dispositivo de filtrado de imagen, de conformidad con la presente invención, es un dispositivo de filtrado de imagen para la adición de un desplazamiento a un valor de píxel de cada píxel que forma una imagen de entrada que está constituida por una pluralidad de zonas unitarias. El dispositivo de filtrado de imagen incluye: un medio de determinación para determinar un tipo de desplazamiento al que pertenece una zona unitaria objeto entre una pluralidad de tipos de desplazamiento; un medio de codificación de desplazamiento para determinar un desplazamiento que tiene una anchura de bit que difiere en función del tipo de desplazamiento, y para codificar el desplazamiento; y un medio de filtrado para añadir el desplazamiento determinado al valor de píxel de cada píxel que forma la imagen de entrada.

40 En el dispositivo de filtrado de imagen, configurado según se describió con anterioridad, entre una pluralidad de tipos de desplazamiento, se determina el tipo de desplazamiento al que pertenece una zona unitaria objeto, se determina un desplazamiento que tiene una anchura de bit que difiere dependiendo del tipo de desplazamiento determinado, y el desplazamiento determinado se añade al valor de píxel de cada píxel que forma la imagen de entrada. Además, se codifica el desplazamiento determinado.

45 En consecuencia, con la configuración descrita anteriormente, es posible realizar un procesamiento de filtrado de desplazamiento adecuado al mismo tiempo que se reduce el tamaño de memoria de una memoria para almacenar desplazamientos. Con la configuración descrita con anterioridad, puesto que se reduce la cantidad de datos necesarios para codificar los datos, se mejora la eficiencia de la codificación.

50 Un dispositivo de codificación de desplazamiento, de conformidad con la presente invención, es un dispositivo de codificación de desplazamiento para codificar cada desplazamiento al que se hace referencia por un filtro de imagen para añadir un desplazamiento a un valor de píxel de cada píxel que forma una imagen de entrada. El dispositivo de codificación de desplazamiento incluye: un medio de determinación de valor de predicción para determinar un valor de predicción de cada desplazamiento desde de un desplazamiento codificado; un medio de cálculo residual, para calcular un residuo de desplazamiento, de cada desplazamiento, y un valor de predicción determinado por el medio de determinación del valor de predicción; y un medio de codificación de residuo de desplazamiento para codificar un residuo de desplazamiento que se calcula por el medio de cálculo de residuo de desplazamiento.

60 En el dispositivo de codificación de desplazamiento, configurado tal como se describió anteriormente, se proporciona el medio de determinación del valor de predicción para determinar un valor de predicción de cada desplazamiento, desde de un desplazamiento codificado, el medio de cálculo de residuo de desplazamiento para calcular un residuo de desplazamiento desde de cada desplazamiento, y un valor de predicción determinado por el medio de determinación del valor de predicción, y el medio de codificación de residuo de desplazamiento para codificar un residuo de desplazamiento que se calcula por el medio de cálculo de residuo de desplazamiento. Por lo tanto, es posible reducir la cantidad de datos necesarios para poder codificar los datos.

65

Una estructura de datos de datos codificados, de conformidad con la presente invención, es una estructura de datos de datos codificados a la que se hace referencia mediante un filtro de imagen para añadir un desplazamiento a un valor de píxel de cada píxel que forma una imagen de entrada que está constituida por una pluralidad de zonas unitarias. La estructura de datos incluye: información de especificación de tipo de desplazamiento que especifica un tipo de desplazamiento al que pertenece cada zona unitaria; y un desplazamiento que tiene una anchura de bit que difiere dependiendo del tipo de desplazamiento. El filtro de imagen se refiere a la información de especificación de tipo de desplazamiento, incluida en los datos codificados, y determina un tipo de desplazamiento al que pertenece la zona unitaria objeto y, además, decodifica un desplazamiento que tiene una anchura de bit que difiere según el tipo de desplazamiento determinado.

Los datos codificados, configurados tal como se describió anteriormente, incluyen un desplazamiento que tiene una anchura de bit que difiere en función del tipo de desplazamiento, con lo que se reduce la cantidad de datos requeridos para codificar los datos. El filtro de imagen que decodifica los datos codificados se refiere a la información de especificación de tipo de desplazamiento, y determina el tipo de desplazamiento al que pertenece la zona unitaria objeto y, además, decodifica un desplazamiento que tiene una anchura de bit que difiere según el tipo de desplazamiento determinado. Por lo tanto, es posible realizar un procesamiento de filtrado de desplazamiento adecuado mientras se reduce el tamaño de memoria de una memoria para almacenar desplazamientos.

La información de especificación de tipo de desplazamiento se puede determinar para cada una de las imágenes de entrada, o para cada una de las zonas unitarias. Como alternativa, la información de especificación de tipo de desplazamiento se puede determinar para cada conjunto predeterminado de imágenes de entrada, o para cada conjunto predeterminado de las zonas unitarias.

Tercera forma de realización

La información de desplazamiento OI, de una tercera forma de realización, se describirá a continuación, en primer lugar, haciendo referencia a la Figura 23. La parte (a) de la Figura 23 ilustra la sintaxis (indicada por "sao_offset_param()") en la parte (a) de la Figura 23) de la información de desplazamiento OI.

Tal como se ilustra en la parte (a) de la Figura 23, la información de desplazamiento OI incluye un parámetro "sao_type_idx [sao_curr_depth] [ys] [xs]". Si el parámetro "sao_type_idx [sao_curr_depth] [ys] [xs]" no es 0, se incluye un parámetro sao_offset [sao_curr_depth] [ys] [xs] [i] en la información de desplazamiento OI.

(sao_curr_depth, ys, xs)

Un argumento "sao_curr_depth", que es un argumento de "sao_type_idx" y "sao_offset", es un parámetro que indica la profundidad de división de una QAOU, e "ys" y "xs" son parámetros que indican, respectivamente, la posición en la dirección y, y la posición en la dirección x, de una QAOU (o una QAOMU, que se examinará más adelante).

Los modos de división de una QAOU, de conformidad con los valores de "sao_curr_depth", son los mismos que los descritos con referencia a la Figura 4.

La parte (b) de la Figura 23 ilustra la sintaxis (indicada por "sao_split_param()") en la parte (b) de la Figura 23) de información de QAOU. Tal como se indica por la sintaxis ilustrada en la parte (b) de la Figura 23, si la profundidad de división "sao_curr_depth" es más pequeña que un valor máximo predeterminado establecido por "saoMaxDepth", se determina mediante el parámetro "sao_split_flag" si una QAOU se dividirá todavía más. Si se divide, todavía más una QAOU, será demandada, de forma recursiva, una profundidad jerárquica posterior "sao_split_param()". Si la profundidad de división ha alcanzado el valor máximo ("sao_curr_depth" no es menor que "saoMaxDepth"), se establece "0" en "sao_split_flag [sao_curr_depth] [ys] [xs]" y no se dividirá todavía más la QAOU.

La Figura 44 ilustra otro ejemplo de sintaxis de información de desplazamiento, y de sintaxis de información de QAOU.

La parte (a) de la Figura 44 ilustra la sintaxis de la información de desplazamiento. La configuración de la sintaxis es similar a la que se ilustra en la parte (a) de la Figura 23, pero se añade el "componente" que indica el valor de un componente de color como un argumento de "sao_offset_param()", y como un índice de una matriz de "sao_split_flag", "sao_type_idx" y "sao_offset". Mediante la adición de un "componente", se puede dividir una QAOU de manera diferente de conformidad con un componente de color, tal como un componente de luminancia o un componente de crominancia y, se pueden aplicar diferentes desplazamientos para dividir QAOUs.

La parte (b) de la Figura 44 ilustra la sintaxis de información de QAOU. Como en la parte (a) de la Figura 44, se añade un componente de color "component" como un argumento a la sintaxis que se ilustra en la parte (b) de la Figura 23.

La parte (c) de la Figura 44 ilustra la sintaxis completa de un filtro de desplazamiento adaptativo que solicita la sintaxis de la parte (a) y la sintaxis de la parte (b) de la Figura 44. El parámetro "sao_flag" es un indicador que indica

si se aplica, o no, un filtro de desplazamiento adaptativo. Solamente cuando este indicador sea verdadero, se utilizarán los siguientes parámetros relacionados con el filtro de desplazamiento adaptativo. Si este indicador es verdadero, la sintaxis "sao_split_param()" y la sintaxis "sao_offset_param ()" se demandan especificando el nivel jerárquico más alto para cada componente de color. Puesto que se especifica el nivel jerárquico más alto, los argumentos proporcionados para cada sintaxis son sao_curr_depth=0, ys=0 y xs=0. Los componentes de color se distinguen entre sí estableciendo los valores de "component" como sigue: el valor es 0 en el caso de la luminancia (Y), el valor es 1 en el caso de la crominancia (Cb), y el valor es 2 en el caso de la crominancia (Cr). Se pueden utilizar otros valores como los valores del componente siempre que los componentes de color se puedan distinguir entre sí.

Con respecto a la crominancia (Cb) y la crominancia (Cr), se utilizan, respectivamente, indicadores "sao_flag_cb" y "sao_flag_cr" para determinar si se aplica un filtro de desplazamiento adaptativo y, si no se aplica, no se memoriza la información de QAOU y la información de desplazamiento en relación con el componente de color asociado.

Puesto que el argumento "component" se añade en la sintaxis ilustrada en la Figura 44, en la descripción que utiliza la Figura 23, el procesamiento se realiza mediante la sustitución del argumento [sao_curr_depth] [ys] [xs] por [sao_curr_depth] [ys] [xs] [component]. En la siguiente descripción, el procesamiento se realiza de una forma similar.

(Dispositivo de decodificación de vídeo 1')

Un dispositivo de decodificación de vídeo 1', de la tercera forma de realización, se describirá a continuación con referencia a las Figuras 22 y 24 a 29. Los elementos que tienen las mismas funciones que los examinados en las formas de realización anteriormente descritas, se designan con números de referencia similares y, por lo tanto, se omite aquí una explicación de los mismos.

Como en el dispositivo de decodificación de vídeo 1, el dispositivo de decodificación de vídeo 1' incluye, parcialmente, un método definido en H. 264/MPEG-4. AVC, un método utilizado en software KTA, que es un códec de desarrollo conjunto en VCEG (Grupo de Expertos de Codificación de Vídeo), un método utilizado en el software TMuC (Modelo de Prueba bajo Consideración), que es un códec sucesor al códec utilizado en el software KTA, y una tecnología utilizada en el software HM (HEVC TestModel). El dispositivo de decodificación de vídeo 1', de esta forma de realización es diferente del dispositivo de decodificación de vídeo 1 de la primera forma de realización, en el hecho de que incluye un filtro de desplazamiento adaptativo 60' en lugar del filtro de desplazamiento adaptativo 60 del dispositivo de decodificación de vídeo 1. Los otros elementos de la configuración del dispositivo de decodificación de vídeo 1' son similares a los del dispositivo de decodificación de vídeo 1.

(Filtro de desplazamiento adaptativo 60')

Los detalles del filtro de desplazamiento adaptativo 60' se explicarán, a continuación, haciendo referencia a la Figura 22. La Figura 22 es un diagrama de bloques que ilustra la configuración del filtro de desplazamiento adaptativo 60'. Tal como se muestra en la Figura 22, el filtro de desplazamiento adaptativo 60' incluye un decodificador de información de filtro de desplazamiento adaptativo 61', y un procesador de filtro de desplazamiento adaptativo 62'.

El decodificador de información de filtro de desplazamiento adaptativo 61' incluye, tal como se ilustra en la Figura 22, una sección de decodificación de información de desplazamiento 611 y una sección de decodificación de estructura de QAOU 612.

La sección de decodificación de información de desplazamiento 611 se refiere a la información de QAOU incluida en los datos codificados nº 1, y decodifica la información de desplazamiento OI incluida en la información de QAOU. Los valores "sao_type_idx [sao_curr_depth] [ys] [xs] [component]" y "sao_offset [sao_curr_depth] [ys] [xs] [i]", obtenidos mediante la decodificación de la información de desplazamiento OI, se proporcionan a la sección de memorización de información de desplazamiento 621 en asociación con los argumentos (sao_curr_depth, ys, xs, i).

Más concretamente, la sección de decodificación de información de desplazamiento 611 decodifica el código de los datos codificados nº 1, y transforma el código de decodificación al valor de "sao_type_idx" y lo proporciona a la sección de memorización de información de desplazamiento 621, en asociación con los argumentos. La sección de decodificación de información de desplazamiento 611 cambia el método de decodificación de código y lo transforma desde del código en el valor de "sao_type_idx" de conformidad con la profundidad jerárquica de una QAOU objeto, para su procesamiento. Las condiciones, tales como la profundidad jerárquica de una QAOU, se denominan como "condiciones de parámetro". La decodificación de información de desplazamiento, de conformidad con las condiciones de parámetros generales, se tratará más adelante haciendo referencia a las Figuras 41 y 42.

En el método de decodificación de código se pueden utilizar diferentes valores máximos dependiendo de si la profundidad jerárquica de una QAOU es menor que un umbral, o es igual o mayor que el umbral. De forma alternativa, el valor máximo se puede utilizar solamente cuando la profundidad jerárquica de una QAOU objeto es igual o mayor que el umbral. En el método de decodificación de código, se pueden utilizar diferentes técnicas de

binarización dependiendo de si la profundidad jerárquica de una QAOU objeto es menor que un umbral, o es igual o mayor, que el umbral. Como alternativa, se pueden utilizar diferentes contextos dependiendo de si la profundidad jerárquica de una QAOU objeto es menor que un umbral, o es igual o mayor, que el umbral.

A modo de ejemplo, si la profundidad jerárquica de una QAOU objeto es más pequeña que un umbral, se puede decodificar el código utilizando una codificación de longitud variable (ue (v)) y, si la profundidad jerárquica de una QAOU objeto es igual, o mayor, que el umbral, el código puede ser decodificado utilizando una codificación truncada (te (v)) que corresponde al número de tipos de desplazamiento. Si el número de tipos de desplazamiento es una potencia de dos, el código se puede decodificar utilizando una codificación de longitud fija. Si el número de tipos de desplazamiento es cuatro, puede representarse mediante dos bits y, por lo tanto, se puede decodificar el código utilizando una codificación de longitud fija con dos bits.

La sección de decodificación de información de desplazamiento 611 transforma el código decodificado en el valor de "sao_type_idx" utilizando tablas de transformación 801 y 802 que se ilustran en la Figura 25. La tabla de transformación 801, ilustrada en la parte (a) de la Figura 25, indica dos patrones de transformación. Entre los dos patrones de transformación, se utiliza un patrón dependiendo de la profundidad jerárquica de una QAOU objeto. Es decir, cuando la profundidad jerárquica de una QAOU objeto es menor que un umbral, la sección de decodificación de información de desplazamiento 611 utiliza una tabla de transformación 801A y, cuando la profundidad jerárquica de una QAOU objeto es igual, o mayor, que el umbral, la sección de decodificación de información de desplazamiento 611 utiliza una tabla de transformación 801B.

En la tabla de transformación 801A, que se muestra en la parte (a) de la Figura 25, se asocia un tipo de desplazamiento "EO_0" ("sao_type_idx"=1) con un código "1", "EO_1" ("sao_type_idx"=2) está asociado con un código "2", y "EO_2" ("sao_type_idx"=3) está asociado con un código "3". Los tipos de desplazamiento están asociados con códigos 4 a 6 de una manera similar.

En la tabla de transformación 801B, ilustrada en la parte (b) de la Figura 25, un tipo de desplazamiento "EO_0" ("sao_type_idx"=1) está asociado con un código "1", "BO_0" ("sao_type_idx"=5) está asociado con un código "2", y "BO_1" ("sao_type_idx"=6) está asociado con un código "3". En la tabla de transformación 801B, no se utilizan los códigos 4 a 6.

De este modo, en la tabla de transformación 801A, se incluyen todos los tipos de desplazamiento utilizados para los desplazamientos adaptativos (SAOs). Por el contrario, en la tabla de transformación 801B, solamente se incluyen algunos de los tipos de desplazamiento utilizados para los desplazamientos adaptativos (SAOs) y, por lo tanto, cuando la profundidad jerárquica de una QAOU objeto es igual o mayor que el umbral, no se pueden utilizar algunos tipos de desplazamiento.

El motivo de lo que antecede es el siguiente. Puesto que la zona de una QAOU se hace más pequeña en un nivel jerárquico más profundo, las características de valores de píxel dentro de dicha QAOU son casi uniformes, y se pueden determinar desplazamientos adecuados sin utilizar numerosos tipos de desplazamiento. En consecuencia, se puede disminuir el número de tipos de desplazamiento, con lo que se reduce un espacio de memoria requerido. Además, la longitud de datos codificados, que indican los tipos de desplazamiento puede disminuirse, lo que permite mejorar la eficiencia de la codificación.

Si el orden de las asociaciones entre los códigos y los tipos de desplazamiento, en la tabla de transformación 801B es el mismo que en la tabla de transformación 801A, solamente se puede utilizar la tabla de transformación 801A.

Una tabla de transformación 802, ilustrada en la parte (b) de la Figura 25, indica varios ejemplos de tablas de transformación que son similares a la tabla de transformación 801B, y que se pueden utilizar en lugar de la tabla de transformación 801B. En todas las tablas de transformación, el modo de los tipos está restringido, como en la tabla de transformación 801B. El espacio en blanco indica que no se utiliza el código asociado.

Una tabla de transformación 802A es un ejemplo en el que solamente se utilizan desplazamientos de borde, y los desplazamientos de banda no están incluidos. El número de desplazamientos de un tipo de desplazamiento de borde suele ser 4 y, el número de desplazamientos de un tipo de desplazamiento de banda es, generalmente, 16. De este modo, el número de desplazamientos de un tipo de desplazamiento de borde es menor que el de un desplazamiento de banda. En consecuencia, al restringir el uso de desplazamientos de banda, se puede reducir el espacio de memoria utilizado para memorizar desplazamientos. En particular, cuando la profundidad jerárquica se hace más profunda, el número de desplazamientos aumenta y, por lo tanto, el espacio de memoria también aumenta. Cuando la profundidad jerárquica se hace más profunda, es menos probable que se seleccione un desplazamiento de banda. En consecuencia, la profundidad jerárquica se utiliza como una condición de parámetro y, cuando la profundidad jerárquica es profunda, se utiliza la tabla de transformación 802A que incluye solamente desplazamientos de borde, y cuando la profundidad jerárquica no es profunda, se utiliza una tabla de transformación (a modo de ejemplo, la tabla de transformación 801A) que incluye los desplazamientos de borde y los desplazamientos de banda. Por lo tanto, es posible reducir el espacio de memoria sin disminuir la eficiencia de codificación. Además, los cálculos de costos para opciones innecesarias se pueden omitir en el dispositivo de

codificación, con lo que se reduce la cantidad de procesamiento.

Una tabla de transformación 802B es un ejemplo en el que solamente se utilizan desplazamientos de banda y no se incluyen los desplazamientos de borde. Un desplazamiento de banda requiere una cantidad menor de cálculo que un desplazamiento de borde. Además, un desplazamiento de banda no utiliza un píxel situado alrededor de un píxel objeto y, por lo tanto, no es necesaria una memoria de línea para la memorización de un píxel de referencia. Por consiguiente, la tabla de transformación 802B se utiliza de conformidad con la condición del parámetro, logrando así las ventajas descritas anteriormente. Una tabla de transformación 802C es un ejemplo en el que solamente se utiliza un desplazamiento de banda "BO_0" ("sao_type_idx"=5).

Una tabla de transformación 802D es un ejemplo en el que se utiliza un desplazamiento de borde y dos desplazamientos de banda, como en la tabla de transformación 801B. Más concretamente, la tabla de transformación 802D es un ejemplo en el que los tipos de desplazamiento "EO_0" ("sao_type_idx"=1), "BO_0" ("sao_type_idx"=5), y "BO_1" ("sao_type_idx"=6) se utilizan y se asignan, preferentemente, códigos más cortos (números de código más pequeños) a los tipos de desplazamiento de banda. Cuando la profundidad jerárquica no es profunda, es más probable que se seleccione un desplazamiento de banda que cuando la profundidad jerárquica es profunda. Por consiguiente, la profundidad jerárquica se utiliza como una condición de parámetro, y cuando la profundidad jerárquica no es profunda, se usa la tabla de transformación 802D y, se asigna un código más corto al tipo que se utiliza con más frecuencia, lo que hace posible mejorar la eficiencia de codificación.

Se puede usar una tabla diferente a las indicadas por los ejemplos que se ilustran en la Figura 25 y, la asociación entre los códigos y los tipos de desplazamiento se puede cambiar de conformidad con la condición del parámetro. Un tipo de desplazamiento que es diferente de cualquiera de los desplazamientos de borde y los desplazamientos de banda, que se ilustran en la tabla de transformación 801A, se puede utilizar de forma individual o junto con otro tipo de desplazamiento de conformidad con una condición, tal como la profundidad jerárquica. Ejemplos específicos de dicho tipo de desplazamiento, son un tipo de desplazamiento que tiene características tanto de EO como de BO, un desplazamiento de borde que detecta un borde en una posición de muestra horizontal diferente de la de un desplazamiento de borde conocido "EO_0", o un desplazamiento de banda al que se asigna una banda se asigna de una manera diferente a las de los desplazamientos de banda conocidos "BO_0" y "BO_1". Estos ejemplos específicos serán discutidos, más adelante, en otra forma de realización.

La sección de decodificación de estructura de QAOU 612 decodifica "sao_split_flag [sao_curr_depth] [ys] [xs]" que se incluye en la información de QAOU con el fin de determinar la estructura de división de QAOU y a continuación, proporciona la información de estructura de QAOU que indica la estructura de división de QAOU determinada, a la sección de memorización de información de desplazamiento 621.

Se puede incluir una sección de establecimiento de atributos de desplazamiento 613 que se describe a continuación. La sección de establecimiento de atributos de desplazamiento 613 determina el valor de desplazamiento de un desplazamiento. Un desplazamiento en los datos codificados se codifica con la profundidad de bits de desplazamiento (también referida como "SAO_DEPTH"), que tiene una precisión menor que la profundidad de bits de píxel (también referida como "PIC_DEPTH"). Es decir, se realiza una cuantización de un desplazamiento en los datos codificados. El valor de desplazamiento es una cantidad de desplazamiento binario que es necesaria para realizar una cuantización inversa. La sección de establecimiento de atributos de desplazamiento 613 determina, además, la profundidad de bits de desplazamiento y el margen de valores de desplazamiento. En este caso, la profundidad de bits de desplazamiento se determina desde de la profundidad de bits de píxel (también denominada "PIC_DEPTH") (no ilustrada) que se introduce en la sección de establecimiento de atributos de desplazamiento 613. La profundidad de bits de píxel representa el margen de valores de píxel de píxeles que se forman una entrada de imagen en el filtro de desplazamiento adaptativo 60' como la anchura de bit. Cuando la profundidad de bits de píxel es de N bits, los valores de píxel toman un margen de 0 a 2^{N-1} .

La profundidad de bits SAO, y el valor de desplazamiento se determinan mediante el uso de las siguientes ecuaciones. Sin embargo, se pueden utilizar otros valores de conformidad con las condiciones de los parámetros, que se examinarán más adelante.

$$SAO_DEPTH = \min(PIC_DEPTH, 10)$$

$$\text{shift value} = PIC_DEPTH - \min(PIC_DEPTH, 10)$$

El margen de valores de desplazamiento cuantizado se determina mediante la siguiente ecuación.

El procesador de filtro de desplazamiento adaptativo 62' incluye, tal como se ilustra en la Figura 22, la sección de memorización de información de desplazamiento 621, un controlador de QAOU 622, una sección de determinación de tipo de desplazamiento 623, una sección de clasificación 624, una sección de determinación de desplazamiento 625, y un sumador de desplazamiento 626.

La sección de memorización de información de desplazamiento 621 gestiona y memoriza el tipo de desplazamiento especificado para cada QAOU, y los valores específicos de desplazamientos con respecto a clases individuales que se pueden seleccionar para el tipo de desplazamiento especificado, sobre la base de la información de estructura de QAOU, "sao_type_idx [sao_curr_depth] [ys] [xs]", y "sao_offset [sao_curr_depth] [ys] [xs] [i]". La sección de memorización de información de desplazamiento 621 tiene una memoria de mapa y una memoria de lista.

La memoria de mapa y la memoria de lista se explicarán a continuación haciendo referencia a la Figura 24. La Figura 24 ilustra ejemplos de información memorizada en la memoria de mapa y la memoria de lista. La parte (a) de la Figura 24 ilustra ejemplos de índices de QAOU que se memorizan en la memoria de mapa, y la parte (b) de la Figura 24 ilustra ejemplos de información memorizada en la memoria de lista.

En una memoria de mapa 601, se memoriza un índice de QAOU asignado a cada unidad mínima de desplazamiento (también conocida como "QAOMU: Unidad Mínima de Desplazamiento Adaptativo de Cuadratura"), que se determina por la profundidad de división. El índice de QAOU se examina más adelante. La parte (a) de la Figura 24 muestra QAOMUs que tienen una profundidad de división de tres y forman la unidad de procesamiento (a modo de ejemplo, una LCU), y los índices de QAOU asignados a QAOMUs. En la parte (a) de la Figura 24, los índices 0 a 9 se asignan a QAOMUs de una forma sencilla, sin tener en cuenta la profundidad de división de QAOU. En el ejemplo que se muestra en la parte (a) de la Figura 24, una QAOU especificada por el índice de QAOU=1 se indica por QAOU1. En la parte (a) de la Figura 24, las líneas finas indican límites de QAOMU, mientras que las líneas gruesas indican límites de QAOU.

Tal como se muestra en la parte (a) de la Figura 24, QAOU0 está constituido por cuatro QAOMUs, y 0 se asigna a estas cuatro QAOMUs como el índice de QAOU. QAOU3 está constituido por una QAOMU y, 3 se asigna a esta QAOMU como el índice de QAOU. De esta manera, en la memoria de mapa, se memorizan índices de QAOU asignados a QAOMUs individuales.

En una memoria de lista 602, el tipo de desplazamiento asociado con cada índice de QAOU, y valores específicos de desplazamientos con respecto a las clases que se pueden seleccionar para este tipo de desplazamiento, se memorizan en asociación con el índice de QAOU.

Lo que antecede se explicará más concretamente, a continuación, haciendo referencia a la parte (b) de la Figura 24. La parte (b) de la Figura 24 ilustra un tipo de desplazamiento asociado con cada uno de los índices de QAOU de 0 a 9 y desplazamientos con respecto a clases individuales que se pueden seleccionar para cada tipo de desplazamiento. En la parte (b) de la Figura 24, "xxx" indica valores numéricos específicos que pueden ser diferentes entre sí.

En la parte (b) de la Figura 24, "BO_1" indica un tipo de desplazamiento especificado por "sao_type_idx=5". "EO_1" indica un tipo de desplazamiento especificado por "sao_type_idx=1". De este modo, desplazamientos de borde, que son de un tipo de desplazamiento especificado por "sao_type_idx=1, 2, 3, 4", se indican, además, mediante EO_1, 2, 3, 4, respectivamente, y desplazamientos de banda, que son de un tipo de desplazamiento especificado por "sao_type_idx=5, 6", también se indican por BO_1, 2, respectivamente.

Tal como se ilustra en la parte (b) de la Figura 24, cuando el tipo de desplazamiento es un desplazamiento de banda, un total de dieciséis desplazamientos, es decir, desde el desplazamiento 1 hasta el desplazamiento 16, se memoriza en la memoria de lista para este tipo de desplazamiento. El desplazamiento 1, al desplazamiento 16, son valores especificados por "sao_offset [sao_curr_depth] [ys] [xs] [1]" hasta "sao_offset [sao_curr_depth] [ys] [xs] [16]", respectivamente, cuando el valor de "sao_type_idx [sao_curr_depth] [ys] [xs]" es "5" o "6".

Por otro lado, cuando el tipo de desplazamiento es un desplazamiento de borde, un total de cuatro desplazamientos, es decir, del desplazamiento 1 al desplazamiento 4, se memorizan en la memoria de lista para este tipo de desplazamiento. El desplazamiento 1 al desplazamiento 4 son valores especificados por "sao_offset [sao_curr_depth] [ys] [xs] [1]" hasta "sao_offset [sao_curr_depth] [ys] [xs] [4]", respectivamente, cuando el valor de "sao_type_idx [sao_curr_depth] [ys] [xs]" es uno de entre "1, 2, 3 y 4". En el caso de un desplazamiento de borde, no se memoriza ningún valor en el desplazamiento 5 al desplazamiento 16.

Se añade un número de QAOMU a cada QAOMU y se pueden distinguir las QAOMUs, entre sí, por los números de QAOMU. En lo sucesivo, una QAOMU que tenga un número de N_Q de QAOMU se denominará, además, como "QAOMUN_Q".

En esta forma de realización, la precisión de desplazamiento puede variar de conformidad con la profundidad jerárquica de una QAOU objeto. El valor de desplazamiento utilizado para la cuantización inversa también difiere en función de la precisión de desplazamiento. En este caso, además, la precisión de desplazamiento y el valor de desplazamiento se determinan utilizando la profundidad de bits de píxel PIC_DEPTH.

A modo de ejemplo de la precisión de desplazamiento, se puede variar la precisión de desplazamiento, por ejemplo, como sigue. Cuando la profundidad jerárquica de una QAOU es menor que un umbral, la precisión de

desplazamiento y el valor de desplazamiento se establecen:

$$SAO_DEPTH = \min(PIC_DEPTH, 10)$$

- 5 valor de desplazamiento = $PIC_DEPTH - \min(PIC_DEPTH, 10)$; y cuando la profundidad jerárquica de una QAOU es igual, o mayor, que el umbral, la precisión de desplazamiento y el valor de desplazamiento se establecen:

$$SAO_DEPTH = \min(PIC_DEPTH, 8)$$

- 10 valor de desplazamiento = $PIC_DEPTH - \min(PIC_DEPTH, 8)$.

Tal como se indicó con anterioridad, la precisión de desplazamiento (profundidad de bits de desplazamiento, SAO_DEPTH), y la profundidad de bits de píxel (PIC_DEPTH), que expresa el margen de valores de píxel de píxels que forman una imagen de entrada como la anchura de bits, están estrechamente relacionados entre sí en términos de errores de cuantización. La profundidad de bits de una salida de imagen, desde del filtro de desplazamiento adaptativo 60, se representa mediante la profundidad de bits de píxel PIC_DEPTH, y SAO_DEPTH representa la profundidad de bits de un desplazamiento que ha de añadirse a un píxel. En consecuencia, incluso si se utiliza un desplazamiento que tiene una precisión mayor que la de la profundidad de bits de píxel, se descarta en el proceso de salida. Por lo tanto, no tiene sentido establecer SAO_DEPTH que exceda PIC_DEPTH. Por otro lado, si SAO_DEPTH es menor que PIC_DEPTH, una imagen de entrada se corrige simplemente con una precisión menor que la precisión (PIC_DEPTH) con la que la imagen de entrada podría corregirse utilizando un filtro, disminuyendo así el efecto de filtrado.

En consecuencia, en esta forma de realización, al restringir un desplazamiento cuantizado a un margen de valores de desplazamiento que se puede representar por una determinada anchura de bits, es posible reducir la anchura de bits para la memorización de un desplazamiento cuantizado en la sección de memorización de información de desplazamiento 621. Con esta disposición, se puede obtener el efecto de reducir el tamaño de memoria, en comparación con la restricción anteriormente descrita que no se impone. Sin embargo, si el margen de valores de desplazamiento está excesivamente restringido, disminuye el efecto de corregir la distorsión de una imagen decodificada, mediante el uso de un desplazamiento. Por consiguiente, no es posible eliminar la distorsión de una imagen decodificada ni siquiera mediante el procesamiento de adición de desplazamiento, lo que puede disminuir la eficiencia de codificación.

En consecuencia, al establecer el margen de valores de desplazamiento en un margen óptimo en un grado tal que no disminuya la eficacia de codificación, se puede mantener el efecto de filtrado al mismo tiempo que se reduce el espacio de memoria que ha de utilizarse.

La longitud binaria máxima que representa el margen de valores de desplazamiento se indica por CLIP_BIT, y mediante el cálculo de $CLIP_BIT = SAO_DEPTH - K$, se determina el margen de valores de desplazamiento para ser 2^{CLIP_BIT-1} a $2^{CLIP_BIT}-1$. A continuación, a través de los experimentos de los inventores, se descubrió que, en el caso de $K=4$, no disminuye la eficacia de codificación incluso si el margen de desplazamiento estaba restringido por el margen de valores de desplazamiento.

Cuando la profundidad de bits del píxel es ocho, la profundidad de bits de desplazamiento SAO_DEPTH también es ocho y, por lo tanto, $CLIP_BIT = 8 - K = 4$. Si se puede memorizar un desplazamiento utilizando cuatro bits, a modo de ejemplo, en un software que gestiona un byte constituido por ocho bits como la unidad, un desplazamiento se puede empaquetar y memorizarse en un byte, reduciendo así, fácilmente, el tamaño de la memoria.

Si la precisión de desplazamiento que se establece varía dependiendo de la profundidad jerárquica de una QAOU objeto, la sección de memorización de información de desplazamiento 621 asegura una zona de memorización de desplazamiento cambiando el tamaño de la unidad de la zona de memorización de desplazamiento de conformidad con la profundidad jerárquica de la QAOU objeto.

Más concretamente, ahora se supone que cuando la profundidad jerárquica de una QAOU objeto es más pequeña que un umbral, la precisión de desplazamiento se establece para ser n_A bits (a modo de ejemplo, $n_A=8$ o 6), y que cuando la profundidad jerárquica de la QAOU objeto es igual, o mayor que el umbral, la precisión de desplazamiento se establece en n_B bits ($n_A > n_B$, a modo de ejemplo, $n_B = n_A - 2$). En este caso, la sección de memorización de información de desplazamiento 621 asegura una zona para la memorización de desplazamientos cuando la profundidad jerárquica de la QAOU objeto es menor que el umbral utilizando los n_A bits como la unidad y, asegura una zona para memorizar desplazamientos cuando la profundidad jerárquica de la QAOU objeto es igual, o mayor, que el umbral mediante la utilización de los n_B bits como la unidad.

Si la profundidad jerárquica de la QAOU objeto es igual a, o mayor que, el umbral, cuando se realiza la lectura y escritura de un desplazamiento, desde y hacia, la sección de memorización de información de desplazamiento 621, es preferible que un desplazamiento se exprese redondeándolo en sentido descendente por $(n_A - n_B)$ bits, y que un

desplazamiento sea objeto de lectura redondeándolo en sentido ascendente por $(n_A - n_B)$ bits. Con esta configuración de entrada/salida, no es necesario que otros módulos realicen el procesamiento considerando la diferencia en la precisión de desplazamiento.

Si el número requerido de clases varía en función de la profundidad jerárquica de una QAOU objeto, la sección de memorización de información de desplazamiento 621 cambia una zona de una memoria de lista para ser asegurada de conformidad con el número requerido de clases. A modo de ejemplo, se considera un desplazamiento de banda que se divide en dieciséis clases si la profundidad jerárquica de una QAOU objeto es menor que un umbral, y que se divide en ocho clases si la profundidad jerárquica de la QAOU objeto es igual, o mayor, que el umbral, (dicho desplazamiento de banda se discutirá más adelante). En este caso, el espacio de memoria necesario para asegurar clases cuando la profundidad jerárquica de la QAOU objeto es igual, o mayor, que el umbral, es la mitad del espacio de memoria cuando la profundidad jerárquica de la QAOU objeto es menor que el umbral. En consecuencia, cuando la profundidad jerárquica de la QAOU objeto es igual o mayor que el umbral, la sección de memorización de información de desplazamiento 621 cambia el tamaño de la memoria de lista para garantizar que sea la mitad del tamaño de la memoria que se establecerá cuando la profundidad jerárquica de la QAOU objeto sea más pequeño que el umbral.

El controlador de QAOU 622 controla los componentes individuales incluidos en el procesador de filtro de desplazamiento adaptativo 62'. El controlador de QAOU 622 hace referencia, además, a información de la estructura de QAOU, y a continuación, divide la imagen decodificada desbloqueada P_DB en una, o una pluralidad, de QAOUs y analiza las QAOUs individuales en un orden predeterminado. El controlador de QAOU 622 proporciona, además, un número de QAOMU, que representa una QAOMU objeto, a la sección de determinación de tipo de desplazamiento 623.

La sección de determinación de tipo de desplazamiento 623 se refiere a la memoria de mapa y la memoria de lista de la sección de memorización de información de desplazamiento 621, y determina el tipo de desplazamiento especificado por el número de QAOMU que se proporciona por el controlador de QAOU 622. La sección de determinación de tipo de desplazamiento 623 proporciona, además, el tipo de desplazamiento determinado a la sección de clasificación 624.

La sección de clasificación 624 clasifica cada píxel incluido en la QAOU objeto como una de las clases que se pueden seleccionar para el tipo de desplazamiento que se proporciona desde de la sección de determinación de tipo de desplazamiento 623. La sección de clasificación 624 proporciona, además, el tipo de desplazamiento y el índice de clase que indica la clase de cada píxel a la sección de determinación de desplazamiento 625. El procesamiento de clasificación específico realizado por la sección de clasificación 624 se examinará más adelante y, por lo tanto, se omite aquí una descripción de los mismos.

La sección de determinación de desplazamiento 625 se refiere a la memoria de lista de la sección de memorización de información de desplazamiento 621 y determina, para cada píxel incluido en la QAOU objeto, un desplazamiento especificado por el tipo de desplazamiento y el índice de clase suministrado desde de la sección de clasificación 624. La sección de determinación de desplazamiento 625 incluye un desplazador inverso de desplazamiento (no ilustrado) que realiza un desplazamiento a la izquierda a nivel de bits en un desplazamiento en una cantidad igual al valor de desplazamiento establecido por la sección de establecimiento de atributos de desplazamiento 613. El desplazador de desplazamiento inverso de desplazamiento realiza una cuantización inversa en el desplazamiento de modo que la profundidad de bits de desplazamiento puede coincidir con la profundidad de bits de píxel. Al realizar dicha cuantización inversa, además del procesamiento realizado por el sumador de desplazamiento 626, que se explicará más adelante, el desplazamiento se puede añadir al valor de píxel utilizando la misma profundidad de bit. El desplazamiento cuantizado inversamente, para cada píxel, se proporciona al sumador de desplazamiento 626.

El sumador de desplazamiento 626 añade un desplazamiento que se proporciona desde de la sección de determinación de desplazamiento 625, a cada píxel que forma la QAOU objeto de la imagen decodificada desbloqueada P_DB. El sumador de desplazamiento 626 genera una imagen obtenida realizando el procesamiento en todas las QAOUs incluidas en la imagen decodificada desbloqueada P_DB como una imagen decodificada de desplazamiento filtrado P_OF.

A continuación, se proporciona una descripción del procesamiento de clasificación realizado por la sección de clasificación 624, con referencia a las Figuras 26 a 29. El procesamiento de clasificación realizado mediante el uso de una condición, como la profundidad jerárquica de una QAOU, como una condición de parámetro, se analiza a continuación. La configuración de una sección de clasificación, basada en condiciones de parámetros generales, se examinará más adelante con referencia a la Figura 34.

La Figura 26 ilustra un procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo 60. La parte (a) de la Figura 26 ilustra gráficos que indican las relaciones de magnitud entre el valor de píxel $\text{pix}[x]$ de un píxel objeto x, y el valor del píxel de un píxel, a o b, y muestra, además, los valores de una función Sign de conformidad con la relación de magnitud. La parte (b) de la Figura 26 ilustra gráficos que indican las relaciones de magnitud entre el valor de píxel del píxel objeto x, y cada uno de los valores de píxel del píxel a, y el píxel b, y

muestra, además, el valor de EdgeType de conformidad con la relación de magnitud. La parte (c) de la Figura 26 ilustra la asociación entre cada gráfico mostrado en la parte (b) de la Figura 26 y class_idx. Las partes (d) a (f) de la Figura 26 ilustran las tablas de transformación utilizadas para la transformación de EdgeType a class_idx.

5 La Figura 27 ilustra el procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo 60: la parte (a) ilustra, de forma esquemática, la clasificación realizada cuando "sao_type_idx =5"; y la parte (b) muestra, esquemáticamente, la clasificación realizada cuando "sao_type_idx=G".

10 La Figura 28 ilustra el procesamiento de desplazamiento realizado por el filtro de desplazamiento adaptativo 60: la parte (a) muestra, esquemáticamente, la clasificación realizada cuando la profundidad jerárquica de una QAOU objeto es menor que un umbral; y la parte (b) ilustra, de forma esquemática, la clasificación realizada cuando la profundidad jerárquica de la QAOU objeto es igual, o mayor, que el umbral.

15 La Figura 29 ilustra un ejemplo de procesamiento de clasificación cuando se especifica un desplazamiento de banda: la parte (a) muestra un ejemplo del procesamiento de clasificación cuando la profundidad jerárquica de una QAOU objeto es menor que un umbral; y la parte (b) ilustra un ejemplo del procesamiento de clasificación cuando la profundidad jerárquica de la QAOU objeto es igual, o mayor, que el umbral.

20 (Cuando el tipo de desplazamiento es uno de entre 1 a 4 (Desplazamiento de Borde))

Cuando el tipo de desplazamiento, que se proporciona desde de la sección de determinación de tipo de desplazamiento 623 es uno de entre 1 a 4, la sección de clasificación 624 realiza el mismo procesamiento que el descrito con referencia a la Figura 10.

25 En consecuencia, tal como se ilustra en la parte (a) de la Figura 26, con respecto a la relación de magnitud entre el valor de píxel pix[x], y el valor de píxel del píxel a o b, cuando el valor de píxel pix[x] es menor que el valor de píxel del píxel a o b, Sign (pix[x] - pix[a]) = -1, cuando el valor de píxel pix[x] es el mismo que el valor de píxel del píxel a o b, Signo (pix[x] - pix[a]) = 0, y cuando el valor de píxel pix[x] es mayor que el valor de píxel del píxel a o b, Sign (pix[x] - pix[a]) = 1. En los gráficos ilustrados en la parte (a) de la Figura 26, el círculo negro con pix[x] indica el valor de píxel del píxel objeto x, y el círculo negro sin pix[x] indica el valor de píxel del píxel objeto a o b. La dirección vertical en los gráficos, que se muestran en la parte (a) de la Figura 26, indica la magnitud del valor de píxel.

30 A continuación, la sección de clasificación 624 encuentra EdgeType de conformidad con la siguiente ecuación matemática (1-1) sobre la base de Sign (pix[x] - pix[a]) y Sign (pix[x] - pix[b]).

35
$$\text{EdgeType} = \text{Sign}(\text{pix}[x] - \text{pix}[a]) + \text{Sign}(\text{pix}[x] - \text{pix}[b]) + 2$$

... (1-1)

40 De conformidad con esta ecuación, tal como se ilustra en la parte (b) de la Figura 26, cuando los valores de píxel de ambos píxeles, a y b, son mayores que el valor de píxel pix[x], EdgeType=0. Cuando el valor de píxel de uno de los píxeles a y b, es mayor que el valor de píxel pix[x] y cuando el valor de píxel del otro píxel, a o b, es el mismo que el valor de píxel pix[x], EdgeType=1. Cuando el valor de píxel de uno de los píxeles, a y b, es menor que el valor de píxel pix[x], y cuando el valor de píxel del otro píxel, a o b, es el mismo que el valor de píxel pix[x], EdgeType=3. Cuando los valores de píxel de ambos píxeles, a y b, son más pequeños que el valor de píxel pix[x], EdgeType=4. Cuando el valor de píxel de uno de los píxeles, a y b, es menor que el valor de píxel pix[x], y cuando el valor de píxel del otro píxel a o b es mayor que el valor de píxel pix[x], o cuando los valores de píxel de ambos píxeles, a y b, son los mismos que el valor de píxel pix[x], EdgeType=2.

50 En la parte (b) de la Figura 26, el círculo negro central en cada gráfico indica el valor de píxel del píxel objeto x, y los círculos negros en ambos lados indican los valores de píxel del píxel a y el píxel b. La dirección vertical en los gráficos que se ilustran en la parte (b) de la Figura 26 indica la magnitud del valor de píxel.

A continuación, sobre la base del EdgeType determinado, la sección de clasificación 624 determina el índice de clase (class_idx) de la clase a la que pertenece el píxel objeto x, de la manera siguiente.

55
$$\text{class_idx} = \text{EoTbl}[\text{EdgeType}]$$

en donde EoTbl [EdgeType] es una tabla de transformación utilizada para la determinación de class_idx desde de EdgeType. Los ejemplos específicos de la tabla de transformación EoTbl se muestran en las partes (d) a (f) de la Figura 26.

60 Una tabla de transformación 1001X, que se muestra en la parte (d) de la Figura 26, es una tabla de transformación que no se cambia de conformidad con la profundidad jerárquica de una QAOU objeto.

Una tabla de transformación 1001A, ilustrada en la parte (e) de la Figura 26, y una tabla de transformación 1001B que se muestra en la parte (f) de la Figura 26, son tablas de transformación que se conmutan de conformidad con la profundidad jerárquica de una QAOU objeto. Si la profundidad jerárquica de una QAOU objeto es menor que un umbral, se utiliza la tabla de transformación 1001A, y si la profundidad jerárquica de la QAOU objeto es igual o mayor que el umbral, se utiliza la tabla de transformación 1001B.

Cuando se utiliza la misma tabla de transformación, sin tener en cuenta la profundidad jerárquica de una QAOU objeto, como lo indica la tabla de transformación 1001X, si no existe un borde en una zona constituida por el píxel objeto x, el píxel a y, el píxel b (en lo sucesivo, este caso se referirá, además, como un caso en el que una zona es plana), es decir, cuando EdgeType=2, la sección de clasificación 624 clasifica el píxel objeto x como clase 0 ("class_idx=0"). La sección de clasificación 624 clasifica, además, EdgeType=0, 1, 3, 4 como class_idx=1, 2, 3, 4, respectivamente.

Cuando la tabla de transformación no se conmuta de conformidad con la profundidad jerárquica de una QAOU objeto, y cuando la profundidad jerárquica de una QAOU objeto es menor que un umbral, tal como lo indica la tabla de transformación 1001A, si no existe un borde en una zona constituida por el píxel objeto x, el píxel a y el píxel b (en adelante, este caso se denominará, además, como un caso en el que una zona es plana), es decir, cuando EdgeType=2, la sección de clasificación 624 clasifica el píxel objeto x como clase 0 ("class_idx=0"). La sección de clasificación 624 clasifica, además, EdgeType=0, 1, 3, 4 como class_idx=1, 3, 4, 2, respectivamente. Cuando la profundidad jerárquica de la QAOU objeto es igual, o mayor, que el umbral, como lo indica la tabla de transformación 1001B, si no existe un borde en una zona constituida por el píxel objeto x, el píxel a y el píxel b (en adelante, este caso se referirá, además, como un caso en el que una zona es plana), es decir, cuando EdgeType=2, la sección de clasificación 624 clasifica el píxel objeto x como clase 0 ("class_idx=0"). La sección de clasificación 624 clasifica, además, EdgeType=0, 1 como class_idx=1 y EdgeType=3, 4 como class_idx=2. En consecuencia, en la tabla de transformación 1001B, un índice de clase (class_id) está asociado con una pluralidad de tipos de borde (EdgeType).

Ha de observarse que la tabla de transformación 1001A se puede formar de la misma manera que la tabla de transformación 1001X. Sin embargo, si la tabla de transformación 1001A es la misma que la tabla de transformación 1001X, se aplican diferentes índices de clase (class_idx) al mismo tipo de borde (EdgeType) en relación con la tabla de transformación 1001B y, el procesamiento varía dependiendo de la profundidad jerárquica. A modo de ejemplo, en el caso de EdgeType=3, cuando la profundidad jerárquica es menor que el umbral, EdgeType=3 se clasifica como class_id=4, y cuando la profundidad jerárquica es igual o mayor que el umbral, EdgeType=3 se clasifica como class_id=2. Por lo tanto, el procesamiento varía dependiendo de la profundidad jerárquica.

En consecuencia, estableciendo la tabla de transformación 1001A para que sea la tabla de transformación 1001B que se muestra en la parte (e) de la Figura 26, se realiza el mismo procesamiento sin tener en cuenta la profundidad jerárquica.

(Cuando el tipo de desplazamiento es 5 o 6 (desplazamiento de banda))

Cuando el tipo de desplazamiento, proporcionado desde de la sección de determinación de tipo de desplazamiento 623, es 5 o 6, la sección de clasificación 624 clasifica el valor de píxel del píxel objeto x como una de entre una pluralidad de clases de conformidad con el valor de píxel $\text{pix}[x]$ del píxel objeto x.

- Cuando el tipo de desplazamiento es 5 (sao_type_idx=5)

Cuando el valor de píxel $\text{pix}[x]$, del píxel objeto x, satisface: $(\text{max} \times 1/4) \leq \text{pix}[x] \leq (\text{max} \times 3/4)$, la sección de clasificación 624 clasifica el píxel objeto como una clase distinta de 0. Es decir, cuando el valor de píxel del píxel objeto está incluido en un margen indicado por la parte sombreada en la parte (a) de la Figura 11, la sección de clasificación 624 clasifica el píxel objeto como una clase distinta de 0. En la condición descrita anteriormente, max indica el valor máximo que puede tomar el píxel objeto x, y por ejemplo, max=255. Cuando max=255, la condición descrita anteriormente se puede representar por $8 \leq (\text{pix}[x]/8) \leq 23$, o $4 \leq (\text{pix}[x]/16) \leq 11$.

- Cuando el tipo de desplazamiento es 6 (sao_type_idx=6)

Cuando el valor de píxel $\text{pix}[x]$, del píxel objeto x, satisface: $\text{pix}[x] \leq (\text{max} \times 1/4)$ o $(\text{max} \times 3/4) \leq \text{pix}[x]$, la sección de clasificación 624 clasifica el píxel objeto como una clase distinta de 0. Es decir, cuando el valor de píxel, del píxel objeto, está incluido en un margen indicado por la parte sombreada en la parte (b) de la Figura 11, la sección de clasificación 624 clasifica el píxel objeto como una clase distinta de 0. En la condición descrita anteriormente, max indica el valor máximo que puede tomar el píxel objeto x, y por ejemplo, max=255. Cuando max=255, la condición anteriormente descrita se puede representar por $(\text{pix}[x]/8) \leq 7$ o $24 \leq (\text{pix}[x]/8)$, o $(\text{pix}[x]/16) \leq 3$ o $12 \leq (\text{pix}[x]/16)$.

El proceso de clasificación realizado por la sección de clasificación 624 se describirá, más concretamente, a continuación.

Cuando el tipo de desplazamiento es 5 o 6, la sección de clasificación 624 determina el índice de clase (class_idx)

de la clase a la que pertenece el píxel objeto x, de conformidad con la profundidad jerárquica de una QAOU objeto, como sigue.

En un caso en el que la profundidad jerárquica de la QAOU objeto es menor que un umbral,

class_idx = EoTbl[sao_type_idx][pix[x]/>>BoRefBit32];
y

En un caso en el que la profundidad jerárquica de la QAOU objeto es igual, o mayor, que el umbral

class_idx = EoTbl[sao_type_idx][pix[x]/>>BoRefBit16]

en donde EoTbl [sao_type_idx] [pix[x]/>> BoRefBit32] y EoTbl [sao_type_idx] [pix[x]/>> BoRefBit16] son tablas de transformación utilizadas para determinar class_idx desde el valor de píxel pix[x] del píxel objeto x y sao_type_idx. Cuando la profundidad de bits de la imagen se indica mediante PIC_DEPTH, BoRefBit32 y BoRefBit16 son valores determinados por PIC_DEPTH-5 y PIC_DEPTH-4, respectivamente, y son valores obtenidos mediante la cuantización de un valor de píxel en 32 o 16 etapas. El valor de píxel cuantizado se indica, además, mediante pixquant. La realización del desplazamiento a la derecha a nivel de bits por BoRefBit32 y BoRefBit16, corresponde a realizar la división por $1 \ll \text{BoRefBit32}$ y $1 \ll \text{BoRefBit16}$, respectivamente. Las anchuras de cuantización de $1 \ll \text{BoRefBit32}$ y $1 \ll \text{BoRefBit16}$ son $1 \ll (8-5)=8$ y $1 \ll (8-4)=16$, respectivamente, cuando la profundidad de bits de los valores de píxel es de ocho bits. El procesamiento de clasificación, cuando la anchura de cuantización es ocho, se examinará a continuación. La anchura de cuantización se refiere como una "anchura de clase".

La sección de clasificación 624 realiza la clasificación cambiando la anchura de clase, en función de la profundidad jerárquica de una QAOU objeto. A modo de ejemplo, tal como se muestra en la parte (a) de la Figura 28, cuando la profundidad jerárquica de una QAOU objeto es menor que un umbral, la sección de clasificación 624 divide los valores de píxel en 32 niveles estableciendo la anchura de clase en "8" y a continuación, realiza la clasificación. Tal como se ilustra en la parte (b) de la Figura 28, cuando la profundidad jerárquica de una QAOU objeto es igual, o mayor, que el umbral, la sección de clasificación 624 divide los valores de píxel en 16 niveles al establecer la anchura de clase a "16" y a continuación, realiza la clasificación.

Ejemplos específicos de la tabla de transformación EoTbl se ilustran en la Figura 29. En las partes (a) y (b) de la Figura 29, "BO_1" indica que "sao_type_index"=5 y, "BO_2" indica que "sao_type_index"= 6. Una tabla de transformación 1301, ilustrada en la parte (a) de la Figura 29, es una tabla de transformación cuando la profundidad jerárquica de una QAOU objeto es más pequeña que un umbral, y una tabla de transformación 1302, que se muestra en la parte (b) de la Figura 29, es una tabla de transformación cuando la profundidad jerárquica de una QAOU objeto es igual, o mayor, que el umbral.

Cuando la profundidad jerárquica de una QAOU objeto es menor que el umbral, y cuando "sao_type_index"=5, tal como se muestra en la parte (a) de la Figura 29, la sección de clasificación 624 clasifica el píxel objeto x como uno de los índices de clase 1 a 16, de conformidad con la magnitud de pix[x], si el valor de píxel pix[x], del píxel objeto x satisface la condición $8 < (\text{pix}[x]/8) \leq 23$.

Cuando "sao_type_idx"=6, la sección de clasificación 624 clasifica el píxel objeto x como uno de los índices de clase 1 a 16, de conformidad con la magnitud de pix[x], si el valor de píxel pix[x] del píxel objeto x cumple la condición $(\text{pix}[x]/8) \leq 7$ o $24 \leq (\text{pix}[x]/8)$.

Cuando la profundidad jerárquica de una QAOU objeto es igual, o mayor, que el umbral y cuando "sao_type_index"=5, tal como se ilustra en la parte (b) de la Figura 29, la sección de clasificación 624 clasifica el píxel objeto x como uno de los índices de clase 1 a 8 de conformidad con la magnitud de pix[x], si el valor de píxel pix[x] del píxel objeto x, satisface la condición $4 \leq (\text{pix}[x]/16) \leq 11$.

Cuando "sao_type_idx"=6, la sección de clasificación 624 clasifica el píxel objeto x como uno de los índices de clase 1 a 8 de conformidad con la magnitud de pix[x], si el valor de píxel pix[x], del píxel objeto x cumple la condición $(\text{pix}[x]/16) \leq 3$ o $12 \leq (\text{pix}[x]/16)$.

(Configuración de la sección de decodificación de información de desplazamiento 611)

La Figura 41 es un diagrama de bloques de la sección de decodificación de información de desplazamiento 611 que cambia el tipo de desplazamiento que ha de utilizarse de conformidad con un tipo de parámetro (condición de parámetro) de un desplazamiento adaptativo y/o, que cambia el número de clases de conformidad con una condición de parámetro de un desplazamiento adaptativo. La sección de decodificación de información de desplazamiento 611 incluye un decodificador de tipo de desplazamiento adaptativo 6111, un selector de tipo de desplazamiento 6112, un decodificador de tipo de desplazamiento 6113, un decodificador de desplazamiento adaptativo 6114, un selector de número de desplazamientos 6115, y un decodificador de desplazamiento 6116. Las condiciones de parámetro son parámetros distintos de los valores calculados desde de valores de píxel. Ejemplos de las condiciones de

parámetros son un tamaño de bloque (tamaño de QAOU), un componente de color (component) y QP, que se analizarán en los apéndices más adelante, además de la profundidad jerárquica y el tipo de desplazamiento descritos anteriormente.

5 El decodificador de tipo de desplazamiento adaptativo 6111 es un medio para decodificar, de forma adaptativa, los tipos de desplazamiento desde de información de QAOU incluida en los datos codificados de conformidad con una condición de parámetro, e incluye el selector de tipo de desplazamiento 6112, y el decodificador de tipo de desplazamiento 6113.

10 El selector de tipo de desplazamiento 6112 es un medio para seleccionar los tipos de desplazamiento que han de utilizarse de conformidad con la condición de parámetro. Una de las condiciones de parámetro es la profundidad jerárquica descrita anteriormente. El número de tipos de desplazamiento que se utilizarán cuando la profundidad jerárquica no sea profunda es menor que cuando la profundidad jerárquica es profunda. El selector de tipo de desplazamiento 6112 introduce una tabla de transformación, como la analizada con referencia a la Figura 28, para el
15 decodificador de tipo de desplazamiento 6113. El selector de tipo de desplazamiento 6112 introduce, además, el número máximo de tipos de desplazamiento utilizables para el decodificador de tipo de desplazamiento 6113.

El decodificador de tipo de desplazamiento 6113 decodifica tipos de desplazamiento de la tabla de transformación de entrada y el número máximo. Si el número máximo de tipos de desplazamiento es N, el margen de códigos que se pueden tomar se restringe a N, desde 0 a N-1, lo que reduce el número de bits necesarios para la codificación de
20 códigos. A modo de ejemplo, cuando el valor máximo es mayor que 2^{m-1} , pero no mayor que 2^m , se puede utilizar la codificación de longitud fija de m-bit. Puede utilizarse una codificación unitaria truncada o codificación de corte truncado que tenga N-1 como el valor máximo.

25 El decodificador de desplazamiento adaptativo 6114 es un medio para decodificar, de forma adaptativa, desplazamientos de información de QAOU, que se incluyen en los datos codificados de conformidad con una condición de parámetro, e incluye el selector de número de desplazamientos 6115 y el decodificador de desplazamiento 6116. El selector de número de desplazamientos 6115 es un medio para seleccionar el número máximo de desplazamientos que han de utilizarse, y la precisión de desplazamiento de conformidad con una
30 condición de parámetro. Una de las condiciones del parámetro es la profundidad jerárquica. El número de desplazamientos cuando la profundidad jerárquica no es profunda es mayor que cuando la profundidad jerárquica es profunda. A modo de ejemplo, cuando la profundidad jerárquica no es profunda, se pueden utilizar dieciséis desplazamientos, tal como se ilustra en la parte (a) de la Figura 29, y cuando la profundidad jerárquica es profunda, se pueden utilizar ocho desplazamientos, según se muestra en la parte (b) de la Figura 29. Además, tal como se
35 indicó con anterioridad, la precisión de desplazamiento (profundidad de bit) se puede cambiar de conformidad con la profundidad jerárquica. El selector de número de desplazamientos 6115 introduce el número máximo de desplazamientos y la precisión de desplazamiento al decodificador de desplazamiento 6116. El decodificador de desplazamiento 6116 decodifica desplazamientos de conformidad con el número máximo de desplazamientos y la precisión de desplazamiento. Si se reduce el número de desplazamientos, se reduce la cantidad de datos para
40 codificar desplazamientos. Además, como en el tipo de desplazamiento, si se determina la precisión de cada desplazamiento, se restringe, además, el margen de un código para codificar cada desplazamiento que se puede tomar, lo que reduce el número de bits necesarios para codificar un código.

La parte (a) de la Figura 42 es un diagrama de bloques de una configuración del selector de tipo de desplazamiento
45 6112.

El selector de tipo de desplazamiento 6112 incluye un selector de tabla de transformación de tipo de desplazamiento 6117, una primera sección de memorización de tabla de transformación de tipo de desplazamiento 6118, y una
50 segunda sección de memorización de tabla de transformación de tipo de desplazamiento 6119.

El selector de tabla de transformación de tipo de desplazamiento 6117 selecciona una tabla de transformación que se memoriza en la primera sección de memorización de tabla de transformación de tipo de desplazamiento 6118, o una tabla de transformación memorizada en la segunda sección de memorización de tabla de transformación de tipo de desplazamiento 6119, de conformidad con una condición de parámetro. En el ejemplo descrito con anterioridad,
55 la tabla de transformación 801A es la tabla de transformación memorizada en la primera sección de memorización de tabla de transformación de tipo de desplazamiento 6118, y la tabla de transformación 801B es la tabla de transformación que se memoriza en la segunda sección de memorización de tabla de transformación de tipo de desplazamiento 6119.

60 La parte (b) de la Figura 42 es un diagrama de bloques de otra configuración del selector de tipo de desplazamiento.

Un selector de tipo de desplazamiento 6112' incluye un selector de tabla de transformación de tipo de desplazamiento 6117', una sección de memorización de tabla de transformación de tipo de desplazamiento de borde y tipo de desplazamiento de banda 6118', y una sección de memorización de tabla de transformación de tipo de
65 desplazamiento de banda y tipo de desplazamiento de borde horizontal 6119'.

(Configuración de sección de clasificación 624)

La Figura 43 es un diagrama de bloques de la configuración de la sección de clasificación 624.

5 La sección de clasificación 624 incluye una sección de clasificación de desplazamiento de borde adaptativo 6241, un selector de clase de desplazamiento de borde 6242, una sección de clasificación de desplazamiento de borde 6243, una sección de clasificación de desplazamiento de banda adaptativo 6244, un selector 6245 de anchura de clase y clase de desplazamiento de banda 6246.

10 La sección de clasificación 624 clasifica cada píxel como una clase de conformidad con una condición de parámetro y un tipo de desplazamiento. Cuando el tipo de desplazamiento indica un desplazamiento de borde, la sección de clasificación 624 clasifica cada píxel utilizando la sección de clasificación de desplazamiento de borde adaptativo 6241. Cuando el tipo de desplazamiento indica un desplazamiento de banda, la sección de clasificación 624 clasifica cada píxel utilizando la sección de clasificación de desplazamiento de banda adaptativo 6244.

15 La sección de clasificación de desplazamiento de borde adaptativo 6241 es un medio para clasificar, de forma adaptativa, cada píxel como una clase de conformidad con una condición de parámetro, e incluye el selector de clase de desplazamiento de borde 6242, y la sección de clasificación de desplazamiento de borde 6243. El selector de clase de desplazamiento de borde 6242 selecciona el tipo de clase que ha de utilizarse. El selector de clase de desplazamiento de borde 6242 introduce un método de clasificación para clasificar valores de píxel en la sección de clasificación de desplazamiento de borde 6243. Más concretamente, el selector de clase de desplazamiento de borde 6242 introduce un método para determinar un valor intermedio EdgeType que se determina, temporalmente, cuando un valor de píxel se clasifica, y se utiliza una tabla de transformación para determinar class_idx de EdgeType. Un ejemplo del método para determinar el valor intermedio EdgeType se ha discutido con referencia a partes (a) y (b) de la Figura 26 y, esto se refiere como un "método básico de clasificación de borde". Como alternativa, se puede utilizar un método de clasificación de borde, que se explicará más adelante con referencia a la Figura 33. Una de las condiciones del parámetro es la profundidad jerárquica, según se indicó anteriormente, y el número de clases que se utilizarán cuando la profundidad jerárquica no sea profunda se establece para ser mayor que cuando la profundidad jerárquica es profunda. Ejemplos de la tabla de transformación, utilizada para determinar class_idx de EdgeType, son las tablas de transformación 1001A y 1001B. Se puede utilizar, adecuadamente, un método para cambiar el método de determinación de borde de conformidad con el componente de color (component). En este caso, para los componentes de luminancia, se utiliza, de forma adecuada, el método básico de clasificación de borde para clasificar bordes en las direcciones horizontal, vertical y oblicua y, para los componentes de crominancia, es adecuado utilizar un método de clasificación de borde horizontal con el fin de reducir el número de memorias de línea. En el método de clasificación de borde horizontal, el margen de píxels de referencia, utilizados para la clasificación de bordes, está restringido a la dirección horizontal con respecto a un píxel objeto. La sección de clasificación de desplazamiento de borde 6243 clasifica píxels sobre la base del método de clasificación proporcionado, y la tabla de transformación utilizada para determinar class_idx desde EdgeType.

40 La sección de clasificación de desplazamiento de banda adaptativa 6244 es un medio para clasificar, de forma adaptativa, cada píxel como una clase de conformidad con una condición de parámetro, e incluye el selector de anchura de clase y clase de desplazamiento de banda 6245, y la sección de clasificación de desplazamiento de banda 6246. El selector de anchura de clase y clase de desplazamiento de banda 6245 introduce un método de clasificación para clasificar valores de píxel en la sección de clasificación de desplazamiento de banda 6246. Más concretamente, el selector de anchura de clase y clase de desplazamiento de banda 6245 introduce una anchura de clase, que es una anchura de cuantización utilizado para clasificar un valor de píxel como un valor intermedio, y una tabla de transformación utilizada para determinar class_idx desde del valor intermedio. Una de las condiciones del parámetro es la profundidad jerárquica, según se indicó anteriormente y la anchura de clase que se utilizará cuando la profundidad jerárquica no sea profunda es menor que cuando la profundidad jerárquica es profunda. La sección de clasificación de desplazamiento de banda 6246 clasifica un valor de píxel como una clase de conformidad con la anchura de la clase recibido, y la tabla de transformación utilizada para determinar class_idx desde del valor intermedio. La anchura de la clase, que es una entrada, no tiene que ser una anchura de clase en sí misma, y puede ser un número entero correspondiente a la anchura de clase. A modo de ejemplo, si se utiliza $1 \ll \text{BoRefBit32}$ como la anchura de clase, el número de bits BoRefBit32, que es el logaritmo de la base 2, utilizado para realizar una cuantización de un píxel que se puede utilizar en lugar de la anchura de clase, o el valor para determinar el número de bits utilizados para realizar una cuantización a un píxel desde de la profundidad de bits de píxel, a modo de ejemplo, si $\text{BoRefBit32} = \text{PIC_DEPTH} - 5$, se puede utilizar 5 en lugar de la anchura de clase.

60 De este modo, los datos codificados se decodifican, de forma adaptativa, de conformidad con una condición de parámetro, y se realiza la clasificación de píxels.

Tal como se describió con anterioridad, en esta forma de realización, el número de clases que se utiliza para clasificar píxels se cambia de conformidad con la profundidad jerárquica de una QAOU objeto. Más concretamente, cuando la profundidad jerárquica de una QAOU objeto es profunda, el número de clases es menor que cuando la profundidad jerárquica de la QAOU objeto no es profunda.

En general, el espacio de memoria requerido se representa mediante: el espacio de memoria = la longitud de datos de un desplazamiento $\times$ el número de clases $\times$ el número de QAOUs. En consecuencia, al disminuir el número de clases, se puede reducir, además, el espacio de memoria que ha de utilizarse.

- 5 En un nivel jerárquico más profundo, la zona de una QAOU se vuelve más pequeña y, además, las características de valores de píxel dentro de dicha QAOU, son casi uniformes. En consecuencia, incluso si el número de clases disminuye, el efecto del desplazamiento no se ve afectado, de forma considerable.

10 Además, en esta forma de realización, la precisión de desplazamiento se cambia de conformidad con la profundidad jerárquica de una QAOU objeto. Más concretamente, cuando la profundidad jerárquica de una QAOU objeto es profunda, la precisión de desplazamiento es menor que cuando la profundidad jerárquica de una QAOU objeto no es profunda. Con esta disposición, se puede reducir la cantidad de codificación requerida cuando la profundidad jerárquica de una QAOU objeto es profunda.

- 15 En un nivel jerárquico más profundo de una QAOU, el número de píxeles incluidos en dicho nivel jerárquico es menor y, por lo tanto, los errores de cuantización en la QAOU completa también son más pequeños que los de un nivel jerárquico superior. En consecuencia, incluso si se reduce la precisión de desplazamiento en un nivel jerárquico más profundo, el efecto de los desplazamientos no se ve afectado de forma considerable.

20 (Dispositivo de codificación de vídeo 2')

25 Un dispositivo de codificación de vídeo 2', para la generación de datos codificados nº 1, mediante la codificación de una imagen objeto se describirá a continuación con referencia a las Figuras 30 y 31. Como en el dispositivo de codificación de vídeo 2, el dispositivo de codificación de vídeo 2' incluye, de forma parcial, un método definido en H. 264/MPEG-4. AVC, un método utilizado en el software KTA, que es un códec de desarrollo conjunto en VCEG (Grupo de Expertos de Codificación de Vídeo), un método utilizado en el software TMuC (Modo de Prueba bajo Consideración), que es un códec sucesor del códec utilizado en el software KTA, y una tecnología utilizada en el software HM (HEVC TestModel).

- 30 El dispositivo de codificación de vídeo 2' incluye un filtro de desplazamiento adaptativo 80' en lugar del filtro de desplazamiento adaptativo 80 del dispositivo de codificación de vídeo 2. Los otros elementos de la configuración del dispositivo de codificación de vídeo 2' son similares a los del dispositivo de codificación de vídeo 2.

35 (Filtro de desplazamiento adaptativo 80')

Un filtro de desplazamiento adaptativo 80' se examinará, a continuación, haciendo referencia a la Figura 30. La Figura 30 es un diagrama de bloques que ilustra la configuración del filtro de desplazamiento adaptativo 80'. El filtro de desplazamiento adaptativo 80' incluye, según se ilustra en la Figura 30, una unidad de establecimiento de información de filtro de desplazamiento adaptativo 81', y un procesador de filtro de desplazamiento adaptativo 82'.

40 La unidad de establecimiento de información de filtro de desplazamiento adaptativo 81' incluye, tal como se ilustra en la Figura 30, un calculador de desplazamiento 811, un desplazador de desplazamiento 816, una sección de recorte de desplazamiento 812, un selector de información de desplazamiento 813 y una sección de establecimiento de atributos de desplazamiento 815.

45 (Calculador de desplazamiento 811)

El calculador de desplazamiento 811 calcula desplazamientos con respecto a todos los tipos de desplazamiento, y todas las clases, para todas las QAOUs hasta una profundidad de división predeterminada, incluida en la unidad de procesamiento (a modo de ejemplo, una LCU) de conformidad con la profundidad jerárquica de una QAOU objeto. En este caso, los tipos de desplazamiento, y las clases, son los mismos que los examinados en una descripción del dispositivo de decodificación de vídeo 1.

50 El calculador de desplazamiento 811 proporciona información de desplazamiento que indica desplazamientos calculados por el procesamiento descrito anteriormente, tipos de desplazamiento, clases e información de estructura de QAOU que indica la estructura de división de QAOU a la sección de recorte de desplazamiento 812.

55 (Sección de recorte de desplazamiento 812)

60 La sección de recorte de desplazamiento 812 realiza un procesamiento de recortes utilizando uno de los siguientes primer procesamiento de recorte, y segundo procesamiento de recorte, en desplazamientos que se proporcionan desde del calculador de desplazamientos 811.

65 (Primer procesamiento de recorte)

La sección de recorte de desplazamiento 812 recorta cada desplazamiento suministrado desde del calculador de

desplazamiento 811 a, por ejemplo, valores desde -8 a 7, expresando así cada desplazamiento en cuatro bits. Los desplazamientos recortados se proporcionan al selector de información de desplazamiento 813. La anchura de bit utilizado para el recorte, se establece de conformidad con la profundidad de bits de imagen y la profundidad de bits de desplazamiento, como en el dispositivo de decodificación de vídeo 1.

Al recortar cada desplazamiento de esta manera, se puede reducir el tamaño de memoria de una memoria (no ilustrado) para memorizar cada desplazamiento. Además, se puede reducir la cantidad de datos necesarios para codificar desplazamientos incluidos en los datos codificados nº 1, lo que hace posible mejorar la eficacia de codificación. Además, puesto que no se añaden excesivos desplazamientos, se puede garantizar un nivel adecuado de calidad de imagen.

(Segundo procesamiento de recorte)

La sección de recorte de desplazamiento 812 puede establecer, de manera diferente, el margen de recorte de cada desplazamiento suministrado desde el calculador de desplazamiento 811, de conformidad con el tipo de desplazamiento.

A modo de ejemplo, si el tipo de desplazamiento es un desplazamiento de borde, el número de bits de un desplazamiento se establece para ser ocho bits y, si el tipo de desplazamiento es un desplazamiento de banda, el número de bits de un desplazamiento se establece en cuatro bits. Más generalmente, cuando el número de bits de un desplazamiento de borde es N bits, y el número de bits de un desplazamiento de banda es M bits, los números de bits de desplazamientos se determinan de modo que se cumpla la condición $N > M$.

De este modo, al variar el número de bits de un desplazamiento de conformidad con el tipo de desplazamiento, se puede mejorar la eficacia de codificación sin requerir un tamaño de memoria excesivo de una memoria para almacenar cada desplazamiento.

Cuando el umbral th para valores de restricción que pueden tomarse como un desplazamiento es mayor que 2^{m-1} , pero no mayor que 2^m , se puede utilizar el método de codificación de longitud fija de m-bit como el método de codificación para codificar el desplazamiento. Más concretamente, se puede utilizar la codificación unitaria truncada o la codificación de corte truncado teniendo th como el valor máximo.

El procesamiento de recortes, realizado por una combinación del primer procesamiento de recorte y el segundo procesamiento de recorte, anteriormente descritos, también está incluido en esta forma de realización. El filtro de desplazamiento adaptativo 80' puede no incluir la sección de recorte de desplazamiento 812.

La sección de recorte de desplazamiento 812 cambia el margen de recorte de conformidad con la profundidad jerárquica de una QAOU objeto, de modo que coincida con la precisión de desplazamiento. Más concretamente, se considera un caso en el que la precisión de desplazamiento es n_A bits cuando la profundidad jerárquica de una QAOU objeto es menor que un umbral, y la precisión de desplazamiento es n_B bits cuando la profundidad jerárquica de una QAOU objeto, es igual o mayor que el umbral. En este caso, cuando la profundidad jerárquica de una QAOU objeto es menor que el umbral, la sección de recorte de desplazamiento 812 establece que el margen de recorte sea $-2^{n_A/2}$ a $2^{n_A/2}-1$. Cuando la profundidad jerárquica de una QAOU objeto es igual o mayor que el umbral, la sección de recorte de desplazamiento 812 establece que el margen de recorte sea $-2^{n_B/2}$ a $2^{n_B/2}-1$ y a continuación, establece los bits más bajos ($n_A - n_B$) para ser "0". Lo que antecede permite que el selector de información de desplazamiento 813 realice el procesamiento utilizando desplazamientos sometidos al procesamiento de recorte realizado por la sección de recorte de desplazamiento 812 sin tener en cuenta la precisión del desplazamiento.

(Selector de información de desplazamiento 813)

El selector de información de desplazamiento 813 determina una combinación de un tipo de desplazamiento, clases, y desplazamientos que minimizan el coste de RD (costo de Distorsión-Tasa) y la estructura de división de QAOU asociada, y proporciona información de QAOU que indica el tipo de desplazamiento determinado, clases, y desplazamientos, y la estructura de división de QAOU asociada al codificador de código de longitud variable 22. El selector de información de desplazamiento 813 proporciona, además, los desplazamientos determinados para cada QAOU al procesador de filtro de desplazamiento adaptativo 82.

El procesamiento realizado por el selector de información de desplazamiento 813 se examinará, más concretamente, con referencia a la Figura 31.

La Figura 31 muestra una visión general de un caso en el que se calcula un error cuadrático con respecto a una QAOU que tiene un índice de QAOU "x", con respecto a cada tipo de desplazamiento. Tal como se muestra en la Figura 31, el selector de información de desplazamiento 813 calcula un error cuadrático con respecto a cada QAOMU en relación a cada tipo de desplazamiento. A continuación, el selector de información de desplazamiento 813 establece el tipo de desplazamiento que minimiza el error cuadrático para que sea el tipo de desplazamiento de la QAOU. Como resultado, se determinan tipos de desplazamiento para la totalidad de QAOMUs (números de

QAOMU de 0 a 340).

A continuación, el selector de información de desplazamiento 813 calcula el coste de RD cuando la profundidad de división es 0 y el coste de RD cuando la profundidad de división es 1. Un método de cálculo específico es el mismo que el descrito con referencia a la Figura 18.

(Procesador de filtro de desplazamiento adaptativo 82')

El procesador de filtro de desplazamiento adaptativo 82', añade un desplazamiento suministrado desde del selector de información de desplazamiento 813 a cada píxel de una QAOU objeto, en la imagen decodificada desbloqueada P_DB. El procesador de filtro de desplazamiento adaptativo 82' genera, como una imagen decodificada de desplazamiento filtrado P_OF, una imagen obtenida mediante la realización del procesamiento en la totalidad de QAOUs incluidas en la imagen decodificada desbloqueada P_DB. La configuración del procesador de filtro de desplazamiento adaptativo 82' es la misma que la del procesador de filtro de desplazamiento adaptativo 62' y, por lo tanto, se omite aquí una explicación del mismo.

Como alternativa, se puede utilizar un tipo de recorte adaptativo. Es decir, como uno de los tipos de desplazamiento, se puede utilizar un tipo de recorte adaptativo (AC). En este caso, existen tres tipos de desplazamiento, EO, BO y AC. En el tipo de recorte adaptativo (AC), se corrige un valor de píxel mediante la utilización de recortes, tal como un valor límite inferior c1, y un valor límite superior c2, sin utilizar desplazamientos. Ejemplos del valor límite inferior c1, y el valor límite superior c2, son c1=16 y c2=235, respectivamente.

La utilización del tipo de recorte adaptativo elimina la necesidad de realizar un procesamiento de codificación de desplazamiento, y proporcionar una memoria para almacenar numerosos desplazamientos. Si el valor límite inferior y el valor límite superior que se utilizan para el recorte se proporcionan, de forma adaptativa, en lugar de utilizar valores fijos, se codifican. En este caso, se pueden codificar, de forma preferentemente, diferencias desde de valores fijos adecuados, a modo de ejemplo, un desplazamiento desde 16, en el caso del valor límite inferior, y un desplazamiento desde 235 en el caso del valor límite superior. El valor límite superior, en particular, puede convertirse en un valor grande. Sin embargo, no disminuye la eficacia de codificación al realizar la codificación de este modo.

(Apéndice 2)

Tal como se indicó anteriormente, "de conformidad con la profundidad jerárquica de una QAOU" se puede leer como "de conformidad con el tamaño de una QAOU". Es decir, el número de tipos de SAO, el número de clases de SAO y la precisión de desplazamiento se pueden cambiar de conformidad con el tamaño de una QAOU.

A modo de ejemplo, si el tamaño de una QAOU es menor que N×N píxels, uno o una pluralidad de factores, tales como el número de tipos de SAO, el número de clases de SAO y la precisión de desplazamiento, se puede restringir mucho más que cuando el tamaño de una QAOU es igual o mayor que N×N píxels. Un valor específico de N es, por ejemplo, N=64 (tamaño de una LCU).

El número de tipos de SAO puede estar restringido, a modo de ejemplo, como sigue: EO está restringido a un tipo horizontal o vertical, BO está restringido a un tipo (tabla de transformación 802C), solamente se utiliza EO, y no se usa BO (tabla de transformación 802A) y, solamente se utiliza EO horizontal y algunos modos de BO (tabla de transformación 801B o 802D). Las tablas de transformación indicadas entre paréntesis se utilizan en la sección de decodificación de información de desplazamiento 611. La restricción del número de clases de SAO, y la precisión de desplazamiento, se pueden imponer, de modo similar, a las formas de realización descritas anteriormente.

Al restringir, de esta manera, los modos de los tipos de SAO, el número de tipos de SAO, el número de clases de SAO, y la precisión de desplazamiento, se puede reducir el espacio de memoria que se va a utilizar y se puede reducir la carga de procesamiento. En particular, si los modos de tipos de SAO están restringidos a BO, o si EO está restringido a un tipo horizontal o vertical, se puede reducir la carga de procesamiento en el dispositivo de codificación y la sección de clasificación del dispositivo de decodificación. Si el número de tipos está restringido, los cálculos de costos para seleccionar un tipo óptimo pueden reducirse, con lo que se reduce la carga de procesamiento en el dispositivo de codificación. Si los modos de tipos de SAO están restringidos a EO, o si el número de clases y la precisión de desplazamiento están restringidos, se puede reducir el espacio de memoria que se utilizará para memorizar los desplazamientos. Además, al restringir los modos de los tipos SAO a EO horizontal y BO, se puede reducir el número de memorias temporales, tal como memorias de línea para memorizar píxels de referencia utilizados para la clasificación y, además, se puede reducir un retardo que se produce mientras se espera la decodificación de píxels de referencia. Con el fin de reducir el espacio de memoria que ha de utilizarse, es más efectivo si el número de clases de BO está restringido que si se restringe el número de clases de EO.

Las restricciones descritas con anterioridad se pueden poner en práctica mediante la configuración del medio ilustrado en las Figuras 41 a 43, si las operaciones de la sección de decodificación de información de desplazamiento 611, y la sección de clasificación 624 se cambian, de forma adaptativa, de conformidad con el

tamaño de una QAOUN que se utiliza como una condición de parámetro.

(Apéndice 3)

Además, el tamaño de una QAOU que ha de utilizarse cuando se añade un desplazamiento a un valor de luminancia (en adelante referido, además, como una "unidad de luminancia") se puede hacer diferente del tamaño de una QAOU que ha de utilizarse cuando se añade un desplazamiento a un valor de crominancia (en lo sucesivo, también denominado como una "unidad de crominancia"). A continuación, los modos de tipos de SAO, el número de tipos de SAO, el número de clases de SAO, la precisión de desplazamiento, y el nivel jerárquico de división máximo se puede restringir en una unidad de crominancia más que en una unidad de luminancia.

A modo de ejemplo, en un formato de datos en el que la resolución de la luminancia es diferente de la crominancia, y la resolución de la crominancia es menor que la de la luminancia, como en el formato de imagen YUV 4: 2: 0, no es necesario dividir una unidad de crominancia a un nivel más pequeño que una unidad de luminancia. En consecuencia, al restringir los modos de tipos de SAO, el número de tipos de SAO, el número de clases de SAO, y la precisión de desplazamiento, y el nivel jerárquico de división máximo de una unidad de crominancia, más que los de una unidad de luminancia, se puede disminuir un espacio de memoria que ha de utilizarse, y se puede reducir la carga de procesamiento. El nivel jerárquico de división máximo, de una unidad de crominancia, se puede restringir haciendo que la profundidad jerárquica máxima de una estructura de árbol de SAO, sea menor que la de una unidad de luminancia.

Las restricciones anteriormente descritas se pueden poner en práctica mediante la configuración del medio ilustrado en las Figuras 41 a 43, si las operaciones de la sección de decodificación de información de desplazamiento 611, y la sección de clasificación 624, se cambian de forma adaptativa, de conformidad con el tamaño de una QAOUN utilizado como condición de parámetro.

En particular, cuando se realiza el procesamiento de SAO, tanto para la luminancia como para la crominancia, se requiere para cada componente de color una memoria temporal, tal como una memoria de línea para memorizar pixels de referencia para la clasificación. Puesto que el componente de luminancia es menos importante que los componentes de crominancia, los componentes de crominancia se utilizan como una condición de parámetro, y los modos de tipos de SAO están restringidos a EO horizontal y BO, lo que hace posible reducir el número de memorias de línea para los componentes de crominancia. En este caso, en la sección de decodificación de información de desplazamiento 611, se utiliza una tabla de transformación, como la tabla de transformación 801B o 802D.

Además, resulta particularmente efectivo que los componentes de crominancia se utilicen como una condición de parámetro y la precisión de desplazamiento para los componentes de crominancia, se establece para ser inferior que la del componente de luminancia. A modo de ejemplo, si la precisión de desplazamiento se establece para el componente de luminancia para ser:

$$\text{SAO_DEPTH} = \text{MIN}(\text{PIC_DEPTH}, \text{AY})$$

$$\text{shift value} = \text{PIC_DEPTH} - \text{MIN}(\text{PIC_DEPTH}, \text{AY});$$

y la precisión de desplazamiento y el valor de desplazamiento para los componentes de crominancia se establecen en:

$$\text{SAO_DEPTH} = \text{MIN}(\text{PIC_DEPTH}, \text{THC})$$

$$\text{shift value} = \text{PIC_DEPTH} - \text{MIN}(\text{PIC_DEPTH}, \text{AC}),$$

las variables AY y AC para controlar la precisión se determinan, adecuadamente, de modo que puedan satisfacer $\text{AY} > \text{AC}$. A modo de ejemplo, AY se establece para ser 10 o 9 y, AC se establece en 8. En este caso, la precisión de desplazamiento para el componente de luminancia es mayor que la de los componentes de crominancia. El valor de desplazamiento para el componente de luminancia, es más bajo que el de los componentes de crominancia.

(Apéndice 4)

Además, cuando el valor de un parámetro de cuantización QP (valor QP) para una CU es igual o mayor que un umbral, los modos de tipos de SAO, el número de tipos de SAO, el número de clases de SAO, la precisión de desplazamiento, y el nivel jerárquico de división máximo, puede estar restringido. Como el valor QP, se puede utilizar el valor QP de la parte iniciación de una imagen, o si una QAOU que ha de someterse al procesamiento de SAO se enfrenta a un límite de una LCU o una CU, se puede utilizar el valor QP correspondiente a la coordenada superior izquierda, o la coordenada central de la QAOU.

A medida que aumenta el valor QP, la calidad de imagen de una imagen de predicción se vuelve más baja y es más

5 difícil realizar un procesamiento y corrección de clasificación detallados utilizando desplazamientos. En consecuencia, incluso si los modos de tipos de SAO, el número de tipos de SAO, el número de clases de SAO, la precisión de desplazamiento y el nivel jerárquico de división máximo están restringidos, la influencia en la calidad de la imagen es pequeña. Por lo tanto, cuando el valor QP es igual o mayor que un umbral, el número de tipos de SAO, el número de clases de SAO, la precisión de desplazamiento y el nivel jerárquico de división máximo están restringidos. Entonces, es posible disminuir un espacio de memoria que se va a utilizar y reducir la carga de procesamiento sin influir en la calidad de la imagen.

10 Las restricciones descritas anteriormente se pueden poner en práctica mediante la configuración del medio ilustrado en las Figuras 41 a 43 si las operaciones de la sección de decodificación de información de desplazamiento 611, y la sección de clasificación 624, se cambian de forma adaptativa, de conformidad con QP, que se utiliza como condición de parámetro.

15 (Apéndice 5)

Por otra parte, para un tipo de imagen específico, a modo de ejemplo, imágenes B e imágenes no de referencia (imágenes IDR), los modos de tipos de SAO, el número de tipos de SAO, el número de clases de SAO, la precisión de desplazamiento y el nivel jerárquico de división máximo, pueden estar restringido.

20 En lo que respecta a las imágenes I y P, la calidad de imagen de estas imágenes influye, en gran medida, en las imágenes subsiguientes y, por lo tanto, es necesario mantener la precisión de SAO a un alto nivel. Por otro lado, imágenes distintas a las imágenes I y las imágenes P, no influyen mucho en las imágenes posteriores y, por lo tanto, en relación con estas imágenes (imágenes B e imágenes IDR), se pueden restringir los modos de tipos de SAO, el número de tipos de SAO, el número de clases de SAO, la precisión de desplazamiento y el nivel jerárquico de división máximo. De este modo, es posible disminuir un espacio de memoria que se va a utilizar y reducir la carga de procesamiento.

30 Las restricciones anteriormente descritas se pueden poner en práctica mediante la configuración del medio ilustrado en las Figuras 41 a 43 si las operaciones de la sección de decodificación de información de desplazamiento 611, y la sección de clasificación 624, se cambian de forma adaptativa, de conformidad con un tipo de imagen utilizado como una condición de parámetro.

(Apéndice 6)

35 Además, de conformidad con la posición de una QAOU en una pantalla, el número de tipos de SAO, el número de clases de SAO, la precisión de desplazamiento y el nivel jerárquico de división máximo, pueden estar restringidos.

40 A modo de ejemplo, en la periferia de una pantalla, el número de tipos de SAO, el número de clases de SAO, la precisión de desplazamiento y el nivel jerárquico de división máximo, pueden estar restringidos.

45 Es probable que un usuario enfoque en y alrededor del centro de la pantalla y, por lo tanto, una disminución en la calidad de imagen, en esta parte, refleja directamente la calidad de imagen subjetiva. Sin embargo, incluso si la calidad de la imagen alrededor de la periferia de la pantalla disminuye, la calidad de imagen objeto no disminuye al mismo nivel que cuando se reduce en, y alrededor, del centro de la pantalla.

Además, si no se utiliza un tipo de SAO que requiere puntos de muestra más allá del borde de una pantalla, la carga en el procesamiento de determinación de límites se puede reducir.

50 Las restricciones descritas anteriormente se pueden poner en práctica mediante la configuración del medio ilustrado en las Figuras 41 a 43, si las operaciones de la sección de decodificación de información de desplazamiento 611, y la sección de clasificación 624, se cambian de forma adaptativa, de conformidad con la posición de una QAOU que se utiliza como una condición de parámetro.

55 Cuarta Forma de realización

A continuación, se describe otra forma de realización de la presente invención haciendo referencia a la Figura 32. Una cuarta forma de realización es diferente de las formas de realización descritas anteriormente por cuanto que EO y BO se conmutan de conformidad con un valor de píxel.

60 Los presentes inventores han encontrado que existen características en las que EO es más eficaz que BO en una zona de tono intermedio de píxeles, y BO es más eficaz que EO en las otras zonas (una zona de valor de píxel inferior y una zona de valor de píxel superior). Por consiguiente, en esta forma de realización, BO se utiliza en la zona de valor de píxel inferior, y la zona de valor de píxel superior, y EO se utiliza en la zona de tono intermedio.

65 Lo que antecede se describirá, a continuación, más concretamente, con referencia a la Figura 32. La Figura 32 ilustra la configuración en la que EO y BO se conmutan de conformidad con el valor de píxel: la parte (a) ilustra una

descripción general de la configuración en la que EO y BO se cambian de conformidad con el valor de píxel; la parte (b) muestra valores específicos que deben conmutarse; y la parte (c) ilustra el contenido de una memoria de lista que se memoriza en la sección de memorización de información de desplazamiento 621.

Tal como se muestra en la parte (a) de la Figura 32, en esta forma de realización, BO se utiliza cuando el valor de píxel (valor de luminancia) está en torno a 0 o 255, y EO se utiliza cuando el valor de píxel está en las demás zonas. Es decir, el tipo de SAO cambia según el valor de píxel, y cuando los valores de píxel están en torno de 0 o 255, se añade un desplazamiento de conformidad con el valor de píxel de un píxel, y cuando el valor de píxel está en las demás zonas, se añade un desplazamiento de conformidad con el tipo de borde.

Se ha encontrado a través de los experimentos de los inventores que existen márgenes particulares de valores de píxel para ser utilizados tanto para EO como para BO. Es decir, EO se utiliza con más frecuencia en la zona de tono intermedio. Esto significa que, en la zona de valor de píxel inferior y en la zona de valor de píxel superior, los valores de píxel influyen en los errores en mayor medida que el tipo de borde.

En consecuencia, en esta forma de realización, EO y BO se conmutan de conformidad con el margen de valor de píxel y, en un tipo de SAO, se utiliza EO o BO que tienen un mayor efecto de corrección de errores, haciendo posible, de este modo, mejorar la eficiencia de codificación. Además, puesto que es posible disminuir el número de tipos de SAO y el número total de clases, se puede reducir el espacio de memoria que se va a usar y, además, se puede reducir la carga de procesamiento.

Esta forma de realización se describirá a continuación con mayor detalle. A diferencia de las formas de realización descritas con anterioridad, en esta forma de realización, los siguientes cuatro tipos de SAO, "sao_type_idx"=1 a "sao_type_idx"=4, se utilizan y definen como sigue.

"sao_type_idx"=1: (BO_EO_0), que corresponde a EO_0 + BO en la primera forma de realización

"sao_type_idx"=2: (BO_EO_1), que corresponde a EO_1 + BO en la primera forma de realización

"sao_type_idx"=3: (BO_EO_2), que corresponde a EO_2 + BO en la primera forma de realización

"sao_type_idx"=4: (BO_EO_3), que corresponde a EO_3 + BO en la primera forma de realización

No se utilizan EO_0 a 3 y BO_0, 1 en la primera forma de realización.

A continuación, tal como se ilustra en la parte (b) de la Figura 32, la sección de clasificación 624 realiza la clasificación utilizando BoTbl [BO_1] en la tabla de transformación 1301, ilustrada en la Figura 29. Cuando el valor de píxel es 1/4 de max o menor, o es 3/4 de max o mayor, la sección de clasificación 624 realiza la clasificación utilizando la tabla de transformación 1301.

Cuando el valor de píxel está entre 1/4 y 3/4 de max, la sección de clasificación 624 determina el tipo de borde de conformidad con EO_0 a 3, que se examinó en las formas de realización anteriormente descritas, y realiza la clasificación.

Con esta operación, la sección de clasificación 624 es capaz de realizar la clasificación cambiando entre BO y EO de conformidad con el valor de píxel.

El calculador de desplazamiento 811 calcula desplazamientos de una manera similar al método de cálculo de desplazamiento que se examina en las formas de realización descritas anteriormente. Sin embargo, en esta forma de realización, el número de modos de tipos de desplazamiento es cuatro, y el calculador de desplazamiento 811 calcula los desplazamientos cambiando el número de modos de tipos de desplazamiento a cuatro.

En una memoria de lista 2101, memorizada en la sección de memorización de información de desplazamiento 621, tal como se ilustra en la parte (c) de la Figura 32, el índice de QAOU, el tipo de desplazamiento y los valores específicos de los desplazamientos, con respecto a clases que se pueden seleccionar para este tipo de desplazamiento, se memorizan asociados entre sí.

Cuando se utiliza un tipo de desplazamiento que tiene características de EO y BO, la clasificación se puede realizar primero utilizando EO y, luego, para algunas clases (clases que tienen un borde plano), la clasificación se puede realizar, además, mediante el uso de BO, o se pueden utilizar tanto EO como BO, tal como se explica en esta forma de realización. Como alternativa, se puede determinar si se proporciona, o no, dependiendo de la condición examinada en las formas de realización descritas anteriormente (dependiendo de si la profundidad jerárquica de una QAOU es más pequeña que un umbral) un tipo de desplazamiento que tenga características de EO y BO. En este caso, se puede reducir el número de tipos y el número de clases.

Quinta forma de realización

A continuación, se describe otra forma de realización de la presente invención con referencia a las Figuras 33, 34 y 42. Una quinta forma de realización difiere de las formas de realización descritas anteriormente en que al realizar la clasificación utilizando EO, los pixels utilizados para determinar el tipo de borde están restringidos a pixels situados en la dirección horizontal de un píxel objeto.

En las formas de realización descritas anteriormente, un píxel situado en la dirección hacia arriba o en sentido descendente de un píxel objeto se utiliza, además, para determinar el tipo de borde. En consecuencia, para un píxel (píxel sometido a procesamiento de desplazamiento) situado en la dirección hacia arriba del píxel objeto, es necesario una memoria intermedia de línea para memorizar el valor de píxel de este píxel que no ha sido sometido al procesamiento de desplazamiento.

En consecuencia, al limitar los pixels utilizados para determinar el tipo de borde a los pixels situados en la dirección horizontal de un píxel objeto, no se hace referencia a pixels en la dirección hacia arriba, lo que hace posible reducir un espacio de memoria en una cantidad igual a una memoria intermedia de línea. Además, no es necesario el procesamiento para determinar un límite en la dirección hacia arriba (borde de la pantalla), por lo que es posible reducir la cantidad de procesamiento.

Además, si se utilizan pixels situados solamente en la dirección horizontal, se pueden utilizar los resultados de procesamiento que se han obtenido hasta ahora, con lo que resulta posible reducir, aún más, la cantidad de procesamiento. Lo que antecede se discutirá más adelante con referencia a la parte (a) de la Figura 33. La parte (a) de la Figura 33 ilustra pixels situados en la dirección horizontal y, x0 a x3 indican valores de píxel de los pixels.

Con respecto al píxel que tiene el valor de píxel x1, la diferencia entre este píxel y los dos pixels adyacentes es la siguiente.

$$s1 = \text{sign}(x1-x0) - \text{sign}(x2-x1)$$

Entonces, la diferencia entre el píxel que tiene el valor de píxel x2 y los dos pixels adyacentes es la siguiente.

$$s2 = \text{sign}(x2-x1) - \text{sign}(x3-x2)$$

En las expresiones anteriormente descritas, s1 y s2 son valores utilizados para clasificar un borde.

De este modo, cuando se calcula s2, se reutiliza sign (x2-x1), que se utiliza para calcular s1. En consecuencia, se puede reducir la cantidad de procesamiento se puede reducir en la cantidad de esta reutilización.

Los detalles de esta forma de realización se discutirán a continuación. En esta forma de realización, el tipo de EO está restringido a dos modos. Es decir, "sao_type_idx"=1, 2 se utilizan como desplazamientos de borde (EO). Cuando "sao_type_idx"=1 (EO_0), el tipo de borde se determina comparando un píxel objeto con pixels situados inmediatamente en los lados derecho e izquierdo del píxel objeto (parte (b) de la Figura 33). Cuando "sao_type_idx"=2 (EO_0), el tipo de borde se determina comparando un píxel objeto con pixels ubicados a dos pixels del píxel objeto en los lados derecho e izquierdo (parte (c) de la Figura 33).

Cuando "sao_type_idx"=3, 4, se utiliza un desplazamiento de banda (BO). En este caso, "sao_type_idx"=5, 6 en la primera forma de realización, se puede leer como "sao_type_idx"=3, 4, respectivamente.

La finalidad de determinar el tipo de borde utilizando pixels situados a dos pixels de un píxel objeto cuando "sao_type_idx"=2, es facilitar la detección de un borde que tiene un ángulo pequeño con respecto a la dirección horizontal. Tal como se ilustra en la parte (d) de la Figura 33, si dos pixels inmediatamente próximos a un píxel objeto se utilizan para la comparación, la diferencia entre el valor de píxel, del píxel objeto, y los de los pixels adyacentes es pequeña y, no se puede considerar como un borde. Incluso en este caso, si el píxel objeto se compara también con pixels situados a dos pixels del píxel objeto, se puede detectar un borde que tiene un ángulo pequeño con respecto a la dirección horizontal.

En lugar de utilizar dos pixels inmediatamente próximos a un píxel objeto, un píxel ubicado inmediatamente en el lado izquierdo de un píxel objeto, y un píxel ubicado a dos pixels del píxel objeto en el lado derecho (parte (a) de la Figura 34), se pueden utilizar para determinar la diferencia. Por el contrario, un píxel situado inmediatamente en el lado derecho del píxel objeto, y un píxel situado a dos pixels de distancia del píxel objeto en el lado izquierdo (parte (b) de la Figura 34), se pueden utilizar para determinar la diferencia. Estos casos son particularmente eficaces cuando los pixels de referencia están situados en el borde de una pantalla.

En algunos casos, puede estar restringido el uso de pixels ubicados solamente en la dirección horizontal. A modo de ejemplo, en el caso de imágenes que influyen menos en otras imágenes, tales como imágenes B e imágenes que no son de referencia, solamente se puede hacer referencia a pixels en la dirección horizontal. En el caso de otras

imágenes, se puede hacer referencia a pixels de una manera similar a la primera forma de realización. Como alternativa, en un borde de una pantalla, o cerca de un límite de un corte, se puede hacer referencia solamente a pixels en la dirección horizontal, y en las otras zonas, se puede hacer referencia a pixels de una manera similar a la primera forma de realización. Con esta disposición, se puede disminuir un espacio de memoria que ha de utilizarse y, además, se puede reducir la cantidad de procesamiento. Además, puesto que no se hace referencia en la dirección hacia arriba, la cantidad de procesamiento de determinación de límite, para la determinación de límites, tal como un borde de una pantalla o un límite de un corte, también puede reducirse.

Los pixels que se utilizarán pueden especificarse, de forma explícita, mediante el uso de un indicador para cada imagen o cada bloque.

A modo de ejemplo, si existen numerosos bordes horizontales (parte (c) de la Figura 34), el procesamiento se realiza utilizando el método dado a conocer en las formas de realización descritas anteriormente, suprimiendo así una disminución en el rendimiento.

Como alternativa, se puede determinar, dependiendo de la condición discutida en las formas de realización anteriormente descritas, (dependiendo de si la profundidad jerárquica de una QAOU es menor que un umbral), si se utilizarán, o no, pixels situados solamente en la dirección horizontal. Además, solamente cuando se utilizan tanto BO como EO, según se explica en la forma de realización descrita anteriormente, solamente se pueden utilizar pixels situados en la dirección horizontal para EO.

El tipo de borde se puede determinar dependiendo de si la diferencia entre un píxel objeto y un píxel de referencia, situado en la dirección horizontal del píxel objeto, es mayor (o menor) que un umbral. Más concretamente, se puede realizar la clasificación de conformidad con las siguientes ecuaciones.

$$\text{Sign}(z) = +1 \quad (\text{cuando } z > \text{th})$$

$$\text{Sign}(z) = 0 \quad (\text{cuando } -\text{th} \leq z \leq \text{th})$$

$$\text{Sign}(z) = -1 \quad (\text{cuando } z < -\text{th})$$

En estas ecuaciones, th indica un umbral que tiene un valor predeterminado.

En el caso de un valor de píxel que indica un componente de crominancia, solamente se pueden utilizar los pixels situados en la dirección horizontal para un desplazamiento del borde.

El efecto obtenido mediante la restricción de pixels a los de la dirección horizontal, se puede conseguir incluso cuando solamente se proporciona un método de clasificación de borde horizontal, a modo de ejemplo, incluso cuando se usa el método ilustrado en la parte (b) de la Figura 33. Si, tal como se describió anteriormente, se combina una pluralidad de métodos de clasificación de borde horizontal diferentes, se puede poner en práctica una clasificación de borde más precisa que cuando solamente se utiliza el método ilustrado en la parte (b) de la Figura 33. Una combinación de una pluralidad de métodos de clasificación de borde horizontal no está restringida al uso de dos ejemplos mediante el cambio de la distancia entre un píxel objeto y los pixels de referencia, como en las partes (b) y (c) de la Figura 33, pero se pueden utilizar dos umbrales para determinar el tipo de borde. A modo de ejemplo, se pueden utilizar dos métodos de clasificación de borde horizontal, es decir, un método cuando el umbral th es 0, y otro método cuando el umbral th es 1.

La configuración en la que el uso de pixels situados solamente en la dirección horizontal está restringido para algunos casos, se puede poner en práctica mediante la configuración del medio ilustrado en las Figuras 41 a 43 de la primera forma de realización, si las operaciones de la sección de decodificación de información de desplazamiento 611, y la sección de clasificación 624, se cambian, de forma adaptativa, de conformidad con una condición de parámetro. A modo de ejemplo, en la sección de decodificación de información de desplazamiento 611, que se ilustra en la Figura 41, los modos de tipos de desplazamiento que se seleccionarán en el selector de tipo de desplazamiento 6112, de conformidad con una condición de parámetro, están restringidos a solamente los métodos de clasificación de borde horizontal, y los modos de los tipos de desplazamiento se decodifican en el decodificador de tipo de desplazamiento 6113.

Según se ilustra en la parte (b) de la Figura 42, el selector de tabla de transformación de tipo de desplazamiento 6117' selecciona la tabla de transformación memorizada en la sección de memorización de tabla de transformación de tipo de desplazamiento de borde y tipo de desplazamiento banda 6118', o la tabla de transformación memorizada en la sección de memorización de tabla de transformación de tipo de desplazamiento de borde horizontal y tipo de desplazamiento de banda 6119', de conformidad con la condición de parámetro. En un ejemplo, si el componente de color es un componente de luminancia, el selector de tabla de transformación de tipo de desplazamiento 6117 selecciona la tabla de transformación memorizada en la sección de memorización de tabla de transformación de tipo de desplazamiento de borde y tipo de desplazamiento banda 6118'. Si el componente de color es un componente de

crominancia, el selector de tabla de transformación de tipo de desplazamiento 6117 selecciona la tabla de transformación que se memoriza en la sección de memorización de tabla de transformación de tipo de desplazamiento de borde horizontal y tipo de desplazamiento de banda 6119'.

- 5 El selector de tipo de desplazamiento 6112 selecciona un método de clasificación de borde que está restringido a un método de clasificación de borde horizontal de conformidad con la condición de parámetro y, en la sección de clasificación 624, la sección de clasificación de desplazamiento de borde 6243 clasifica los pixels. En un ejemplo, si el componente de color es un componente de luminancia, se selecciona el método de clasificación de bordes básico, y si el componente de color es un componente de crominancia, se selecciona un método de clasificación de borde horizontal.

Sexta forma de realización

- 15 Otra forma de realización de la presente invención se describirá, a continuación, haciendo referencia a las Figuras 35 a 37. Una sexta forma de realización es diferente de las formas de realización anteriormente descritas por cuanto que se mejora la precisión de desplazamiento, o se realiza una clasificación más precisa en torno al centro de los valores de píxel de componentes de crominancia.

- 20 Una imagen tiene características en las que los errores en torno a un color acromático son notables en términos de la calidad de imagen subjetiva. En los valores de píxel de cada componente de crominancia, un color acromático está situado en el centro del margen de valor, y si la profundidad de bits es de ocho bits, el color acromático es un color que tiene un valor de píxel de 128. En consecuencia, en esta forma de realización, en el caso de BO, en torno al valor de píxel de 128 de un componente de crominancia, se mejora la precisión de desplazamiento o se realiza una clasificación más precisa utilizando una anchura de clase más pequeño. Con respecto a los valores de píxel de un componente de crominancia, el margen de valor entorno a un color acromático se denomina como un "margen de valor de color acromático".

- 30 Con esta disposición, puesto que la precisión de desplazamiento se puede mejorar en el margen de valor de color acromático, se puede mejorar el rendimiento en la realización de correcciones mediante el uso de desplazamientos en el margen de valor de color acromático. En consecuencia, se reducen los errores causados por la desviación desde la imagen original, en el margen de valor de color acromático, lo que hace posible mejorar la calidad de imagen subjetiva. Además, al realizar una clasificación más precisa utilizando una anchura de clase más pequeño, en el margen de valor de color acromático, que, en otros márgenes de valor, se pueden realizar, con mayor precisión, las correcciones que utilizan desplazamientos en el margen de valor de color acromático. En consecuencia, se reducen los errores causados por la desviación procedente de la imagen original, en el margen de valor de color acromático, lo que hace posible mejorar la calidad de imagen subjetiva.

- 40 En esta forma de realización, se mejora la precisión de desplazamiento, o se realiza una clasificación más precisa para el margen de valor de color acromático. Sin embargo, el margen de valor sometido a este procesamiento no está restringido al margen de valor de color acromático. Se puede realizar una clasificación más precisa, o se puede mejorar la precisión del desplazamiento, para un margen de valor que probablemente influya en la calidad de imagen subjetiva, lo que hace posible mejorar la calidad de imagen subjetiva. Además, para el valor de píxel de un componente de luminancia, así como el de un componente de crominancia, para un margen de valor en el que es probable que la degradación de la calidad de imagen sea importante en términos de calidad de imagen subjetiva, la precisión de desplazamiento puede mejorar, de forma similar, o se puede realizar, de forma similar una clasificación más precisa, logrando así ventajas similares a las obtenidas para un componente de crominancia.

- 50 A continuación, se describirá en primer lugar un caso en el que se mejora la precisión de desplazamiento haciendo referencia a la Figura 35. La Figura 35 ilustra una visión general de un caso en el que se mejora la precisión de desplazamiento. Con el fin de mejorar la precisión de desplazamiento, tal como se ilustra en la Figura 35, la precisión de desplazamiento en torno al valor de píxel de 128 de un componente de crominancia, se establece para ser alta, y la precisión de desplazamiento en los otros márgenes de valor se establece para ser baja.

- 55 La sección de memorización de información de desplazamiento 621 asegura una zona de memorización de n_B bits (a modo de ejemplo, $n_B=8$) para desplazamientos utilizados para un margen de valor que tiene una mayor precisión de desplazamiento, y asegura una zona de memorización de n_A bits (a modo de ejemplo, $n_A=6$) para desplazamientos utilizados para un margen de valor que tiene una precisión de desplazamiento más baja. Si existe una zona de memorización suficiente, la zona de memorización se puede configurar para que sea, de forma incondicional, n_B bits, en cuyo caso, se facilita la puesta en práctica.

- 60 La sección de recorte de desplazamiento 812 utiliza diferentes márgenes de recorte para un margen de valor que tiene una alta precisión de desplazamiento, y un margen de valor que tiene una baja precisión de desplazamiento. A modo de ejemplo, para un margen de valor que tiene una precisión de desplazamiento baja, el margen de recorte se establece en $-2^{n_A/2}$ a $2^{n_A/2}-1$, y para un margen de valor que tiene una precisión de desplazamiento alta, el margen de recorte se establece en $-2^{n_B/2}$ a $2^{n_B/2}-1$.

A continuación, se examina un caso en el que se realiza una clasificación más precisa haciendo referencia a la Figura 36. La Figura 36 ilustra una visión general de un caso en el que se realiza una clasificación más precisa: la parte (a) ilustra una tabla de transformación; y las partes (b) a (d) muestran que están divididos valores de píxel.

5 Cuando se realiza una clasificación más precisa, en una memoria de lista, para un componente de crominancia, en la sección de memorización de información de desplazamiento 621, se asegura una zona de memorización de desplazamiento correspondiente a "sao_type_idx"=5, de conformidad con el número de niveles de clasificación.

10 En el caso de BO, la sección de clasificación 624 considera si el valor de píxel de un píxel objeto está incluido, o no, dentro del margen de valor de color acromático cuando se determina una clase desde del valor de píxel. Más concretamente, la sección de clasificación 624 realiza la clasificación utilizando uno de los patrones en una tabla de transformación 2501 que se ilustra en la parte (a) de la Figura 36.

15 En la tabla de transformación 2501, la clasificación se realiza utilizando "class_idx=BoTbl [sao_type_idx] [pix[x]/4]", y las clases se asignan a anchuras más pequeñas de valores de píxel que los de las tablas de transformación 1301 y 1302, que se ilustran en la Figura 29.

"BoTbl [BO_0] [pix/4] (a)" en la tabla de transformación 2501, ilustra un ejemplo (parte (b) de la Figura 36) en el que existen dieciséis clases y, el margen de valor de color acromático se divide en anchuras más pequeñas y las zonas de tono intermedio (pix/4=16 a 21, 42 a 47) se dividen en anchuras más grandes. "BoTbl [BO_0] [pix/4] (b)" ilustra un ejemplo (parte (c) de la Figura 36) en el que existen dieciséis clases y, el margen en el que se añaden los desplazamientos se reduce, y el margen de valor del color acromático se divide en anchuras más pequeñas, mientras que el número máximo de clases, y la anchura de clase máximo se mantienen. "BoTbl [BO_0] [pix/4] (c)" muestra un ejemplo (parte (d) de la Figura 36) en el que existen dieciocho clases, y el margen de valor del color acromático se divide en anchuras más pequeñas mientras que se mantiene la anchura de clase máximo. Además, en otro ejemplo, la anchura de clase de las zonas de tono intermedio se establece para ser relativamente mayor que el margen de valor de color acromático disminuyendo, de este modo, el número de clases completas. En este ejemplo, es posible reducir el espacio de la zona de memorización de desplazamiento al mismo tiempo que se mantiene la precisión de corrección en el margen de valor de color acromático. Dicha disposición es posible debido a que la frecuencia con la que se utilizan los desplazamientos para la corrección en las zonas de tono intermedio es menor que en el margen de valor de color acromático.

35 Como alternativa, se puede realizar tanto una mejora en la precisión de desplazamiento como una clasificación más precisa en el margen de valor de color acromático.

De forma alternativa, la precisión de desplazamiento se puede cambiar en función de si el valor de píxel indica un componente de luminancia o un componente de crominancia. Es decir, la precisión de desplazamiento para el valor de píxel, que indica un componente de crominancia, se puede establecer para que sea inferior a la del valor de píxel que indica un componente de luminancia.

40 Apéndice 7

La adición de un desplazamiento se puede realizar de modo que el valor de píxel se aproxime a un color acromático (= valor de píxel de 128 de un componente de crominancia). Es decir, el sumador de desplazamiento 626 puede gestionar desplazamientos de modo que el signo de un desplazamiento que se añade al valor de píxel de 128, o mayor, sea opuesto al signo de un desplazamiento que ha de añadirse al valor de píxel menor de 128. A modo de ejemplo, cuando un desplazamiento se indica con a, el signo del desplazamiento se puede determinar de modo que $x' = x + a$ ($x < 128$) y $x' = x - a$ ($x \geq 128$).

50 Si 128 está situado entre el valor de píxel al que no se ha añadido un desplazamiento, y el valor de píxel al que se ha añadido un desplazamiento, el valor de píxel al que se ha añadido un desplazamiento se puede recortar por 128. A modo de ejemplo, si el valor de píxel es "126 (azul)" y el desplazamiento es "3", en lugar de añadir, directamente, el desplazamiento al valor de píxel, de modo que $x = 126 + 3 = 129$, x se puede establecer en $x = \text{clip}(126 + 3, 0, 128) = 128$. En esta ecuación, clip (x, y, z) indica el procesamiento para restringir el valor de x a $y \leq x \leq z$.

55 Si el valor de píxel es "129 (rojo)" y el desplazamiento es "-4", en lugar de añadir, directamente, el desplazamiento al valor de píxel de modo que $x = 129 + (-4) = 125$, se puede establecer x a $x = \text{clip}(129 + (-4), 128, 255) = 128$.

60 (Apéndice 8)

Para un componente de crominancia, se pueden añadir desplazamientos en una de las zonas de tono intermedio, y las otras zonas. A modo de ejemplo, tal como se ilustra en la parte (a) de la Figura 37, la clasificación se puede realizar mediante el enfoque en los valores de píxel de un color acromático. En el ejemplo ilustrado en la parte (a) de la Figura 37, en el caso de "sao_type_idx"=5 (BO_0), se asignan clases a las zonas de tono intermedio, y en el caso de "sao_type_idx"=6 (BO_1), se asignan clases a márgenes de valor distintos de las zonas de tono intermedio.

Como alternativa, tal como se ilustra en la parte (b) de la Figura 37, la clasificación se puede realizar de forma asimétrica en dos márgenes de valor con respecto a los valores de píxel de un color acromático. En el ejemplo ilustrado en la parte (b) de la Figura 37, la anchura de clase es el mismo que el de la parte (a) de la Figura 37, y las posiciones de partición de las clases son distintas de las de la parte (a) de la Figura 37.

De forma alternativa, tal como se ilustra en las partes (c) y (d) de la Figura 37, se puede realizar la clasificación de forma diferente de conformidad con el canal de crominancia (Cr o Cb). En los ejemplos mostrados en las partes (c) y (d) de la Figura 37, la anchura de clase es diferente entre el componente de crominancia (Cr), y el componente de crominancia (Cb). La anchura de clase se puede cambiar estableciendo niveles de clasificación más precisos, como en la tabla de transformación 2501.

Tal como se ilustra en la parte (e) de la Figura 37, para los componentes de crominancia, se puede realizar la clasificación utilizando solamente un tipo de BO.

Séptima forma de realización

Otra forma de realización de la presente invención se describirá, a continuación, haciendo referencia a las Figuras 38 a 40. Una séptima forma de realización difiere de las formas de realización descritas anteriormente en cuanto que, en lugar de utilizar el valor de un desplazamiento por sí mismo, el valor de un desplazamiento se somete a codificación predictiva, es decir, se codifica un residuo de desplazamiento calculado utilizando el valor de un desplazamiento, y un valor de predicción del valor de desplazamiento, y en el que "0" se puede proporcionar como el valor de predicción de un desplazamiento.

En condiciones normales, en SAO, se codifica un desplazamiento de conformidad con una clase que indica una zona clasificada. Si no existe un píxel clasificado en una clase determinada (la clase está vacía), se codifica "0". En general, desplazamientos en clases adyacentes, o zonas adyacentes, tienen valores similares. En consecuencia, si un valor de desplazamiento que ya se ha decodificado se utiliza como valor de predicción de otro desplazamiento, y la diferencia de este valor de predicción se codifica, la cantidad de datos para codificar los desplazamientos se puede reducir. Sin embargo, si la clase está vacía, aumenta la cantidad de datos para codificar los desplazamientos.

En esta forma de realización, por lo tanto, "0" se incluye como valores de predicción de desplazamientos, y se selecciona un valor de predicción desde de una pluralidad de candidatos de valores de predicción que incluyen "0", y a continuación, se codifica la diferencia de predicción, con lo que se reduce la cantidad de datos para codificar desplazamientos.

Con esta disposición, independientemente de si una clase está vacía, o si existe un píxel clasificado en esta clase y se codifica un desplazamiento, se puede asignar un valor de predicción adecuado, lo que hace posible reducir la cantidad de datos para codificar los desplazamientos mediante la realización de codificación predictiva.

La configuración de una sección de decodificación de información de desplazamiento 611', de esta forma de realización, se describirá a continuación con referencia a la Figura 38. En lugar de la sección de decodificación de información de desplazamiento 611' se proporciona la sección de decodificación de información de desplazamiento 611 ilustrada en la Figura 22, e incluye un decodificador de residuo de desplazamiento 651, una sección de reconstrucción de desplazamiento 652, y una sección de determinación de valor de predicción 653. La sección de determinación de valor de predicción 653 incluye un decodificador de indicador de candidato de predicción 661, un calculador de valor de predicción fijo 662, un calculador de valor de predicción codificado 663, y un selector de candidato de valor de predicción 664.

El decodificador de residuo de desplazamiento 651 decodifica un residuo de desplazamiento desde de información de QAOU, incluida en los datos codificados nº 1, y proporciona el residuo de desplazamiento decodificado a la sección de reconstrucción de desplazamiento 652.

La sección de determinación de valor de predicción 653 determina un valor de predicción de un desplazamiento. El decodificador de indicador de candidato de predicción 661 decodifica un indicador de candidato de predicción desde de la información de QAOU, y proporciona el indicador de candidato de predicción decodificado al selector de candidato de valor de predicción 664. El decodificador de indicador de candidato de predicción 661 se puede configurar de tal modo que no decodifique un indicador de sección candidato si un valor de desplazamiento decodificado previamente es "0".

El calculador de valor de predicción fijo 662 calcula un valor fijo (en este caso, "0"), que es un desplazamiento que ha de codificarse cuando no existe un píxel clasificado en una determinada clase, como un valor de predicción, y proporciona el valor fijo calculado para el selector de candidato de valor de predicción 664.

El calculador de valor de predicción codificado 663 efectúa la lectura de un desplazamiento decodificado desde de la sección de memorización de información de desplazamiento 621, y calcula un valor de predicción y lo proporciona al selector de candidato de valor de predicción 664.

El selector de candidato de valor de predicción 664 selecciona, de conformidad con el indicador de candidato de predicción, un valor de predicción ha de proporcionarse a la sección de reconstrucción de desplazamiento 652 entre el valor de predicción proporcionado desde el calculador de valor de predicción fijo 662, y el valor de predicción suministrado desde del calculador de valor de predicción codificado 663.

La sección de reconstrucción de desplazamiento 652 reconstruye un desplazamiento (Offset) desde del valor de predicción (pred) que se proporciona desde del selector de candidato de valor de predicción 664, y desde el residuo de desplazamiento (sao_offset) suministrado desde del decodificador de residuo de desplazamiento 651 de conformidad con la siguiente ecuación.

$$\text{"Offset"} = \text{"pred"} + \text{"sao_offset"}$$

La razón por la cual aumenta la cantidad de datos requeridos para codificación aumenta si existe una clase (clase vacía) bajo la cual no se clasifica ningún píxel, se describirá, a continuación, haciendo referencia a la Figura 39. La Figura 39 ilustra una visión general de un caso en el que existen clases vacías. Tal como se muestra en la Figura 39, en una clase que no está vacía, se codifica un desplazamiento utilizado para esta clase. A continuación, para cada clase, utilizando el desplazamiento anterior como un valor de predicción, se codifica la diferencia entre el valor de predicción y el desplazamiento actual. Sin embargo, si existe una clase vacía, el valor de predicción se convierte en "0", y el valor de diferencia aumenta. Como resultado, aumenta la cantidad de datos necesarios para la codificación.

En esta forma de realización, por lo tanto, mediante el uso de un indicador de candidato de predicción, el valor de predicción se puede seleccionar de entre "0" y "desplazamiento anterior (distinto de 0)".

La sintaxis que ha de utilizarse cuando se utiliza el indicador de candidato de predicción, se describirá a continuación, con referencia a la Figura 40. La Figura 40 ilustra la sintaxis 2901 que se usará cuando se use un indicador de candidato de predicción. Según se ilustra en la Figura 40, se añade un indicador de candidato de predicción "sao_pred_flag", y si "sao_pred_flag"=0, el desplazamiento decodificado previamente (distinto de 0) se establece como un valor de predicción offsetp. Si "sao_pred_flag"=1, o el desplazamiento actual es el primer desplazamiento, el valor de predicción offsetp se establece en 0. En la sintaxis 2901, "sao_offset_delta" representa una diferencia d, desde del valor de predicción y,

$$\text{"sao_offset[sao_curr_depth][ys][xs]"} = \text{"offsetp+d"}.$$

El indicador de candidato de predicción no se puede utilizar dependiendo del tipo de desplazamiento. A modo de ejemplo, en el caso de EO, no se utiliza el indicador de candidato de predicción y, en el caso de BO, se utiliza el indicador de candidato de predicción. Lo que antecede se debe a que, en BO, clases vacías se pueden generar, con frecuencia, dependiendo de la distribución de los valores de píxel.

Además, se puede realizar una determinación de si efectuar la codificación predictiva en sí misma, o no, dependiendo del tipo de desplazamiento. A modo de ejemplo, en el caso de EO, se puede realizar una codificación predictiva, y en el caso de BO, no se puede realizar una codificación predictiva.

(Apéndice 9)

En las tercera a séptima formas de realización descritas con anterioridad, mediante el uso del modo intra-predicción, las clases de desplazamiento de borde se pueden estimar, restringir y reorganizar para cada LCU.

Una dirección de borde seleccionada como un tipo de EO de SAO puede tener correlación con un modo intra-predicción. En consecuencia, la referencia al modo de intra-predicción de una CU situada en una posición correspondiente a una QAOU de SAO se puede utilizar para seleccionar una clase de EO.

A modo de ejemplo, utilizando el modo de intra-predicción, se puede estimar y determinar una clase de EO, los candidatos de clase de EO pueden restringirse, o el orden (índice) de clases de EO se puede reorganizar en el orden en el que es más probable que se seleccione.

Lo que antecede es particularmente fácil de poner en práctica si una imagen se divide de forma que el tamaño de una QAOU de cada nivel jerárquico de SAO sea el mismo que el de una CU, a modo de ejemplo, de forma que el tamaño máximo de una QAOU de SAO sea igual al tamaño de una LCU.

(Apéndice 10)

En un filtro de desplazamiento adaptativo conocido, puesto que se proporcionan numerosos modos de desplazamientos (tipos y clases), puede ser necesario un gran tamaño de memoria. Por consiguiente, en esta forma

de realización, a modo de ejemplo, se pone en práctica un dispositivo de filtrado de imagen que es capaz de reducir la distorsión de bloque mientras se suprime un aumento en el tamaño de la memoria.

Un dispositivo de filtrado de imagen, según un ejemplo útil para comprender la invención, es un dispositivo de filtrado de imagen para añadir un desplazamiento, seleccionado de entre una pluralidad de desplazamientos, a un valor de píxel de cada píxel que forma una imagen de entrada que está constituida por una pluralidad de zonas unitarias. El dispositivo de filtrado de imagen incluye: un medio de determinación de desplazamiento para determinar, para cada zona unitaria, un desplazamiento que se agregará al valor de píxel de cada píxel incluido en la zona unitaria; y un medio de filtrado para añadir un desplazamiento determinado por el medio de determinación de desplazamiento al valor de píxel de cada píxel incluido en la zona unitaria. En un caso en el que el tamaño de una zona unitaria, para la cual se determina un desplazamiento es más pequeño que un tamaño predeterminado, el medio de determinación de desplazamiento determina un desplazamiento que ha de añadirse al valor de píxel de cada píxel incluido en la zona unitaria de entre un número más restringido de desplazamientos seleccionables que en un caso en el que el tamaño de la zona unitaria es igual o mayor que el tamaño predeterminado.

Con la configuración descrita anteriormente, en un caso en el que el tamaño de una zona unitaria, para la cual se determina un desplazamiento, es menor que el tamaño predeterminado, se determina un desplazamiento que ha de añadirse entre un número más restringido de desplazamientos que en un caso en el que el tamaño de la zona unitaria es igual, o mayor, que el tamaño predeterminado.

Cuando el tamaño de una zona unitaria es pequeño, el número de píxels incluidos en esta zona unitaria también es pequeño, y es más probable que los píxels en esta zona unitaria tengan valores similares. En consecuencia, cuando el tamaño de una zona unitaria es pequeño, incluso si el número de desplazamientos seleccionables está restringido, la influencia en una imagen a la que se aplica un desplazamiento es pequeña. Además, al restringir el número de desplazamientos seleccionables, se puede reducir el espacio de memoria requerido.

Por lo tanto, con la configuración anteriormente descrita, es posible reducir un espacio de memoria que ha de utilizarse al mismo tiempo que la influencia en una imagen a la que se aplica un desplazamiento disminuye. Además, puesto que el número de desplazamientos seleccionables disminuye, la cantidad de datos para codificar desplazamientos se puede reducir, con lo que se mejora la eficiencia de codificación.

Con el fin de resolver el problema descrito anteriormente, un dispositivo de filtrado de imagen, de conformidad con la presente invención, es un dispositivo de filtrado de imagen para añadir un desplazamiento a un valor de píxel de cada píxel que forma una imagen de entrada que está constituida por una pluralidad de zonas unitarias. El dispositivo de filtrado de imagen incluye: un medio de determinación de desplazamiento para determinar, para cada zona unitaria, un desplazamiento que ha de añadirse al valor de píxel de cada píxel incluido en la zona unitaria; y un medio de filtrado para añadir un desplazamiento determinado por el medio de determinación de desplazamiento, al valor de píxel de cada píxel incluido en la zona unitaria. En un caso en el que el tamaño de una zona unitaria para la cual se determina un desplazamiento es más pequeño que un tamaño predeterminado, el medio de filtrado añade un desplazamiento que tiene una precisión menor que en un caso en el que el tamaño de la zona unitaria es igual a o mayor que el tamaño predeterminado.

Con la configuración anteriormente descrita, en un caso en el que el tamaño de una zona unitaria para la cual se determina un desplazamiento es más pequeño que el tamaño predeterminado, un desplazamiento que tiene una precisión menor que la de un caso en el que el tamaño de la zona unitaria es igual o mayor que el tamaño predeterminado que se añade.

Cuando el tamaño de una zona unitaria es pequeño, el número de píxels incluidos en esta zona unitaria también es pequeño y, es más probable que los píxels en esta zona unitaria tengan valores similares. En consecuencia, cuando el tamaño de una zona unitaria es pequeño, incluso si disminuye la precisión de un desplazamiento, la influencia sobre los errores de cuantización es pequeña y, por lo tanto, la influencia sobre una imagen a la que se aplica un desplazamiento también es pequeña. Además, al disminuir la precisión de un desplazamiento, se puede reducir un espacio de memoria requerido.

Por lo tanto, con la configuración descrita anteriormente, es posible reducir un espacio de memoria que se utilizará, mientras que disminuye la influencia en una imagen a la que se aplica un desplazamiento. Además, puesto que la precisión de un desplazamiento disminuye, se puede reducir la cantidad de datos para codificar, con lo que se mejora la eficacia de codificación.

Un dispositivo de filtrado de imagen, de conformidad con la presente invención, es un dispositivo de filtrado de imagen para realizar un desplazamiento adaptativo (SAO: Desplazamiento Adaptativo de Muestra) en un valor de píxel de cada píxel que forma una imagen de entrada que está constituida por una pluralidad de zonas unitarias. El dispositivo de filtrado de imagen incluye: un medio de determinación de desplazamiento para determinar, para cada zona unitaria, un tipo de desplazamiento de desplazamiento que ha de añadirse al valor de píxel de cada píxel incluido en la zona unitaria; y un medio de filtrado para añadir un desplazamiento de un tipo de desplazamiento determinado por el medio de determinación de desplazamiento, al valor de píxel de cada píxel incluido en la zona

unitaria. El medio de determinación de desplazamiento determina que el tipo de desplazamiento sea un desplazamiento de banda (BO) en un caso en el que el valor de píxel, de un píxel objeto, al que se añade un desplazamiento está situado en un margen de valor en torno a un valor máximo, o un valor mínimo, y determina que el tipo de desplazamiento sea un desplazamiento de borde (EO) en un caso en el que el valor de píxel del píxel objeto está situado en un margen de valor distinto del margen de valor en torno al valor máximo y el valor mínimo.

Con la configuración descrita anteriormente, cuando se realiza un desplazamiento adaptativo, se aplica un desplazamiento de banda a píxels que tienen valores de píxel situados en torno al valor máximo o el valor mínimo, y se aplica un desplazamiento de borde a píxels que tienen valores de píxel situados en márgenes de valor distintos que los márgenes de valor del valor máximo, y el valor mínimo.

En un margen de valor de píxel más alto, o en un margen de valor de píxel más bajo, es más probable que los valores de píxel influyan en los errores que lo que hacen los bordes. En consecuencia, con la configuración descrita anteriormente, se puede mejorar la eficacia en la corrección de errores, con lo que se mejora la eficiencia de codificación.

Si un tipo de desplazamiento incluye tanto un desplazamiento de banda como un desplazamiento de borde, el número de tipos de desplazamientos adaptativos se puede disminuir, con lo que se reduce un espacio de memoria que ha de utilizarse y, además, se reduce la cantidad de procesamiento.

En la configuración descrita anteriormente, el procesamiento de desplazamiento de banda es un procesamiento de desplazamiento para añadir una de entre una pluralidad de desplazamientos, al valor de píxel de un píxel objeto de conformidad con la magnitud del valor de píxel del píxel objeto (esto se aplica, además, a la descripción posterior). El procesamiento de desplazamiento de borde es un procesamiento de desplazamiento para añadir una, de entre una pluralidad de desplazamientos, al valor de píxel de un píxel objeto de conformidad con la diferencia entre el valor de píxel del píxel objeto, y el valor de píxel de un píxel situado en torno al píxel objeto (esto se aplica, además, a la descripción posterior).

En el dispositivo de filtrado de imagen, de conformidad con la presente invención, el medio de determinación de desplazamiento pueden determinar que el tipo de desplazamiento sea un desplazamiento de banda para píxels situados en un margen de valor desde el valor mínimo hasta $1/4$ del valor máximo, o un margen de valor desde $3/4$ del valor máximo al valor máximo y puede determinar que el tipo de desplazamiento sea un desplazamiento de borde para píxels situados en márgenes de valor distintos a estos márgenes de valor.

Con la configuración descrita anteriormente, es posible separar, claramente, los píxels a los que se aplica un desplazamiento de banda desde de píxels a los que se aplica un desplazamiento de borde.

Un dispositivo de filtrado de imagen, de conformidad con la presente invención, es un dispositivo de filtrado de imagen para realizar un desplazamiento adaptativo (SAO: Desplazamiento Adaptativo de Muestra) en una imagen de entrada. El dispositivo de filtrado de imagen incluye: un medio de clasificación para determinar un borde con el fin de determinar una clase que se utilizará en un desplazamiento de borde (EO) que hace referencia a píxels ubicados solamente en la dirección horizontal de un píxel objeto; y un medio de filtrado para añadir un desplazamiento asociado con una clase determinada por el medio de clasificación en un caso en el que se aplica un desplazamiento de borde (EO).

Con la configuración descrita con anterioridad, cuando se determina un borde, se hace referencia a píxels situados solamente en la dirección horizontal, lo que hace posible reducir el espacio de memoria requerido, en comparación con un caso en el que se hace referencia, además, píxels en la dirección hacia arriba. Además, no es necesario el procesamiento de determinación de límites para determinar límites en la dirección ascendente, lo que permite reducir la cantidad de procesamiento.

En el dispositivo de filtrado de imagen, de conformidad con la presente invención, el medio de clasificación puede determinar un borde haciendo referencia a un píxel situado a dos píxels de distancia del píxel objeto en la dirección horizontal.

Con la configuración descrita anteriormente, puesto que se hace referencia a un píxel situado a dos píxels de distancia del píxel objeto, es posible detectar incluso un borde que tiene un ángulo pequeño.

Un dispositivo de filtrado de imagen, según un ejemplo útil para comprender la invención, es un dispositivo de filtrado de imagen para realizar un desplazamiento adaptativo (SAO: Desplazamiento Adaptativo de Muestra) en una imagen de entrada. El dispositivo de filtrado de imagen incluye: un medio de clasificación para determinar una clase que se utilizará en un desplazamiento de banda (BO) estableciendo la anchura de división de clases que se han de utilizar en un margen de valores de píxel que indican la prominencia en torno de un valor central, que es un valor situado en el centro entre el valor máximo y el valor mínimo, para ser más pequeño que en los otros márgenes de los valores de píxel; y un medio de filtrado para añadir un desplazamiento asociado con una clase determinada por el medio de clasificación en un caso en el que se aplica un desplazamiento de banda (BO).

Con la configuración anteriormente descrita, la anchura de división de clases que se usarán en un margen de valores de píxel que indica la crominancia en torno al valor central, que es un valor situado en el centro entre el valor máximo y el valor mínimo, se establece para ser más pequeño que el de los otros márgenes de los valores de píxel, y a continuación, se determinan las clases que se utilizarán cuando se aplica un desplazamiento de banda (BO).

Cuando el valor de píxel que indica la crominancia es el valor central, este píxel es un color acromático. Los errores de un color acromático son perceptibles para el ojo humano, lo que disminuye la calidad de imagen subjetiva. A continuación, como en la configuración descrita anteriormente, si la anchura de clase en un margen en torno al valor central se establece para ser pequeño, se pueden establecer los desplazamientos de manera precisa para píxels en torno al valor central. Como resultado, se puede aumentar la calidad de imagen subjetiva.

En el dispositivo de filtrado de imagen, según la presente invención, si el valor central se coloca entre el valor de píxel al que no se ha añadido un desplazamiento y el valor de píxel al que se ha añadido un desplazamiento, el medio de filtrado puede establecer el valor de píxel al que se ha añadido un desplazamiento como el valor central.

Con la configuración descrita anteriormente, no se añade un desplazamiento si el valor de píxel al que se ha añadido un desplazamiento está más allá del valor central. Cuando el valor de píxel que indica la crominancia es el valor central, este píxel es un color acromático, y en márgenes de valor con el valor central entre ellos, el color percibido por el ojo humano se cambia del color acromático. Por lo tanto, con la configuración descrita anteriormente, es posible evitar que el color percibido por el ojo humano cambie, lo que sería causado, de otro modo, añadiendo un desplazamiento.

Un dispositivo de filtrado de imagen, según un ejemplo útil para comprender la invención, es un dispositivo de filtrado de imagen para realizar un desplazamiento adaptativo (SAO: Desplazamiento Adaptativo de Muestra) en una imagen de entrada. El dispositivo de filtrado de imagen incluye: un medio de filtrado para añadir un desplazamiento que tiene una mayor precisión, a un píxel que tiene un valor de píxel que indica la crominancia en torno a un valor central, que es un valor situado en el centro entre el valor máximo y el valor mínimo, que la precisión de un desplazamiento que se añadirá a un píxel en los otros márgenes de valor.

Con la configuración descrita anteriormente, se añade un desplazamiento que tiene una mayor precisión, a un píxel que tiene un valor de píxel que indica la crominancia en torno a un valor central, que es un valor situado en el centro entre el valor máximo y el valor mínimo, que la precisión de un desplazamiento que ha de añadirse a un píxel en los otros márgenes de valor.

Cuando el valor de píxel de un píxel que indica la crominancia es el valor central, este píxel es un color acromático. Los errores de un color acromático son perceptibles para el ojo humano, lo que disminuye la calidad de imagen subjetiva. Por lo tanto, como en la configuración descrita anteriormente, si se mejora la precisión de un desplazamiento que se añadirá a los píxels en torno a un valor central, se pueden añadir desplazamientos con precisión a los píxels en torno al valor central. Como resultado, se puede aumentar la calidad de imagen subjetiva.

Un dispositivo de decodificación de desplazamiento, según un ejemplo útil para comprender la invención, es un dispositivo de decodificación de desplazamiento para decodificar cada desplazamiento al que se hace referencia mediante un filtro de imagen para añadir un desplazamiento a un valor de píxel de cada píxel que forma una imagen de entrada. El dispositivo de decodificación de desplazamiento incluye: un medio de decodificación de residuo de desplazamiento para decodificar cada residuo de desplazamiento desde de datos codificados; un medio de determinación de valor de predicción para determinar un valor de predicción de cada desplazamiento desde de un desplazamiento decodificado, o un valor predeterminado; y un medio de cálculo de desplazamiento para calcular cada desplazamiento desde de un valor de predicción determinado por el medio de determinación de valor de predicción, y un residuo de desplazamiento decodificado por el medio de decodificación de residuo de desplazamiento.

Con la configuración descrita anteriormente, puesto que un desplazamiento se decodifica desde de un residuo, puede disminuirse la cantidad de datos para codificar un desplazamiento, en comparación con un caso en el que cada desplazamiento se codifica. Además, un valor de predicción para determinar un residuo se determina desde de un desplazamiento decodificado, o un valor predeterminado. Por lo tanto, es posible evitar que la cantidad de datos para codificar los datos de diferencia llegue a ser mayor que cuando se codifica un desplazamiento lo que, de otro modo, sería causado por el uso de solamente desplazamientos decodificados.

El valor predeterminado puede ser, a modo de ejemplo, "0".

Un dispositivo de codificación de desplazamiento, según un ejemplo útil para comprender la invención, es un dispositivo de codificación de desplazamiento para codificar cada desplazamiento al que se hace referencia mediante un filtro de imagen para añadir un desplazamiento a un valor de píxel de cada píxel que forma una imagen de entrada. El dispositivo de codificación de desplazamiento incluye: un medio de determinación de valor de predicción para determinar un valor de predicción de cada desplazamiento desde de un desplazamiento codificado,

o un valor predeterminado; un medio de cálculo de residuo de desplazamiento para calcular un residuo de desplazamiento desde de cada desplazamiento, y un valor de predicción determinado por el medio de determinación de valor de predicción; y un medio de codificación de residuo de desplazamiento para codificar un residuo de desplazamiento calculado por el medio de cálculo de residuo de desplazamiento.

Con la configuración anteriormente descrita, puesto que un desplazamiento se decodifica desde de un residuo, la cantidad de datos para codificar un desplazamiento puede disminuir, en comparación con un caso en el que cada desplazamiento está codificado. Además, un valor de predicción para determinar un residuo, se determina desde de un desplazamiento decodificado o un valor predeterminado. Por lo tanto, es posible evitar que la cantidad de datos para codificar los datos de diferencia llegue a ser mayor que cuando se codifica un desplazamiento, lo que se causaría, de otro modo, por el uso de solamente desplazamientos decodificados.

Una estructura de datos de datos codificados, de conformidad con la presente invención, es una estructura de datos, de datos codificados, a los que se hace referencia mediante un filtro de imagen para añadir un desplazamiento a un valor de píxel de cada píxel que forma una imagen de entrada que está constituida por una pluralidad de zonas unitarias. La estructura de datos incluye: información de determinación de valor de predicción que indica si un valor de predicción se determinará desde de un desplazamiento decodificado, o un valor predeterminado. El filtro de imagen se refiere a la información de determinación de valor de predicción que se incluye en los datos codificados, y a continuación, determina un valor de predicción y decodifica un desplazamiento.

Con la configuración anteriormente descrita, es posible determinar, utilizando la información de determinación del valor de predicción, si se establecerá un valor de predicción como un desplazamiento decodificado o un valor predeterminado.

(Ejemplos de aplicación)

El dispositivo de decodificación de vídeo 1 (1'), y el dispositivo de codificación de vídeo 2 (2') descritos con anterioridad, pueden estar montados y utilizarse para varios aparatos para la transmisión, recepción, registro y reproducción de imágenes de vídeo. Las imágenes de vídeo pueden ser imágenes de vídeo no artificiales capturadas por, a modo de ejemplo, una cámara, o imágenes de vídeo artificiales (incluyendo CG y GUI) generadas por, a modo de ejemplo, un ordenador.

El uso del dispositivo de decodificación de vídeo 1, y el dispositivo de codificación de vídeo 2, descritos anteriormente, para transmitir y recibir imágenes de vídeo se describirá a continuación haciendo referencia a la Figura 45.

La parte (a) de la Figura 45 es un diagrama de bloques que ilustra la configuración de un aparato de transmisión A en el que está montado el dispositivo de codificación de vídeo 2. Tal como se ilustra en la parte (a) de la Figura 45, el aparato de transmisión A incluye un codificador A1 que codifica imágenes de vídeo con el fin de obtener datos codificados, un modulador A2 que modula una onda portadora utilizando los datos codificados obtenidos por el codificador A1, de modo que se obtenga una señal de modulación, y un transmisor A3 que transmite la señal de modulación obtenida por el modulador A2. El dispositivo de codificación de vídeo 2 descrito anteriormente se utiliza como este codificador A1.

Como fuentes de suministro para la entrada de imágenes de vídeo en el codificador A1, el aparato de transmisión A puede incluir, además, una cámara A4 que captura imágenes de vídeo, un medio de registro A5 en el que se registran imágenes de vídeo, un terminal de entrada A6 para recibir imágenes de vídeo desde una fuente externa, y un procesador de imagen A7 que genera o procesa imágenes. Aunque la configuración del aparato de transmisión A, que incluye todos estos elementos se muestra en la parte (a) de la Figura 45, algunos elementos pueden omitirse.

El medio de registro A5 puede ser un medio de registro en el que se registran imágenes de vídeo que no están codificadas, o imágenes de vídeo que están codificadas por un método de codificación de registro, que es diferente de un método de codificación de transmisión. Si las imágenes de vídeo que están codificadas por un método de codificación de registro, se registran en el medio de registro A5, un decodificador (no ilustrado) para decodificar datos codificados objeto de lectura desde del medio de registro A5 utilizando un método de codificación de registro puede interponerse entre el medio de registro A5 y el codificador A1.

La parte (b) de la Figura 45 es un diagrama de bloques que ilustra la configuración de un aparato de recepción B en el que está montado el decodificador de vídeo 1. Tal como se ilustra en la parte (b) de la Figura 45, el aparato de recepción B incluye un receptor B1 que recibe una señal de modulación, un demodulador B2 que demodula la señal de modulación recibida por el receptor B1, con el fin de obtener datos codificados, y un decodificador B3 que decodifica los datos codificados obtenidos por el demodulador B2 de modo que se obtengan imágenes de vídeo. El dispositivo de decodificación de vídeo 1 descrito anteriormente se utiliza como este decodificador B3.

Como destinos a los que se suministran imágenes de vídeo procedentes del decodificador B3, el aparato de recepción B puede incluir, además, una pantalla B4 que muestra imágenes de vídeo, un medio de registro B5 para

grabar imágenes de vídeo, y un terminal de salida B6 para emitir vídeo Imágenes a una fuente externa. Aunque la configuración del aparato de recepción B, que incluye todos estos elementos se ilustra en la parte (b) de la Figura 45, se pueden omitir algunos elementos.

El medio de registro B5 puede ser un medio de registro para grabar imágenes de vídeo que no están codificadas, o imágenes de vídeo que están codificadas por un método de codificación de registro, que es diferente de un método de codificación de transmisión. Si el medio de registro B5 es un medio de registro para grabar imágenes de vídeo que están codificadas por un método de codificación de registro, se puede interponer un codificador (no ilustrado), para codificar imágenes de vídeo obtenidas del decodificador B3 utilizando un método de codificación de registro, entre el decodificador B3 y el medio de registro B5.

Un medio de transmisión para transmitir una señal de modulación puede ser un medio inalámbrico o un medio cableado. Un modo de transmisión en el que se transmite una señal de modulación puede ser la difusión (en este caso, un modo de transmisión en el que un destino de transmisión no se especifica por anticipado), o la comunicación (en este caso, un destino de transmisión se especifica por anticipado). Es decir, la transmisión de una señal de modulación se puede poner en práctica por una cualquiera de entre radiodifusión, transmisión por cable, radiocomunicación y comunicación por cable.

A modo de ejemplo, una estación de difusión (tal como un equipo de transmisión), y una estación de recepción (tal como un receptor de televisión) de difusión terrestre digital son, respectivamente, un ejemplo del aparato de transmisión A que transmite una señal de modulación a través de radiodifusión y un ejemplo del aparato de recepción B que recibe una señal de modulación a través de radiodifusión. Una estación de transmisión (tal como un equipo de difusión), y una estación de recepción (tal como un receptor de televisión) de difusión de televisión por cable son, respectivamente, un ejemplo del aparato de transmisión A que transmite una señal de modulación a través de una difusión por cable, y un ejemplo del aparato de recepción B, que recibe una señal de modulación a través de la difusión por cable.

Un servidor (tal como una estación de trabajo), y un cliente (tal como un receptor de televisión, un ordenador personal o un teléfono inteligente), que utilizan servicios VOD (Vídeo Bajo Demanda), o servicios de alojamiento de vídeos en Internet son, respectivamente, un ejemplo del aparato de transmisión A que transmite una señal de modulación a través de la comunicación, y el aparato de recepción B que recibe una señal de modulación a través de la comunicación (generalmente, en el caso de una red LAN, se utiliza un medio de transmisión cableado o inalámbrico, y en el caso de una red WAN, se utiliza un medio de transmisión por cable). Ejemplos de ordenador personal son un ordenador de escritorio PC, un ordenador portátil y una tableta electrónica. Un ejemplo de teléfono inteligente es un terminal de teléfono móvil multifunción.

Un cliente que utiliza servicios de alojamiento de vídeo tiene una función de codificación de imágenes de vídeo capturadas por una cámara, y cargar las imágenes de vídeo codificadas a un servidor, así como una función de decodificación de datos codificados descargados desde un servidor, y mostrar los datos decodificados en una pantalla. Es decir, un cliente que utiliza servicios de alojamiento de vídeo sirve tanto como aparato de transmisión A como aparato de recepción B.

El uso del dispositivo de decodificación de vídeo 1 y el dispositivo de codificación de vídeo 2, descritos con anterioridad, para la registro y reproducción de imágenes de vídeo, se describirá a continuación haciendo referencia a la Figura 46.

La parte (a) de la Figura 46 es un diagrama de bloques que ilustra la configuración de un aparato de registro C en el que está montado el dispositivo de decodificación de vídeo 1. Tal como se ilustra en la parte (a) de la Figura 46, el aparato de registro C incluye un codificador C1 que codifica imágenes de vídeo con el fin de obtener datos codificados, y un dispositivo de escritura C2 que escribe los datos codificados obtenidos por el codificador C1 en un medio de registro M. El dispositivo de codificación de vídeo 2, descrito anteriormente, se utiliza como este codificador C1.

El medio de registro M puede ser (1) un tipo integrado en el aparato de registro C, tal como un HDD (Unidad de Disco Duro) o un SSD (Unidad de Estado Sólido), (2) un tipo conectado al aparato de registro C, tal como una tarjeta de memoria SD, o una memoria instantánea USB (Bus Serie Universal), o (3) un tipo cargado en una unidad (no ilustrado) que está integrado en el aparato de registro C, tal como un DVD (Disco Versátil Digital) o un BD (Disco Blu-ray: marca registrada).

Como fuentes de suministro para la entrada de imágenes de vídeo en el codificador C1, el aparato de registro C puede incluir, además, una cámara C3 que captura imágenes de vídeo, un terminal de entrada C4 para recibir imágenes de vídeo desde una fuente externa, un receptor C5 para recibir imágenes de vídeo, y un procesador de imagen C6 que genera o procesa imágenes. Aunque la configuración del aparato de registro C, que incluye todos estos elementos se ilustra en la parte (a) de la Figura 46, se pueden omitir algunos elementos.

El receptor C5 puede ser un receptor que recibe imágenes de vídeo que no están codificadas, o imágenes de vídeo

que están codificadas por un método de codificación de transmisión, que es diferente de un método de codificación de registro. Si el receptor C5 es un receptor que recibe imágenes de vídeo codificadas por un método de codificación de transmisión, un decodificador de transmisión (no ilustrado) para decodificar datos codificados por un método de codificación de transmisión se puede interponer entre el receptor C5 y el codificador C1.

Ejemplos del aparato de registro C son una grabadora de DVD, una grabadora de BD y una grabadora de HD (Disco Duro) (en este caso, el terminal de entrada C4, o el receptor C5, es una fuente de alimentación principal para imágenes de vídeo). Otros ejemplos del aparato de registro C son una videocámara (en este caso, la cámara C3 es una fuente de alimentación principal para imágenes de vídeo), un ordenador personal (en este caso, el receptor C5, o el procesador de imagen C6, es una fuente de alimentación principal para imágenes de vídeo) y un teléfono inteligente (en este caso, la cámara C3, o el receptor C5, es una fuente principal de suministro de imágenes de vídeo).

La parte (b) de la Figura 46 es un diagrama de bloques que ilustra la configuración de un aparato de reproducción D en el que está montado el dispositivo de decodificación de vídeo 1. Tal como se ilustra en la parte (b) de la Figura 46, el aparato de reproducción D incluye un dispositivo de lectura D1 que lee datos codificados escritos en un medio de registro M, y un decodificador D2 que decodifica los datos codificados objeto de lectura por el dispositivo de lectura D1 con el fin de obtener imágenes de vídeo. El dispositivo de decodificación de vídeo 1, descrito anteriormente, se utiliza como este decodificador D2.

El medio de registro M puede ser (1) un tipo integrado en el aparato de reproducción D, tal como un disco duro HDD o un SSD, (2) un tipo conectado al aparato de reproducción D, tal como una tarjeta de memoria SD, o una memoria instantánea USB, o (3) un tipo cargado en una unidad (no ilustra) integrada en el aparato de reproducción D, tal como un DVD o un BD.

Como destinos a los que se suministran imágenes de vídeo desde el decodificador D2, el aparato de reproducción D puede incluir, además, una pantalla D3 que muestra imágenes de vídeo, un terminal de salida D4 para emitir imágenes de vídeo a una fuente externa, y un transmisor D5 que transmite imágenes de vídeo. Aunque la configuración del aparato de reproducción D, que incluye todos estos elementos se ilustra en la parte (b) de la Figura 46, se pueden omitir algunos elementos.

El transmisor D5 puede ser un transmisor que transmite imágenes de vídeo que no están codificadas, o imágenes de vídeo que están codificadas por un método de codificación de transmisión, que es diferente de un método de codificación de registro. Si el transmisor D5 es un transmisor que transmite imágenes de vídeo codificadas por un método de codificación de transmisión, se puede interponer un codificador (no ilustrado) para codificar imágenes de vídeo mediante un método de codificación de transmisión entre el decodificador D2 y el transmisor D5.

Los ejemplos del aparato de reproducción D son un reproductor de DVD, un reproductor BD y un reproductor HDD (en este caso, el terminal de salida D4 al que está conectado un receptor de televisión, a modo de ejemplo, está conectado a un destino principal al que se suministran las imágenes de vídeo). Otros ejemplos del aparato de reproducción D son un receptor de televisión (en este caso, la pantalla D3 es un destino principal al que se suministran las imágenes de vídeo), un PC de escritorio (en este caso, el terminal de salida D4, o el transmisor D5, es un destino principal al que se suministran imágenes de vídeo), un ordenador portátil o tableta electrónica PC (en este caso, la pantalla D3, o el transmisor D5, es un destino principal al que se proporcionan imágenes de vídeo), un teléfono inteligente (en este caso, la pantalla D3 o el transmisor D5 es un destino principal al que se suministran imágenes de vídeo) y la señalización digital (también denominada como un sistema de placa de boletín electrónico y, en este caso, la pantalla D3 o el transmisor D5 es un destino principal al que se suministran las imágenes de vídeo).

(Configuración puesta en práctica mediante software)

Los bloques individuales del dispositivo de decodificación de vídeo 1 (1') y el dispositivo de codificación de vídeo 2 (2'), en particular, el decodificador de código de longitud variable 13, la unidad de reconstrucción de vector de movimiento 14, el generador de imagen de inter-predicción 16, el generador de imagen de intra-predicción 17, la unidad de determinación del método de predicción 18, la unidad de transformación inversa y cuantización inversa 19, el filtro de desbloqueo 41, el filtro adaptativo 50, el filtro de desplazamiento de adaptación 60 (60'), la unidad de transformación y cuantización 21, el codificador de código de longitud variable 22, la unidad de transformación inversa y cuantización inversa 23, el generador de imagen de intra-predicción 25, el generador de imagen de inter-predicción 26, el detector de vector de movimiento 27, el controlador de método de predicción 28, la unidad de eliminación de redundancia de vector de movimiento 29, el filtro de desbloqueo 33, el filtro adaptativo 70, y el filtro de desplazamiento adaptativo 80 (80') se pueden poner en práctica mediante hardware utilizando un circuito lógico formado en un circuito integrado (circuito integrado IC), o se puede poner en práctica mediante software utilizando una CPU (unidad central de procesamiento).

Si los elementos descritos anteriormente se ponen en práctica mediante un software, el dispositivo de decodificación de vídeo 1 y el dispositivo de codificación de vídeo 2 incluyen, cada uno, una unidad CPU que ejecuta órdenes de un

programa de control que pone en práctica las funciones individuales, una memoria ROM (memoria de solamente lectura) que memoriza este programa, una memoria RAM (memoria de acceso aleatorio) que carga este programa, un dispositivo de memorización (medio de registro), tal como una memoria, que memoriza este programa y varios elementos de datos, etc. El objeto de la presente invención se puesta en práctica, además, proporcionando un medio de registro en el que un código de programa (un programa de forma de ejecución, un programa de código intermedio y un programa fuente), del programa de control para el dispositivo de decodificación de vídeo 1 y el dispositivo de codificación de vídeo 2, que es un software que pone en práctica las funciones descritas anteriormente, se registra, de forma legible por ordenador, para el dispositivo de decodificación de vídeo 1 y el dispositivo de codificación de vídeo 2, y mediante la lectura y realización del código de programa, que se registra en el medio de registro por un ordenador (o una CPU o una MPU (unidad de microprocesador) de cada uno de entre el dispositivo de decodificación de vídeo 1 y el dispositivo de codificación de vídeo 2.

Como el medio de registro descrito anteriormente, a modo de ejemplo, se puede utilizar un tipo de cinta, tal como una cinta magnética o una cinta de casete, un tipo de disco que incluye un disco magnético, como un disquete (marca registrada), o un disco duro, y un disco óptico, tal como un CD-ROM (memoria de solamente lectura de disco compacto), un disco MO (disco magneto-óptico), un MD (Mini-disco), un DVD (disco versátil digital) o un CD-R (CD grabable), un tipo de tarjeta, tal como una tarjeta IC (incluyendo una tarjeta de memoria), o una tarjeta óptica, un tipo de memoria de semiconductor, tal como una memoria ROM de máscara, una memoria EPROM (memoria de solamente lectura programable borrrable), una memoria EEPROM (memoria de solamente lectura programable y eléctricamente borrrable) (marca registrada), o una ROM instantánea, o un tipo de circuito lógico, tal como un PLD (Dispositivo Lógico Programable) o una FPGA (Matriz de Puerta Programable de Campo).

El dispositivo de decodificación de vídeo 1, y el dispositivo de codificación de vídeo 2, se pueden conectar a una red de comunicación y el código de programa descrito anteriormente se puede proporcionar a través de la red de comunicación. Esta red de comunicación no está particularmente restringida siempre que sea capaz de transmitir el código del programa. A modo de ejemplo, se puede utilizar Internet, una intranet, una extranet, una red LAN (red de área local), RDSI (red digital de servicios integrados), una VAN (red de valor añadido), una red de comunicación CATV (antena comunitaria de televisión/televisión por cable), una VPN (red privada virtual), una red telefónica conmutada pública, una red de comunicación móvil, una red de comunicación vía satélite, etc. Además, un medio de transmisión que forma esta red de comunicación no está restringido a una configuración específica, o un tipo específico, siempre que sea capaz de transmitir el código del programa. A modo de ejemplo, un medio de transmisión por cable, tal como IEEE (instituto de ingenieros eléctricos y electrónicos) 1394, USB, comunicación por línea eléctrica, una línea de TV por cable, una línea telefónica, o un circuito ADSL (bucle de abonado digital asimétrico) o un medio de transmisión inalámbrica, tal como infrarrojos, a modo de ejemplo, IrDA (asociación de datos por infrarrojos) o un controlador distante, Bluetooth (marca registrada), IEEE802. 11 radio, HDR (alta tasa de datos), NFC (Comunicación de Campo Cercano), DLNA (Alianza para el Estilo de Vida Digital en Red), una red de telefonía móvil, un circuito de satélite, o una red digital terrestre.

La presente invención no está restringida a las formas de realización descritas con anterioridad, y se pueden realizar diversas modificaciones dentro del alcance de las reivindicaciones.

Aplicabilidad industrial

La presente invención se aplica, de forma adecuada, a un filtro de imagen que realiza un filtrado de desplazamiento en datos de imagen. La invención se puede aplicar adecuadamente, además, a un dispositivo de decodificación que decodifica datos codificados, y a un dispositivo de codificación que codifica datos codificados.

Lista de signos de referencia

1 Dispositivo de decodificación de vídeo (dispositivo de decodificación)

41 Filtro de desbloqueo

50 Filtro adaptativo

60, 60' Filtro de desplazamiento adaptativo (dispositivo de filtrado de imagen)

611, 611' Sección de decodificación de información de desplazamiento (dispositivo de decodificación de desplazamiento, medio de determinación, medio de establecimiento de atributos de desplazamiento, medio de decodificación de desplazamiento, medio de decodificación de residuo de desplazamiento, medio de determinación de valor de predicción)

612 Sección de decodificación de estructura de QAOU

613 Sección de establecimiento de atributos de desplazamiento (medio de establecimiento de atributos de desplazamiento)

	621 Sección de memorización de información de desplazamiento
5	622 Controlador de QAOU
	623 Sección de determinación de tipo de desplazamiento
	624 Sección de clasificación (medio de cálculo, medio de desplazamiento binario, medio de clasificación)
10	625 Sección de determinación de desplazamiento (medio de desplazamiento inverso)
	626 Sumador de desplazamiento (medio de desplazamiento)
15	2 Dispositivo de codificación de vídeo (dispositivo de codificación)
	33 Filtro de desbloqueo
	70 Filtro adaptativo
20	80, 80' Filtro de desplazamiento adaptativo (dispositivo de filtrado de imagen)
	811 Calculador de desplazamiento
25	812 Sección de recorte de desplazamiento
	813 Selector de información de desplazamiento
	814 Sección de determinación de residuo de desplazamiento
30	815 Sección de establecimiento de atributos de desplazamiento (medio de establecimiento de atributos de desplazamiento)
	816 Desplazador de desplazamiento (medio de desplazamiento)

REIVINDICACIONES

1. Un dispositivo de filtrado de imagen para recibir una imagen decodificada desbloqueada, y para añadir un desplazamiento a un valor de píxel de cada píxel de una zona unitaria de la imagen decodificada desbloqueada que está constituida por una pluralidad de zonas unitarias, que comprende:

una unidad de establecimiento de atributos de desplazamiento configurada para establecer un margen de valores de desplazamiento de conformidad con una profundidad de bits de desplazamiento SAO_DEPTH, en donde la SAO_DEPTH se establece haciendo referencia a datos codificados, y en donde una longitud binaria máxima que representa el margen de valores de desplazamiento es igual a $\text{SAO_DEPTH} - K$ y el valor de K es igual a 4;

una unidad de decodificación de desplazamiento configurada para decodificar un desplazamiento que está restringido al margen de valores de desplazamiento y asociado con una de una pluralidad de clases de desplazamiento;

un medio de cálculo para determinar el signo de un valor de diferencia entre un valor de píxel de un píxel objeto de la zona unitaria y cada uno de los valores de píxel de dos píxeles que son adyacentes al píxel objeto;

un medio de clasificación para clasificar el píxel objeto como uno de una pluralidad de clases de desplazamiento de conformidad con el signo del valor de diferencia calculado mediante el medio de cálculo cuando un tipo de desplazamiento es un tipo de desplazamiento de borde; y

una unidad de filtrado configurada para añadir el desplazamiento asociado con la clase de desplazamiento del píxel objeto clasificado por el medio de clasificación al valor de píxel del píxel objeto de la zona unitaria de la imagen decodificada desbloqueada.

2. El dispositivo de filtrado de imagen según la reivindicación 1, en donde:

la unidad de establecimiento de atributos de desplazamiento está configurada para establecer la profundidad de bits de desplazamiento de conformidad con una profundidad de bits de valores de píxel.

3. El dispositivo de filtrado de imagen según la reivindicación 2, en donde la unidad de establecimiento de atributos de desplazamiento está configurada para limitar la profundidad de bits de desplazamiento al establecer que la profundidad de bits de desplazamiento sea igual a la profundidad de bits de los valores de píxel en un caso en que la profundidad de bits de los valores de píxel sea diez o más pequeño, y la unidad de establecimiento de atributos de desplazamiento está configurada para establecer que la profundidad de bits de desplazamiento sea diez, en un caso en que la profundidad de bits de los valores de píxel sea once o mayor, en donde

la unidad de establecimiento de atributos de desplazamiento está configurada para establecer que un margen de valores de desplazamiento máximo sea $2^{(\text{la profundidad de bits de desplazamiento} - K) - 1} - 1$ o menor, en donde el valor de K es igual a un número entero.

4. Un método de filtrado para recibir una imagen decodificada desbloqueada y para añadir un desplazamiento a un valor de píxel de cada píxel de una zona unitaria de la imagen decodificada desbloqueada que está constituida por una pluralidad de zonas unitarias, que comprende:

establecer un margen de valores de desplazamiento de conformidad con una profundidad de bits de desplazamiento SAO_DEPTH, en donde la SAO_DEPTH se establece haciendo referencia a datos codificados, y en donde una longitud binaria máxima que representa el margen de valores de desplazamiento es igual a $\text{SAO_DEPTH} - K$ y el valor de K es igual a 4;

decodificar un desplazamiento que está restringido al margen de valores de desplazamiento y asociado con una de una pluralidad de clases de desplazamiento;

determinar el signo de un valor de diferencia entre un valor de píxel de un píxel objeto de la zona unitaria y cada uno de los valores de píxel de dos píxeles que son adyacentes al píxel objeto;

clasificar el píxel objeto como uno de una pluralidad de clases de desplazamiento de conformidad con el signo del valor de diferencia cuando un tipo de desplazamiento es un tipo de desplazamiento de borde;

añadir el desplazamiento asociado con la clase de desplazamiento del píxel objeto al valor de píxel del píxel objeto de la zona unitaria de la imagen decodificada desbloqueada.

5. El método de filtrado de imagen según la reivindicación 4, en donde:

la profundidad de bits de desplazamiento se limita al establecerse de conformidad con una profundidad de bits de los valores de píxel.

6. El método de filtrado de imagen según la reivindicación 5, en donde la profundidad de bits de desplazamiento se establece para que sea igual a la profundidad de bits de los valores de píxel en un caso en el que la profundidad de bits de los valores de píxel es diez o menor, y la unidad de establecimiento de atributos de desplazamiento está configurada para establecer que la profundidad de bits de desplazamiento sea diez en un caso en el que la profundidad de bits de los valores de píxel es once o mayor, en donde

un margen de valores de desplazamiento máximo ha de ser $2^{(\text{la profundidad de bits de desplazamiento} - K) - 1} - 1$ o menor, en donde el valor de K es igual a un número entero.

FIG. 1

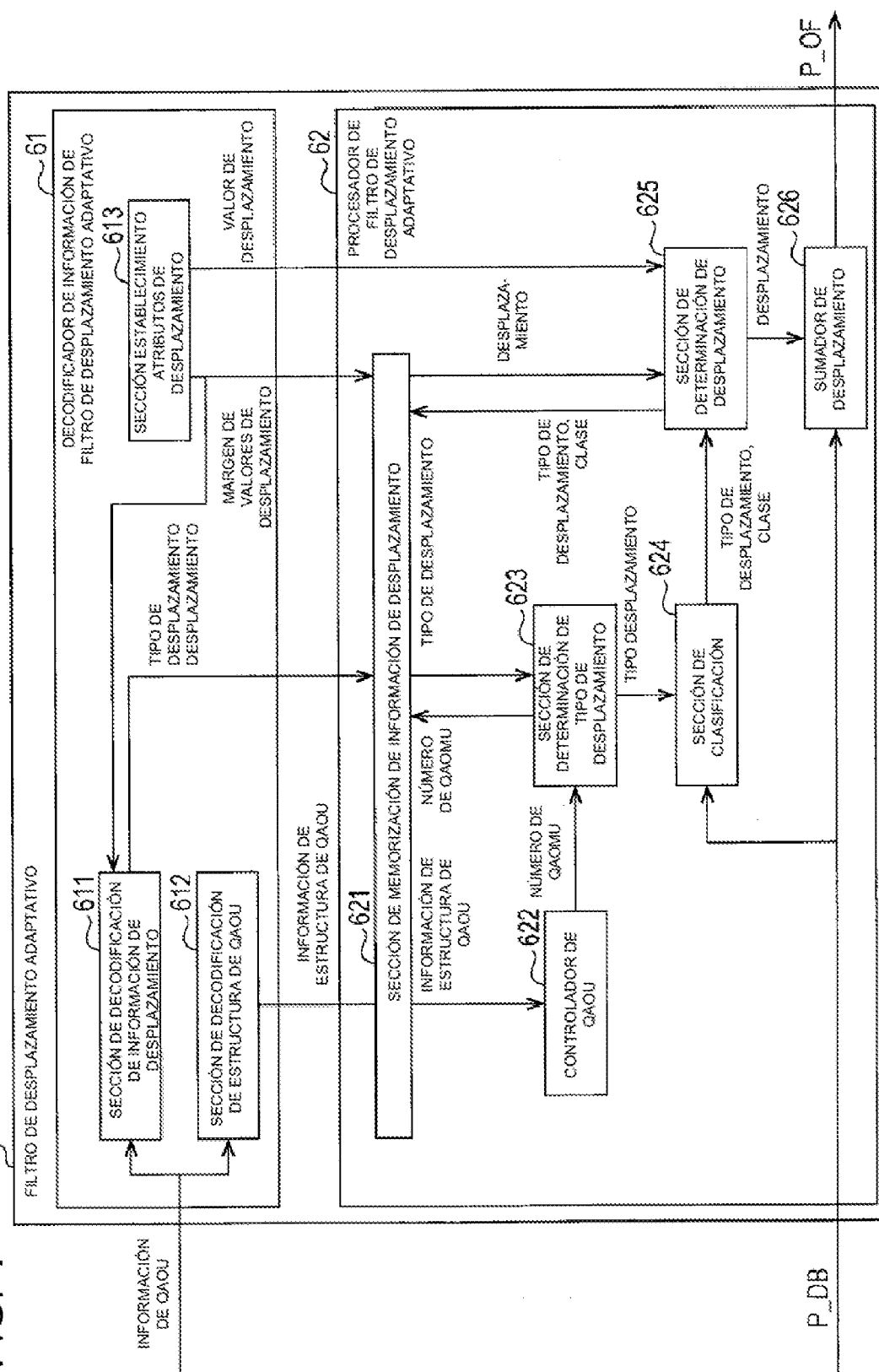


FIG. 2

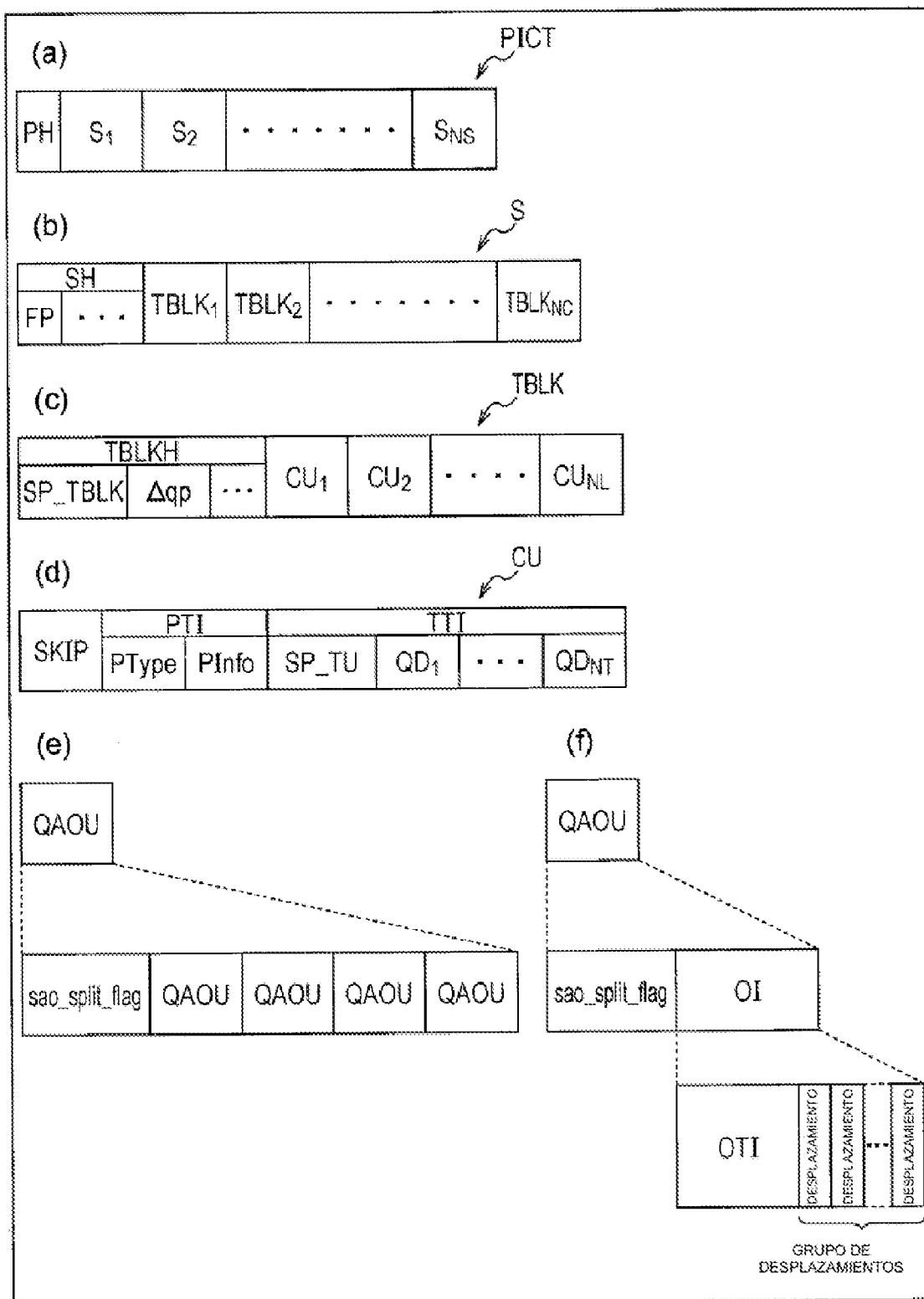


FIG. 3

sao_offset_param(xs, ys, sao_curr_depth) {	C	Descriptor
if (sao_split_flag [sao_curr_depth][ys][xs]) {		
sao_offset_param(xs + 0, ys + 0, sao_curr_depth + 1)		
sao_offset_param(xs + 1, ys + 0, sao_curr_depth + 1)		
sao_offset_param(xs + 0, ys + 1, sao_curr_depth + 1)		
sao_offset_param(xs + 1, ys + 1, sao_curr_depth + 1)		
} else {		
sao_type_idx [sao_curr_depth][ys][xs]	2	ue(v)
curr_num = sao_offset_num [sao_type_idx [sao_curr_depth][ys][xs]]		
if (sao_type_idx [sao_curr_depth][ys][xs] != 0) {		
for (i = 0; i < curr_num; i++) {		
sao_offset [sao_curr_depth][ys][xs][i]	2	se(v)
}		
}		
}		
}		

FIG. 4

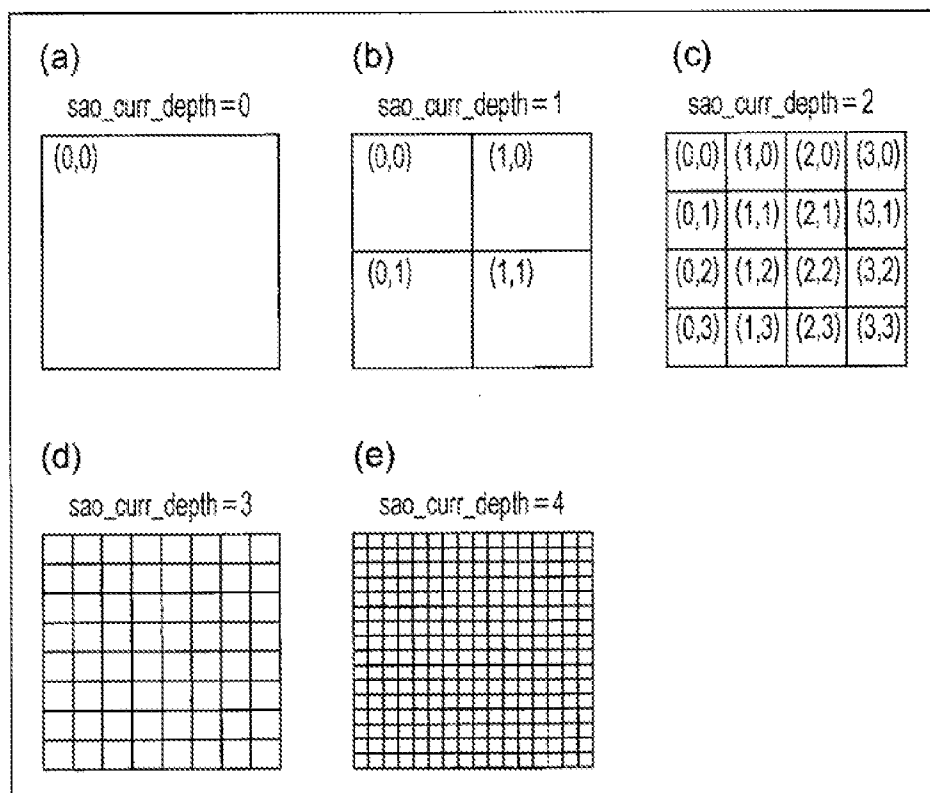


FIG. 5

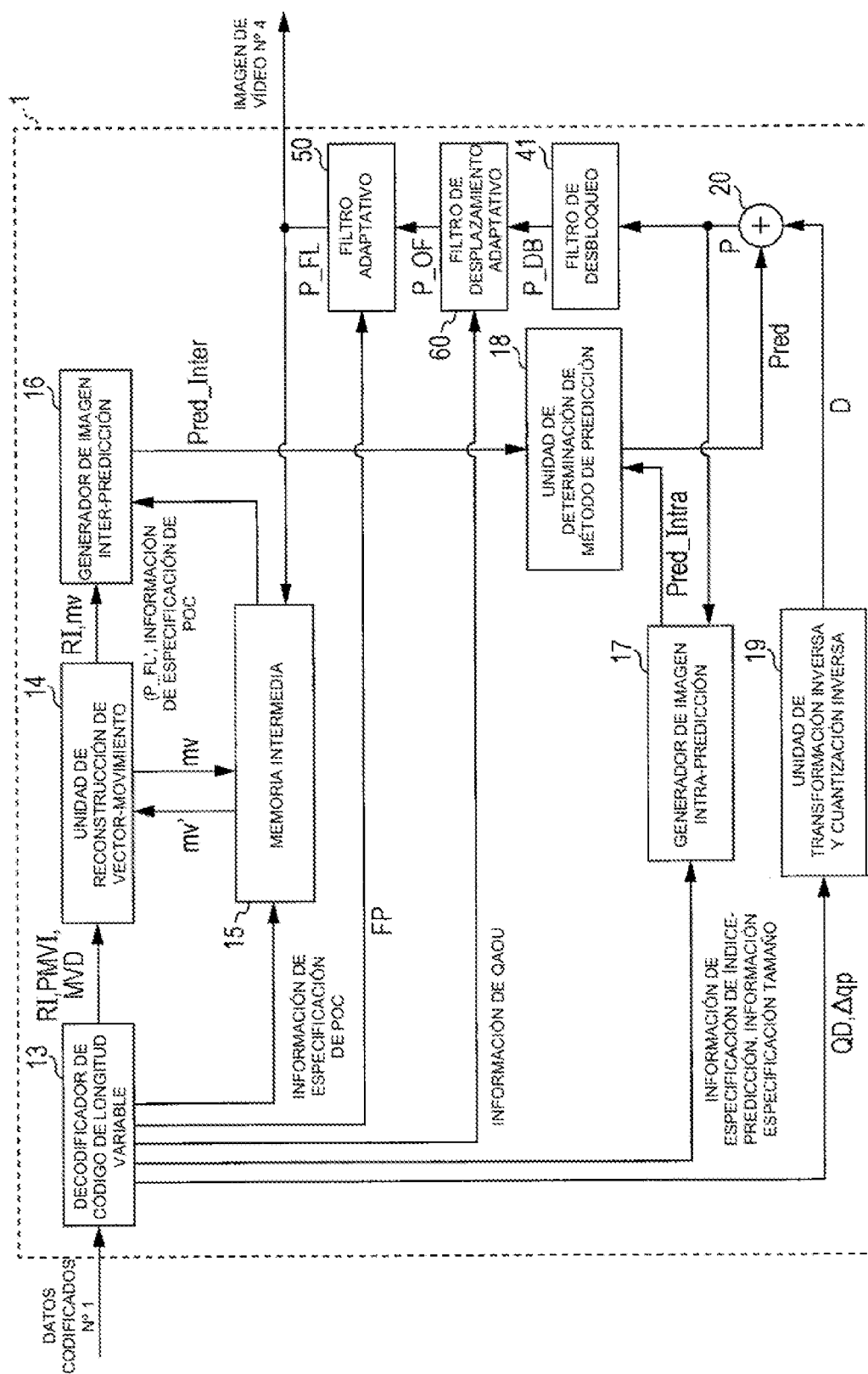


FIG. 6

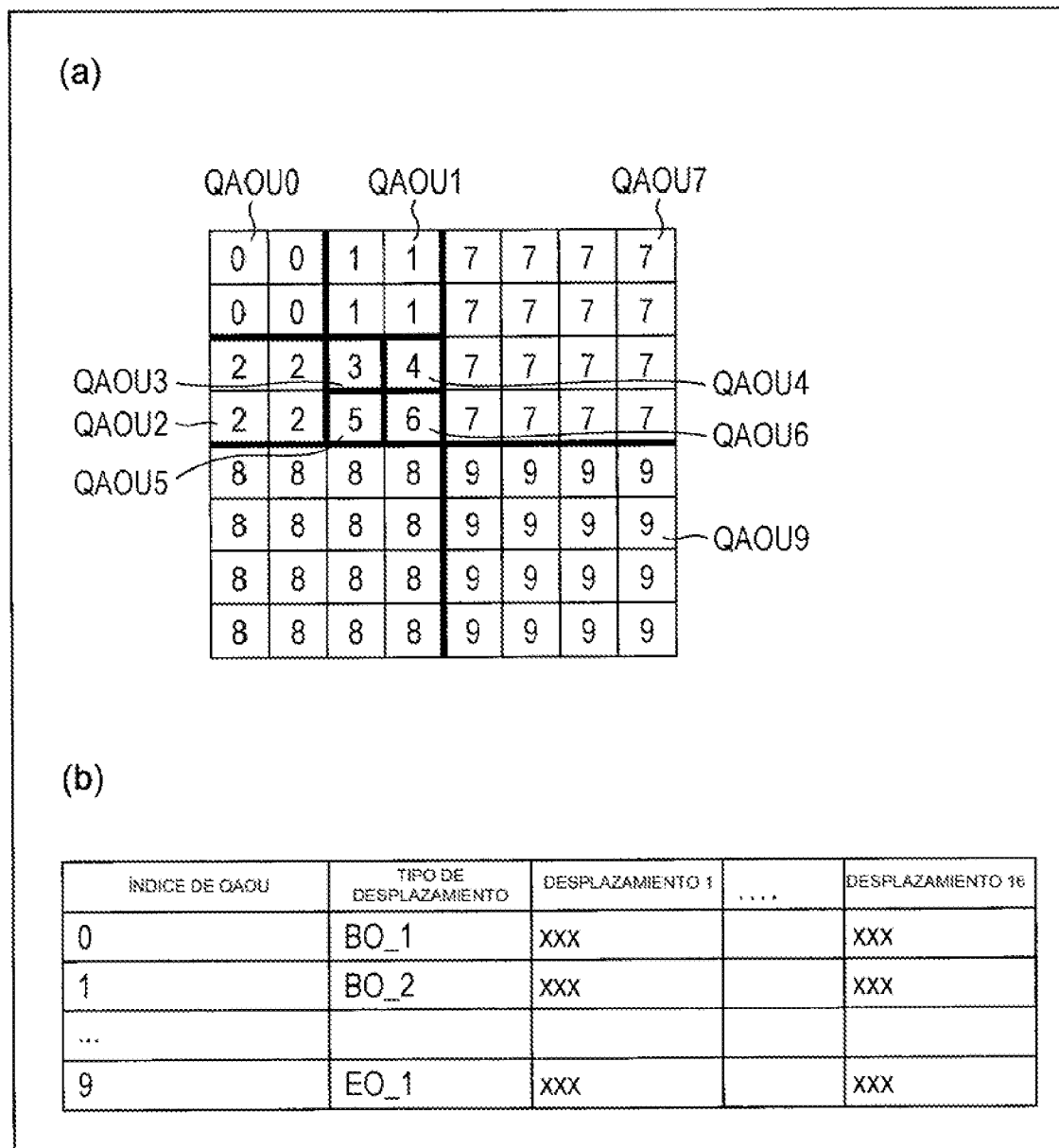


FIG. 7

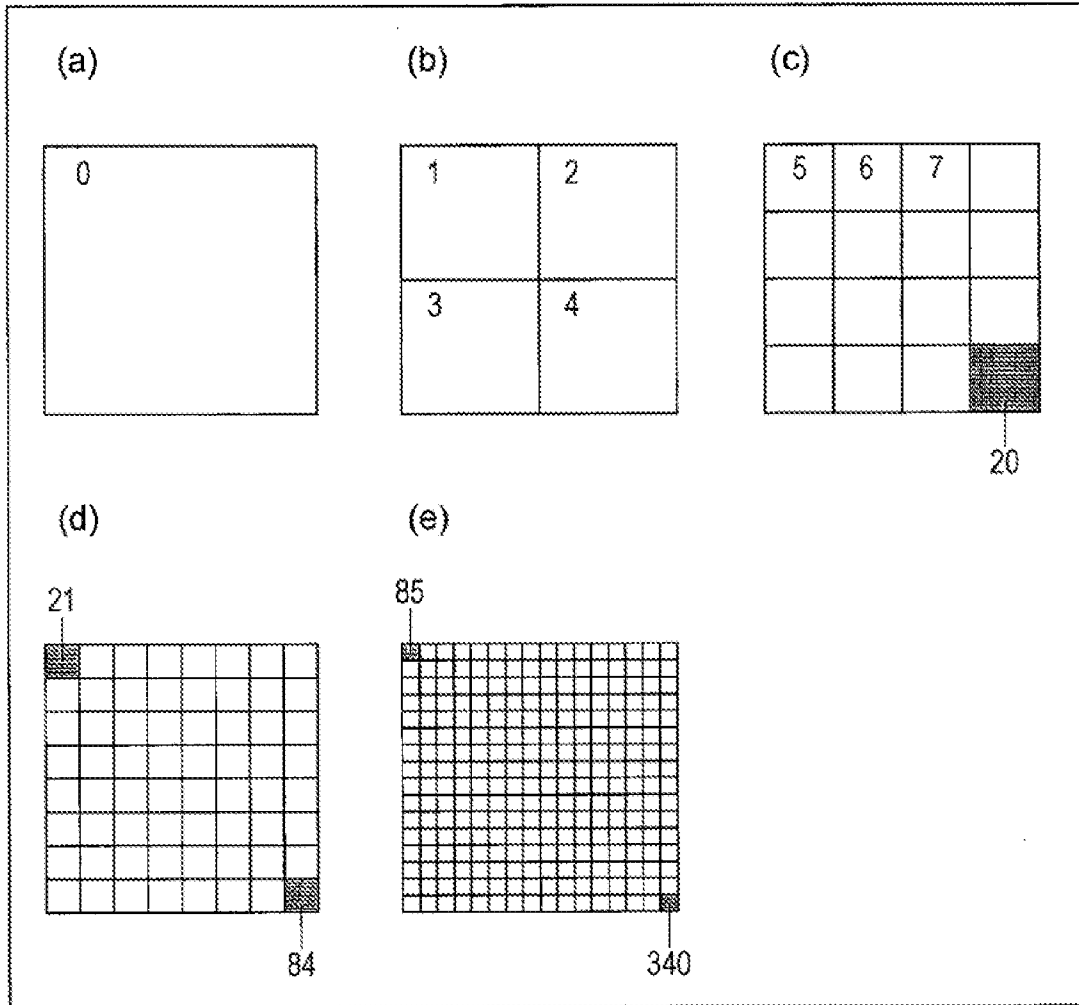


FIG. 8

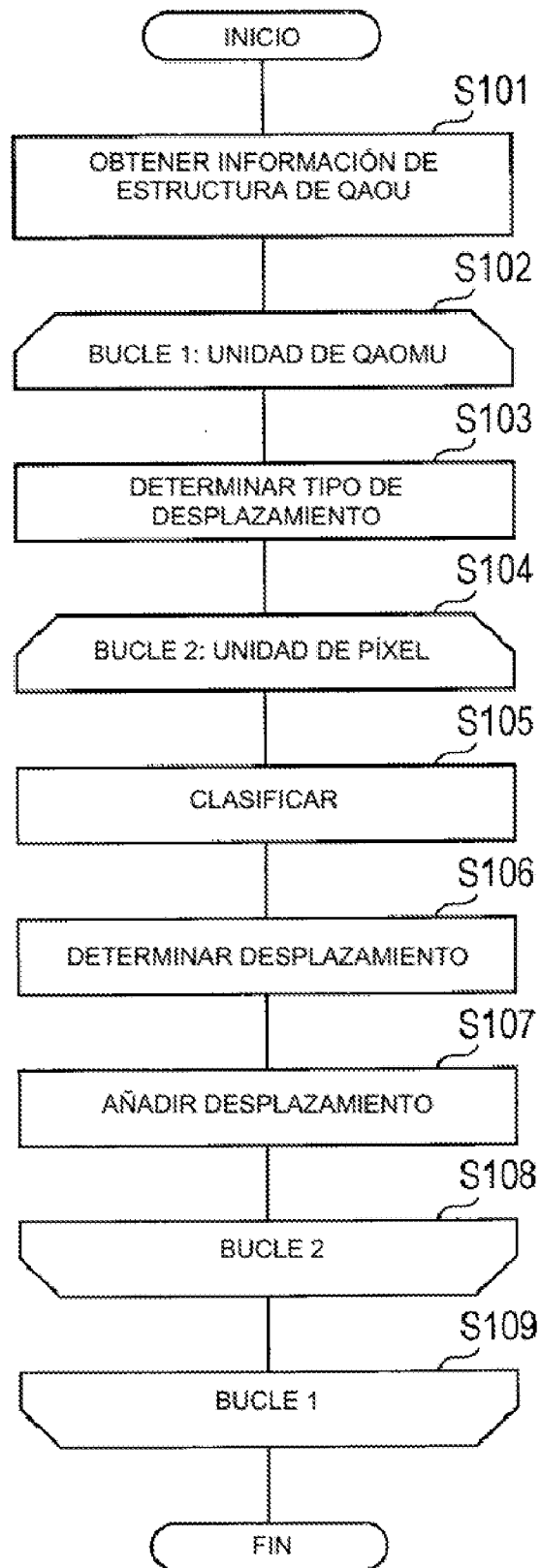


FIG. 9

(a)			(b)			(c)		
PROF. BITS PIXEL PIC_ DEPTH	PROF. BITS DESPLAZ. SAO_ DEPTH	VALOR DESPLAZ. (PIC_DEP TH_SAO_ DEPTH)	PROF. BITS PIXEL PIC_ DEPTH	PROF. BITS DESPLAZ. SAO_ DEPTH	VALOR DESPLAZ. (PIC_DEP TH_SAO_ DEPTH)	PROF. BITS PIXEL PIC_ DEPTH	PROF. BITS DESPLAZ. SAO_ DEPTH	VALOR DESPLAZ. (PIC_DEP TH_SAO_ DEPTH)
14	14	0	14	10	4	14	9	5
13	13	0	13	10	3	13	9	4
12	12	0	12	10	2	12	9	3
11	11	0	11	10	1	11	9	2
10	10	0	10	10	0	10	9	1
9	9	0	9	9	0	9	9	0
8	8	0	8	8	0	8	8	0

(d)		
PROFUNDIDAD BITS PIXEL PIC_DEPTH	PROFUNDIDAD BITS DESPLAZAMIENTO SAO_DEPTH	VALOR DESPLAZAMIENTO (PIC_DEPTH_SAO_DEPTH)
14	$10 + \text{floor}((\text{PIC_BIT} - 10) / \text{STEP})$	$\text{PIC_BIT} - 10 - \text{floor}((\text{PIC_BIT} - 10) / \text{STEP})$
13	$10 + \text{floor}((\text{PIC_BIT} - 10) / \text{STEP})$	$\text{PIC_BIT} - 10 - \text{floor}((\text{PIC_BIT} - 10) / \text{STEP})$
12	$10 + \text{floor}((\text{PIC_BIT} - 10) / \text{STEP})$	$\text{PIC_BIT} - 10 - \text{floor}((\text{PIC_BIT} - 10) / \text{STEP})$
11	$10 + \text{floor}((\text{PIC_BIT} - 10) / \text{STEP})$	$\text{PIC_BIT} - 10 - \text{floor}((\text{PIC_BIT} - 10) / \text{STEP})$
10	10	0
9	9	0
8	8	0

(e)		
PROFUNDIDAD BITS PIXEL PIC_DEPTH	PROFUNDIDAD BITS DESPLAZAMIENTO SAO_DEPTH	VALOR DESPLAZAMIENTO (PIC_DEPTH_SAO_DEPTH)
14	12	2
13	11	2
12	11	1
11	10	1
10	10	0
9	9	0
8	8	0

FIG. 10

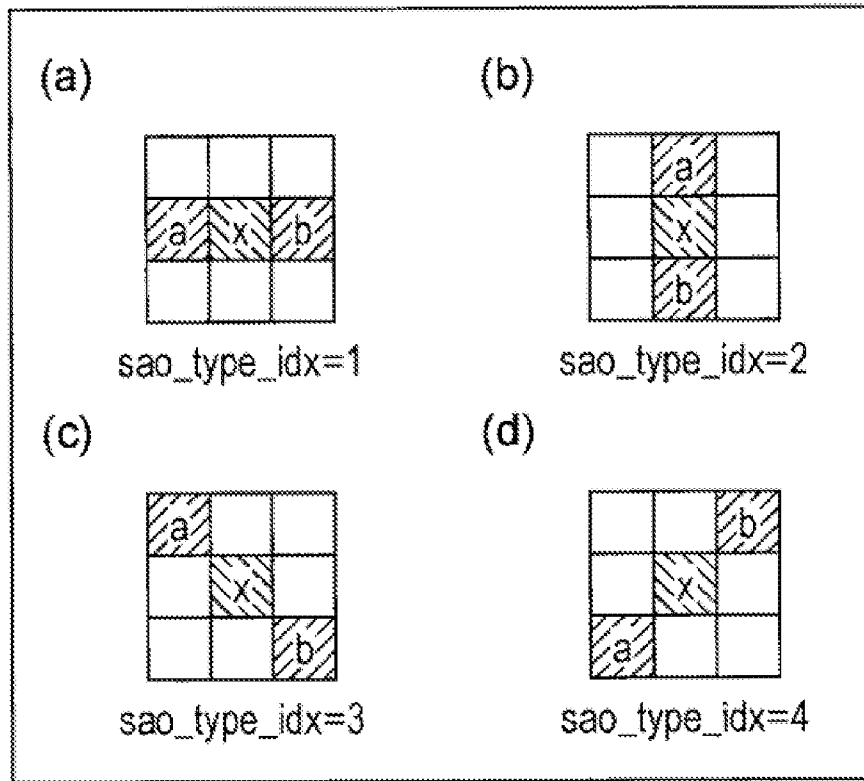


FIG. 11

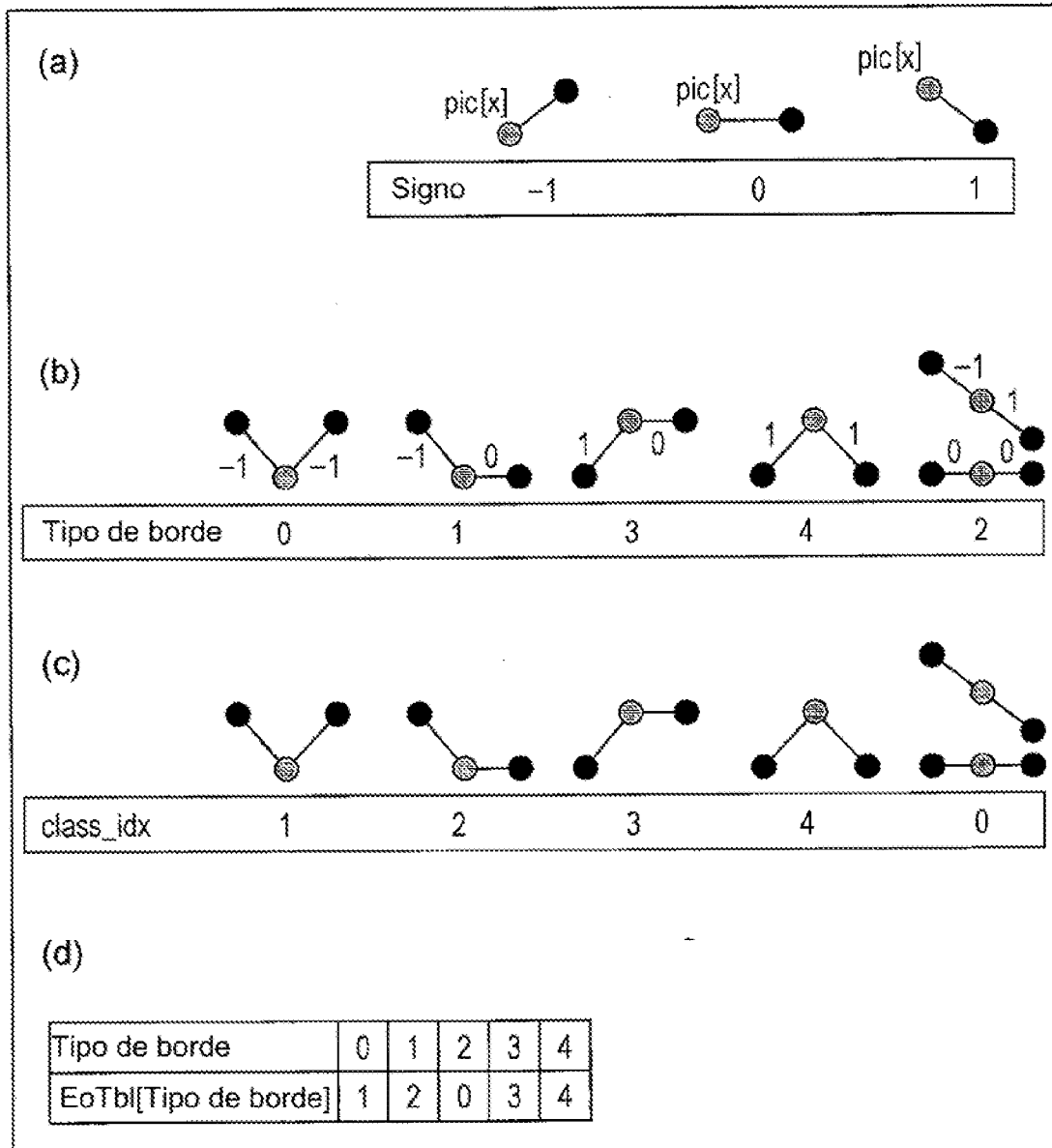


FIG. 12

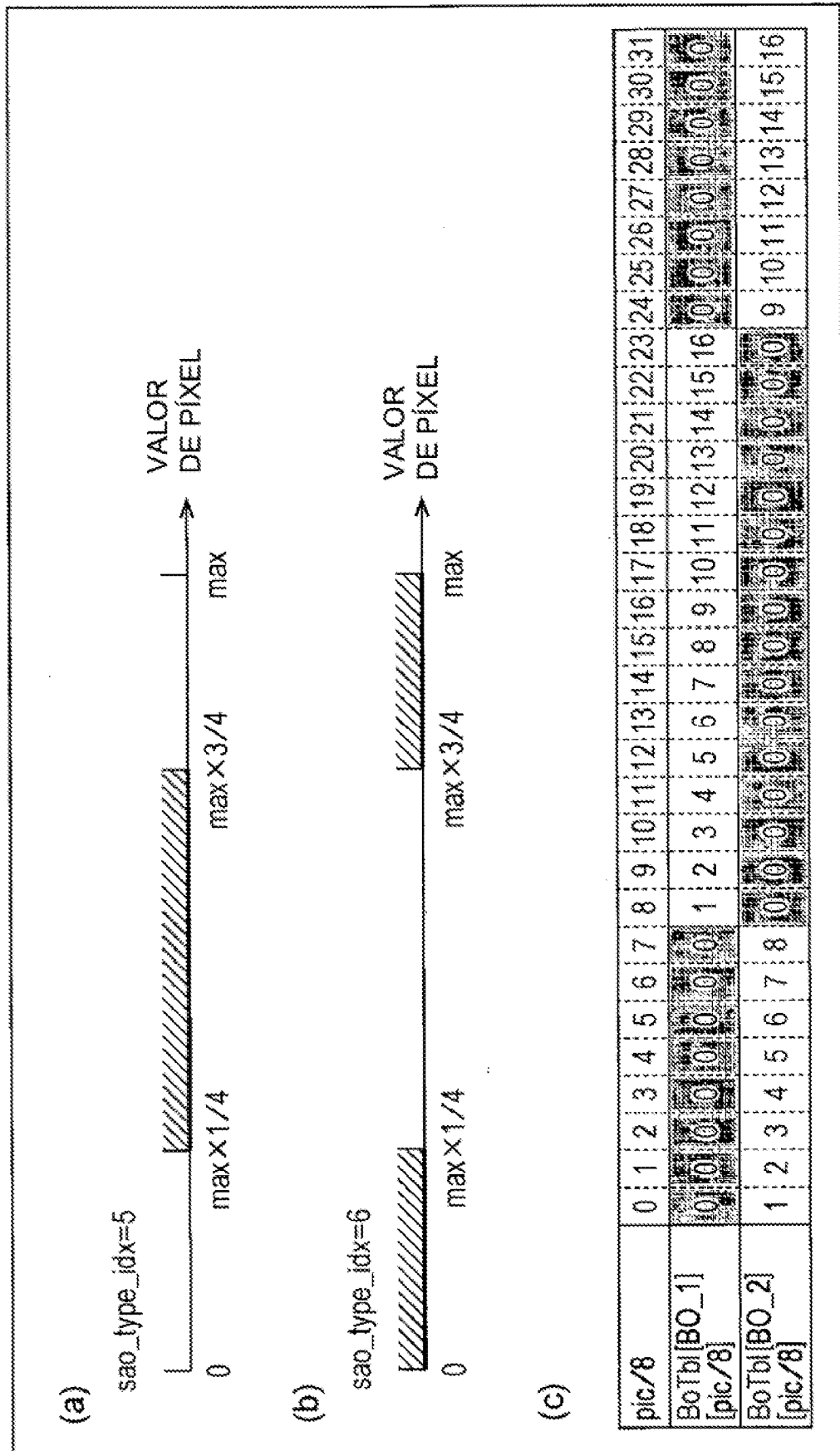


FIG. 13

pic/8	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
BoTbl[BO_1] [pic/8]	C	C	0	0	0	0	0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	0	0	0	0	0	0	0	C	C
BoTbl[BO_2] [pic/8]	C	C	1	2	3	4	5	6	7	8	10	0	0	0	0	0	0	0	0	0	0	0	0	9	10	11	12	13	14	15	16	C	C

FIG. 14

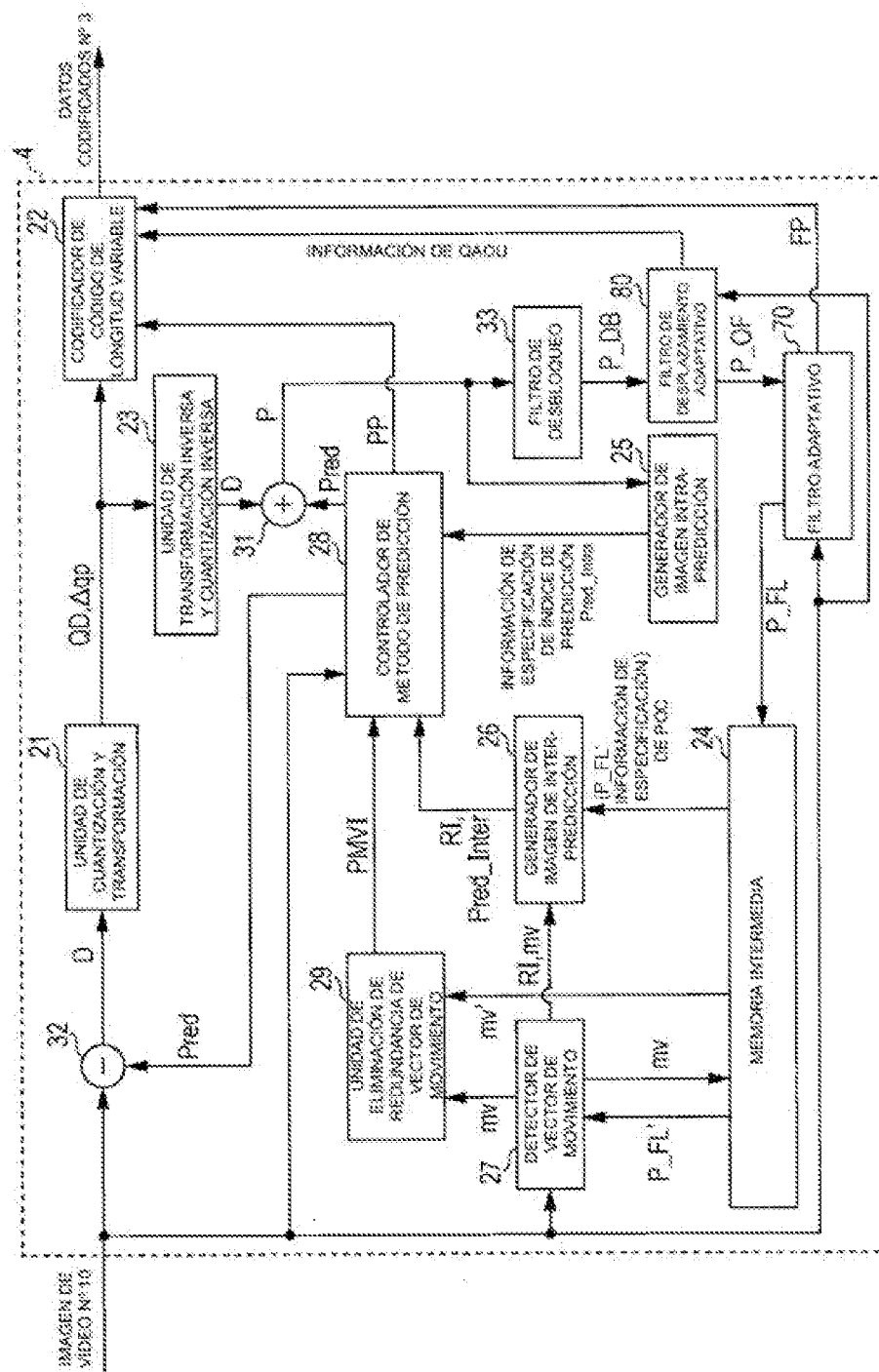


FIG. 15

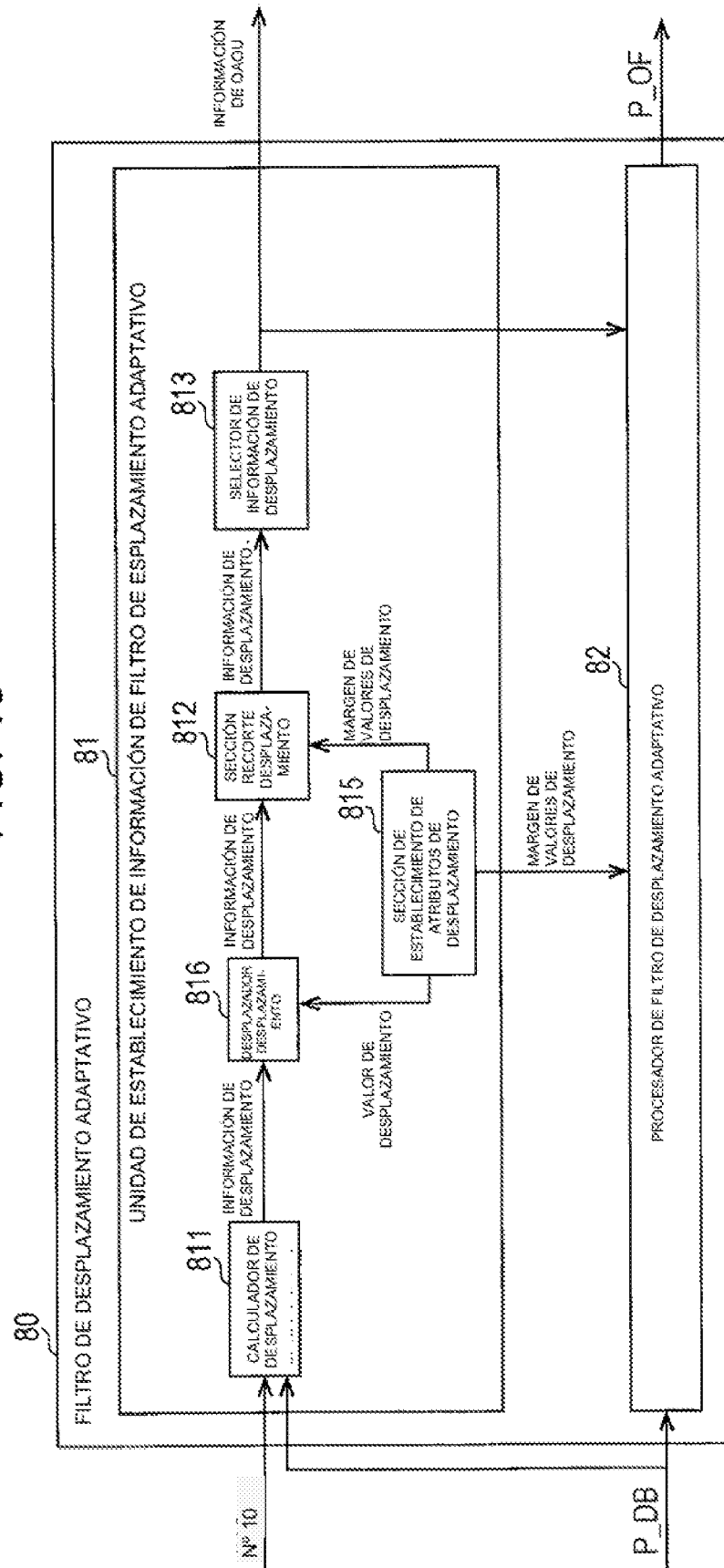


FIG. 16

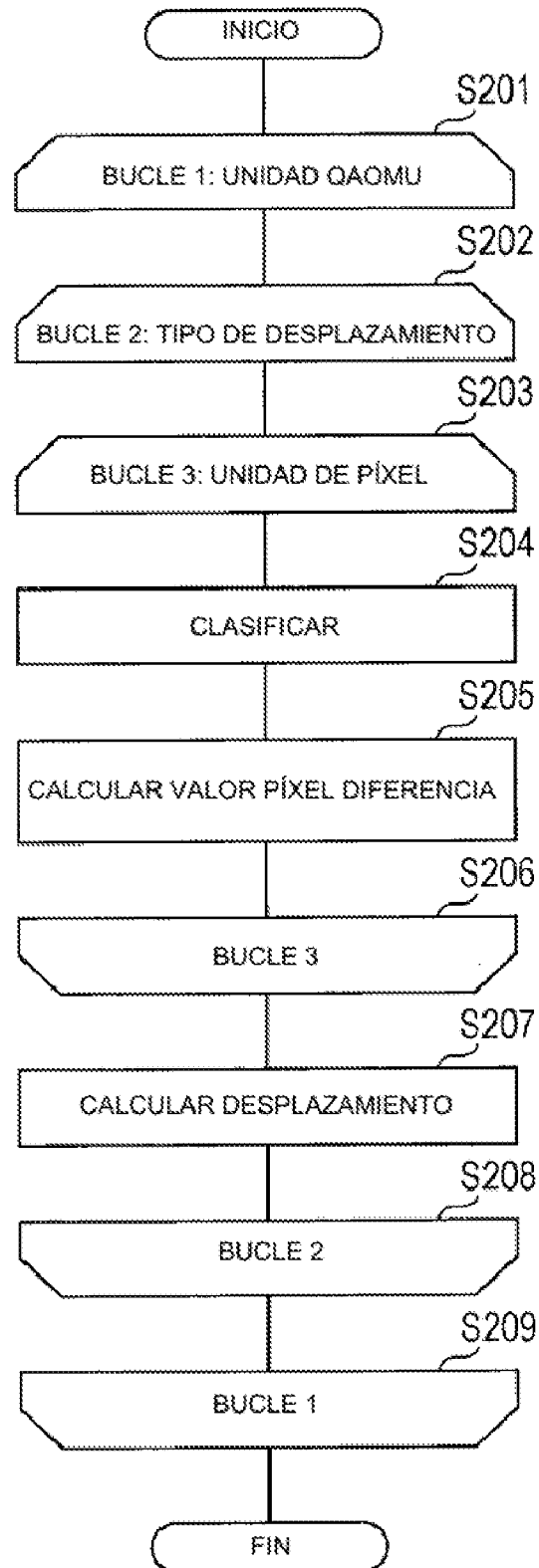


FIG. 17

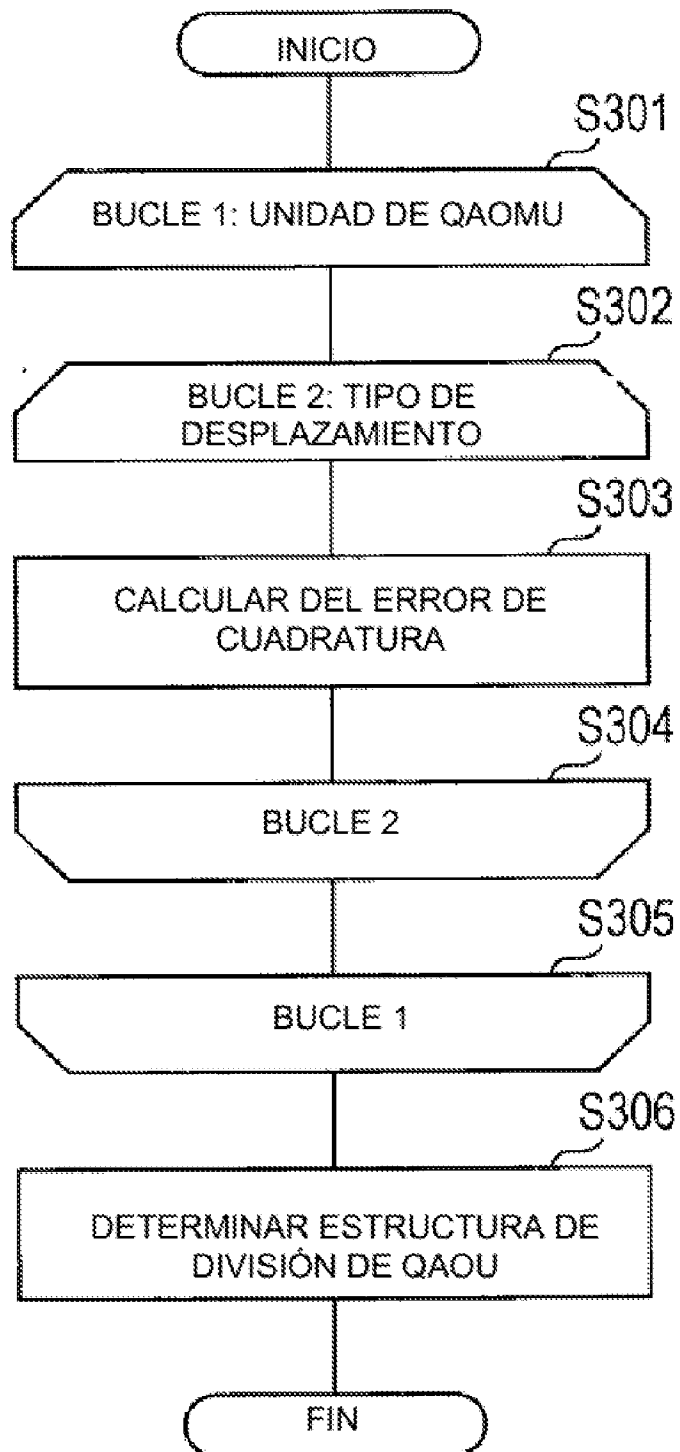
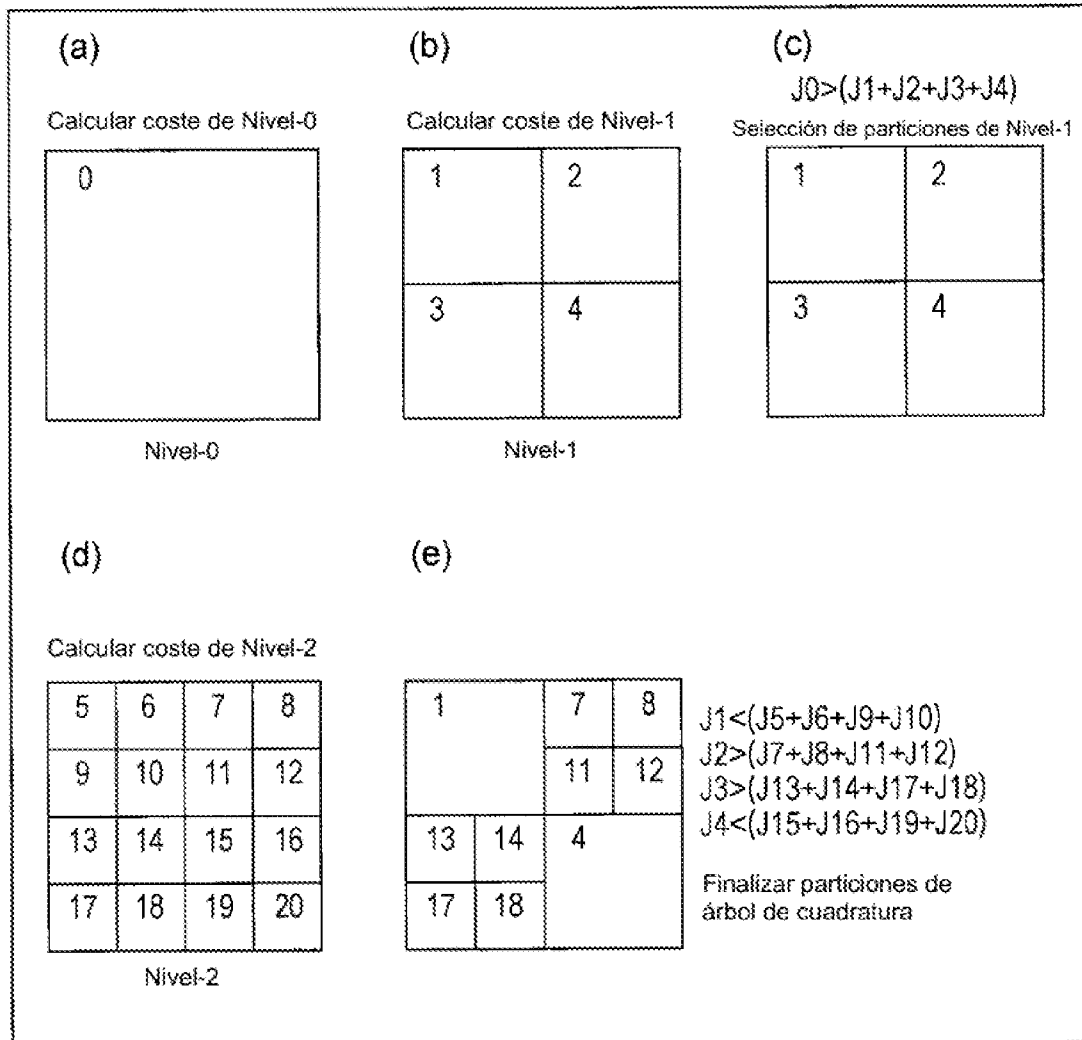


FIG. 18



901

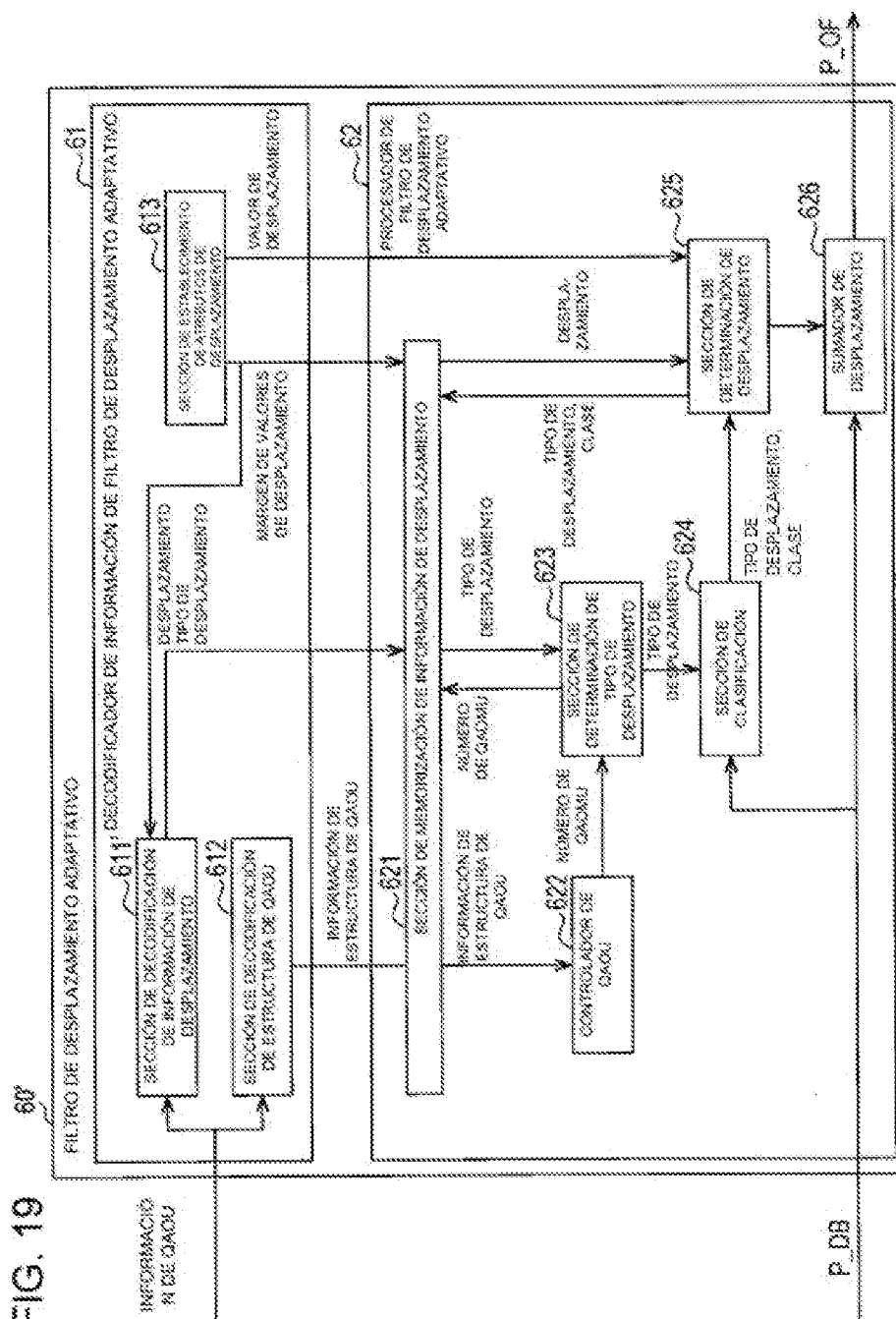


FIG. 20

(a)

sao_type_idx	merge_tbl[sao_type_idx]
0	NA
1	0
2	1
3	2
4	3
5	4
6	5

(b)

sao_type_idx	merge_tbl[sao_type_idx]
0	NA
1	0
2	0
3	0
4	0
5	1
6	2

FIG. 21

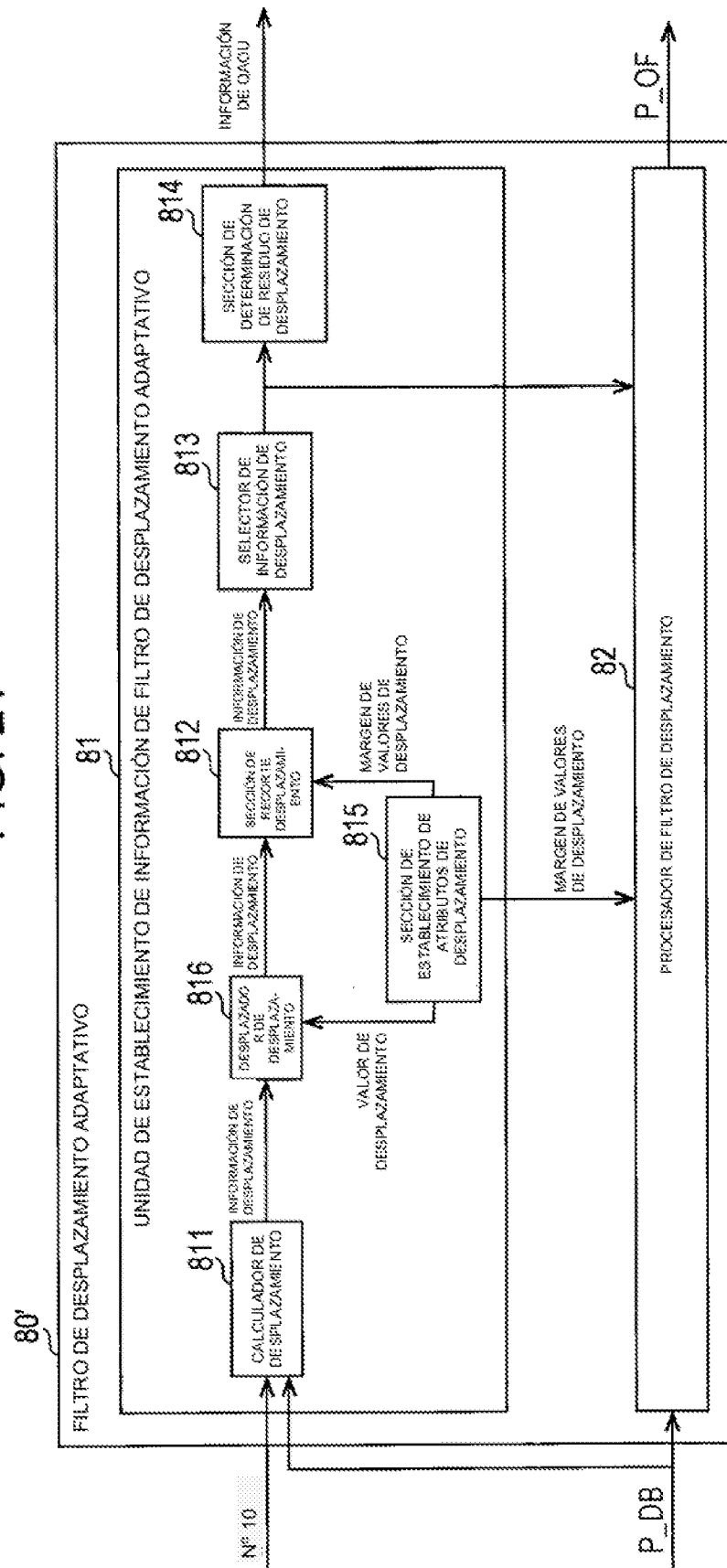


FIG. 22

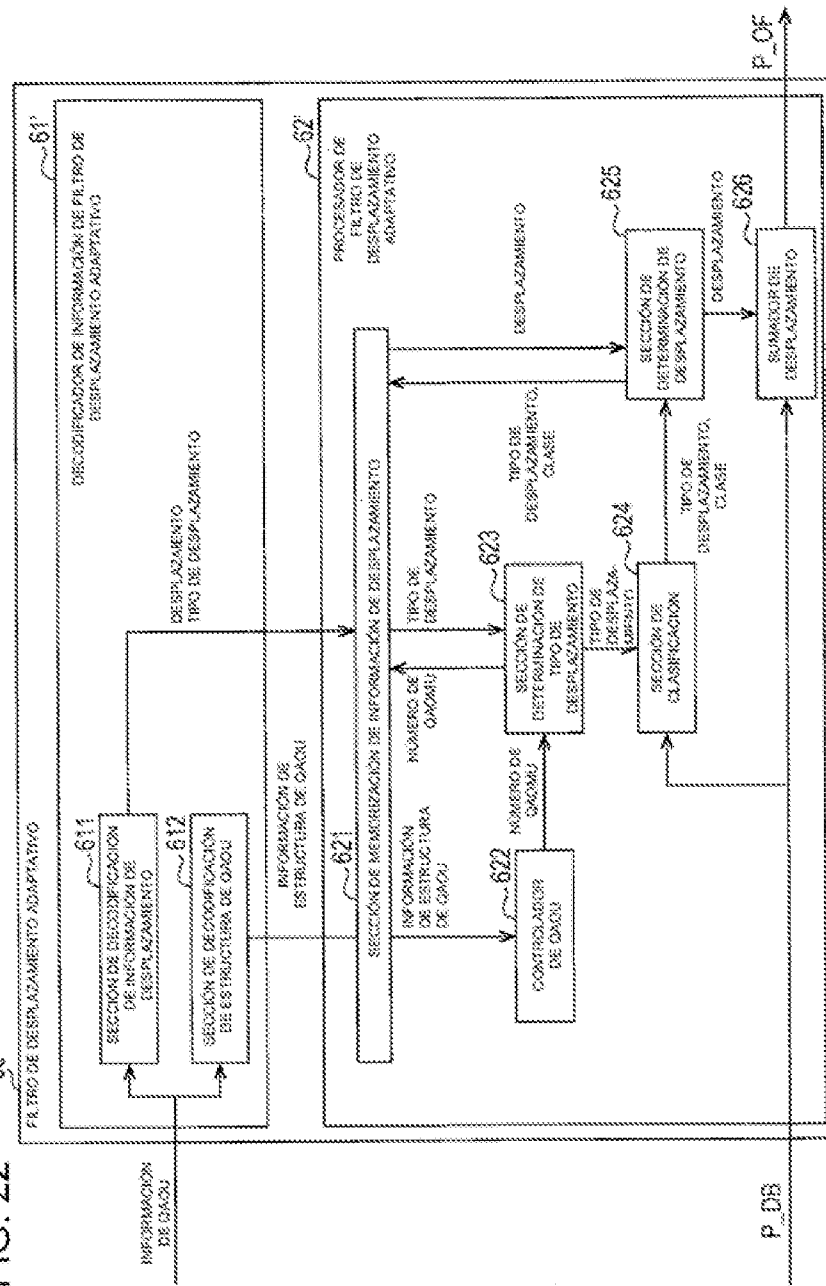


FIG. 23

(a) 301A		
	C	Descriptor
sao_offset_param(xs, ys, sao_curr_depth) {		
if (sao_split_flag[sao_curr_depth][ys][xs]) {		
sao_offset_param(xs + 0, ys + 0, sao_curr_depth + 1)		
sao_offset_param(xs + 1, ys + 0, sao_curr_depth + 1)		
sao_offset_param(xs + 0, ys + 1, sao_curr_depth + 1)		
sao_offset_param(xs + 1, ys + 1, sao_curr_depth + 1)		
} else {		
sao_type_idx[sao_curr_depth][ys][xs]	2	ue(v) ae(v)
curr_num = sao_offset_num[sao_type_idx[sao_curr_depth][ys][xs]		
if (sao_type_idx[sao_curr_depth][ys][xs] != 0) {		
for (i = 0; i < curr_num; i++) {		
sao_offset[sao_curr_depth][ys][xs][i]	2	se(v) ae(v)
}		
}		
}		
}		

(b) 301B		
	C	Descriptor
sao_split_param(xs, ys, sao_curr_depth) {		
if (sao_curr_depth < SaoMaxDepth)		
sao_split_flag[sao_curr_depth][xs][ys]		u(1) ae(v)
else		
sao_split_flag[sao_curr_depth][xs][ys] = 0		
if (sao_split_flag[sao_curr_depth][ys][xs]) {		
sao_split_param(xs + 0, ys + 0, sao_curr_depth + 1)		
sao_split_param(xs + 1, ys + 0, sao_curr_depth + 1)		
sao_split_param(xs + 0, ys + 1, sao_curr_depth + 1)		
sao_split_param(xs + 1, ys + 1, sao_curr_depth + 1)		
}		
}		

FIG. 24

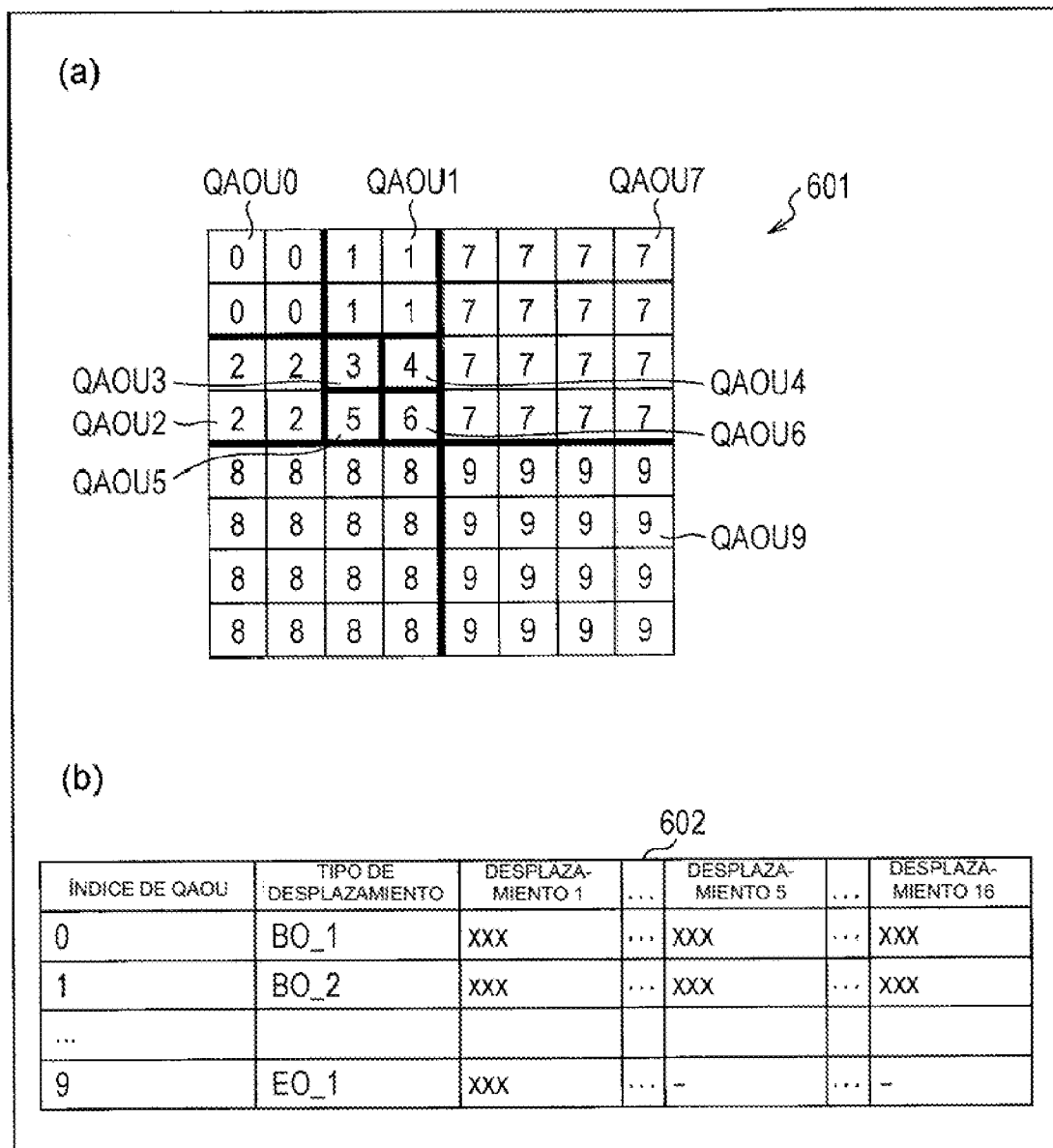


FIG. 25

(a)

801

CÓDIGO	TIPO (sao_type_idx)	TIPO (sao_type_idx)
0	(NO DESPLAZAMIENTO) (0)	(NO DESPLAZAMIENTO) (0)
1	EO_0 (1)	EO_0 (1)
2	EO_1 (2)	BO_0 (5)
3	EO_2 (3)	BO_1 (6)
4	EO_3 (4)	
5	BO_0 (5)	
6	BO_1 (6)	

801A 801B

(b)

802

COD.	TIPO(sao_type_idx)	TIPO (sao_type_idx)	TIPO (sao_type_idx)	TIPO (sao_type_idx)
0	(NO DESPLAZAMIENTO) (0)	(NO DESPLAZAMIENTO) (0)	(NO DESPLAZAMIENTO) (0)	(NO DESPLAZAMIENTO) (0)
1	EO_0 (1)	BO_0 (5)	BO_0 (5)	BO_0 (5)
2	EO_1 (2)	BO_1 (6)		BO_1 (6)
3	EO_2 (3)			EO_0 (1)
4	EO_3 (4)			
5				
6				

802A 802B 802C 802D

FIG. 26

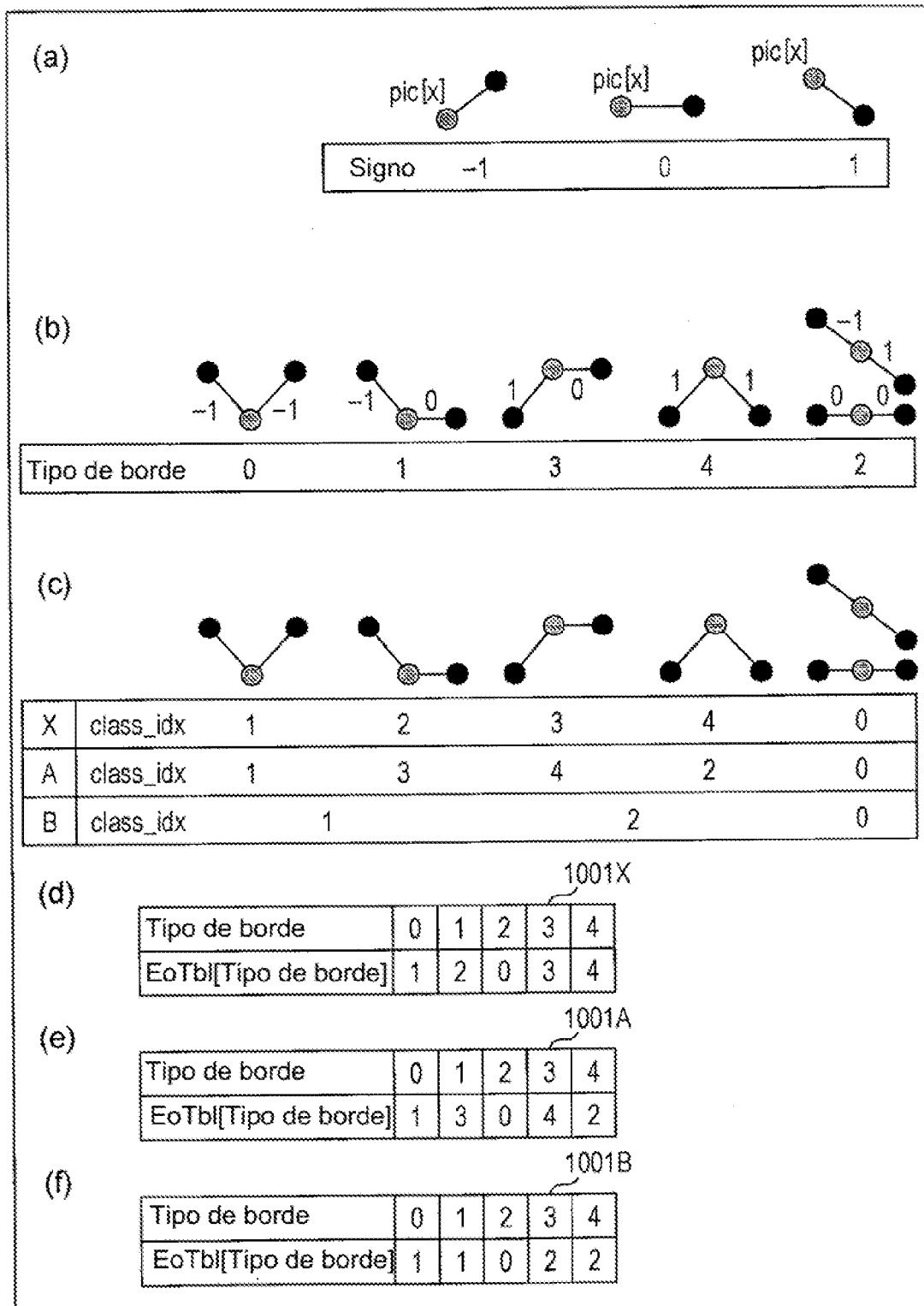


FIG. 27

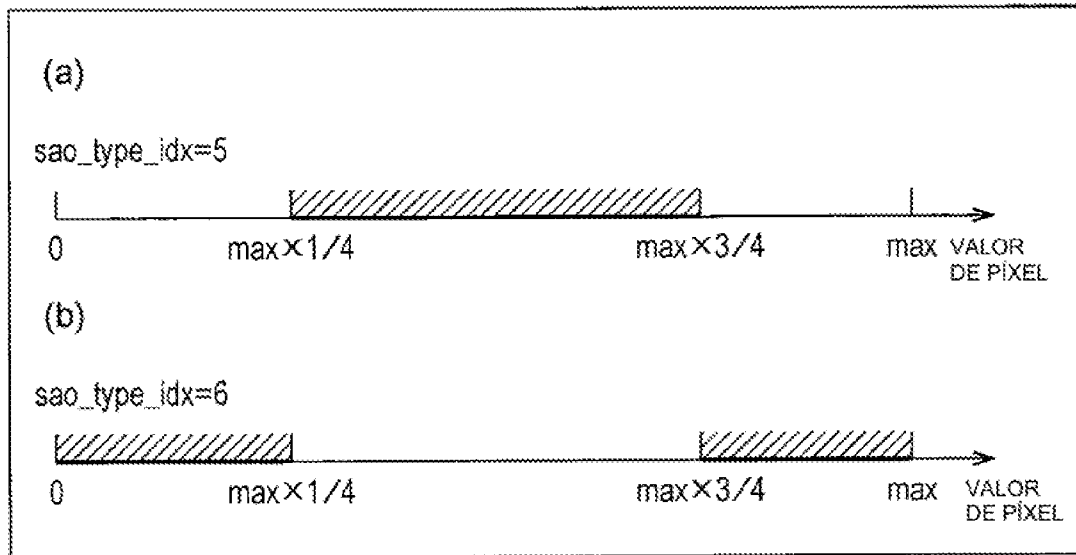


FIG. 28

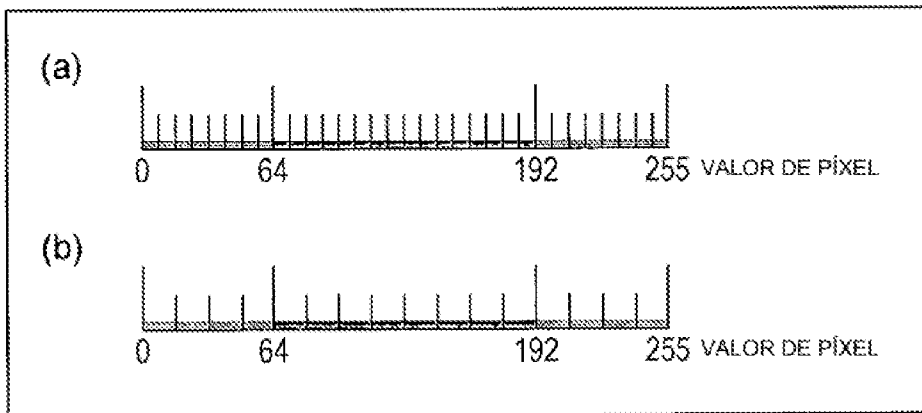


FIG. 29

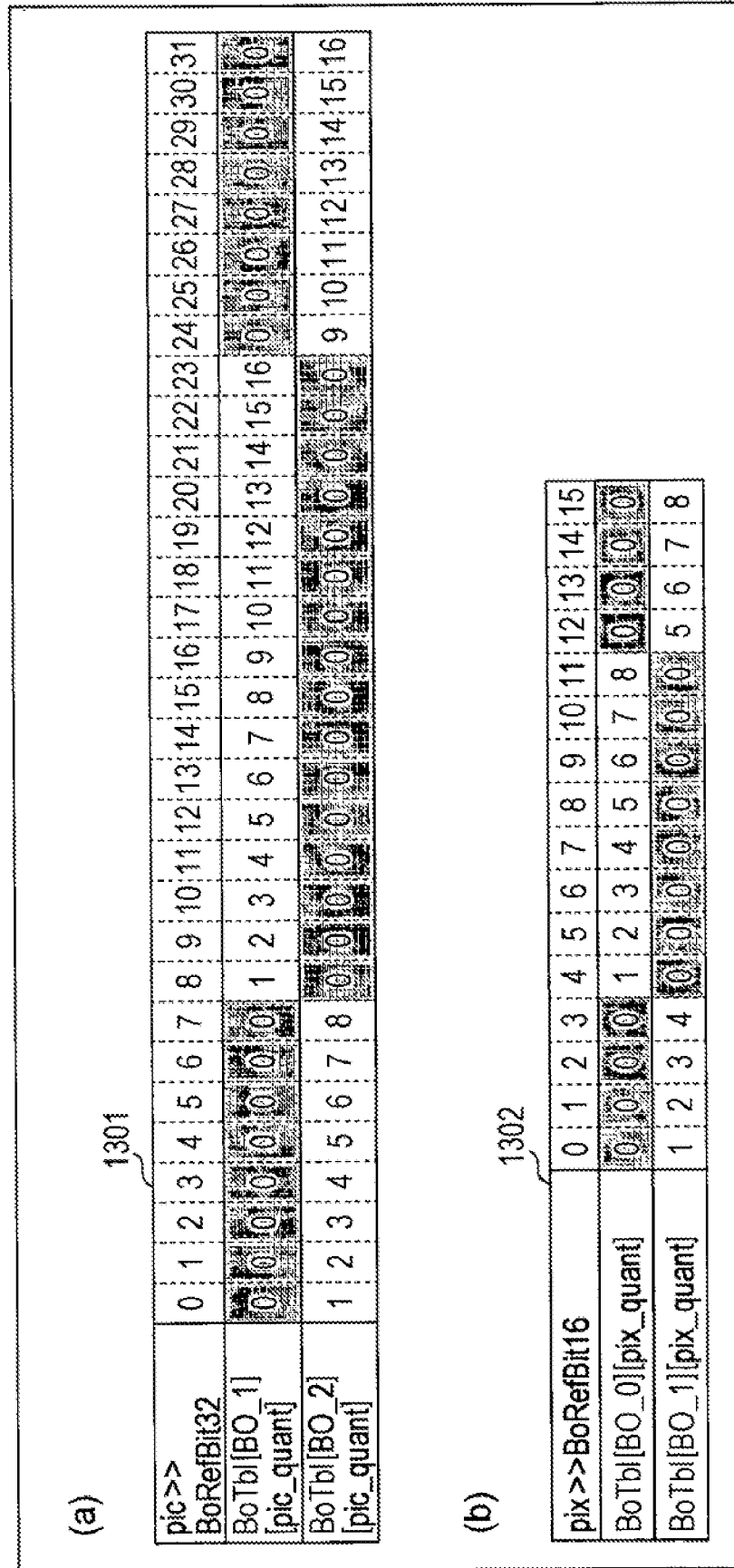


FIG. 30

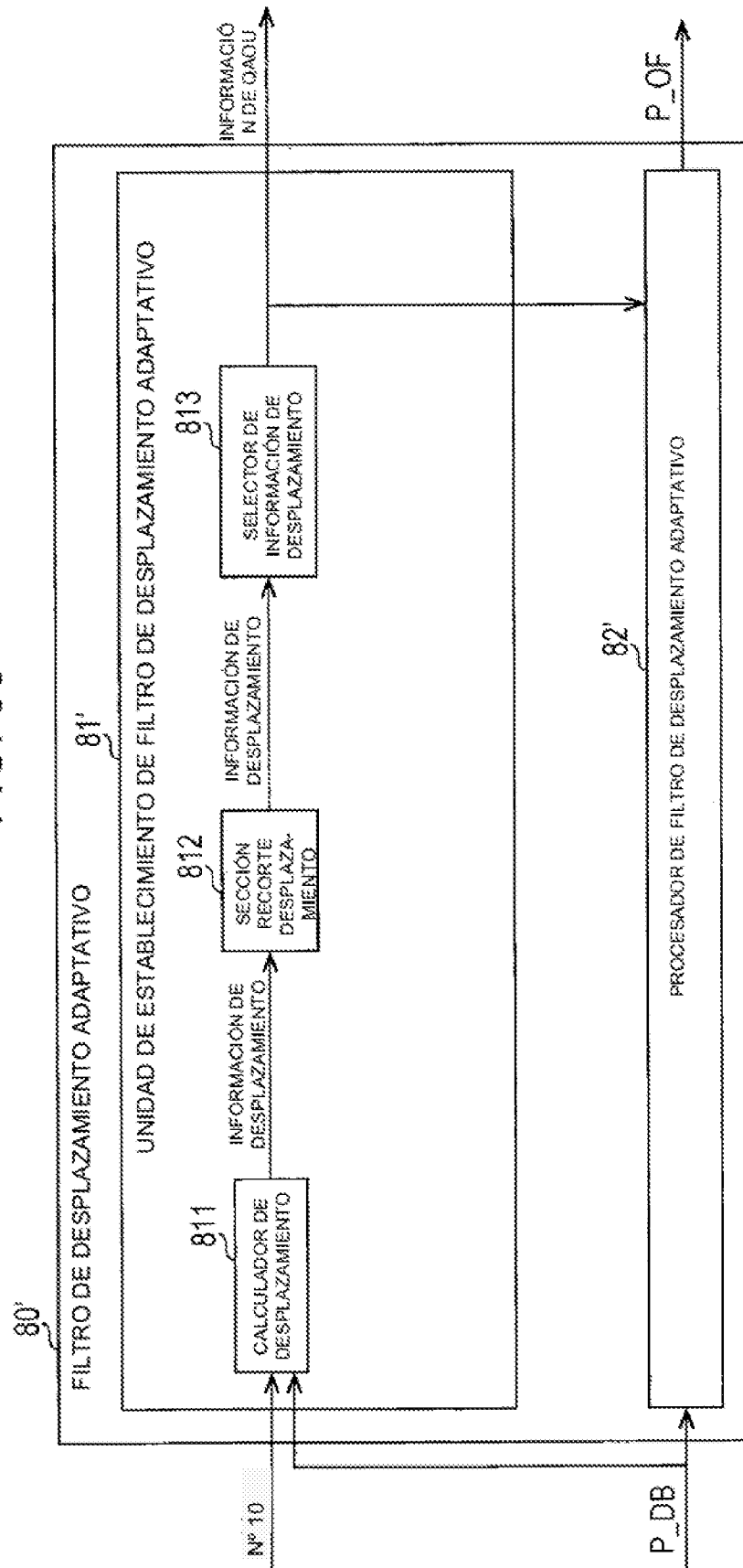


FIG. 31

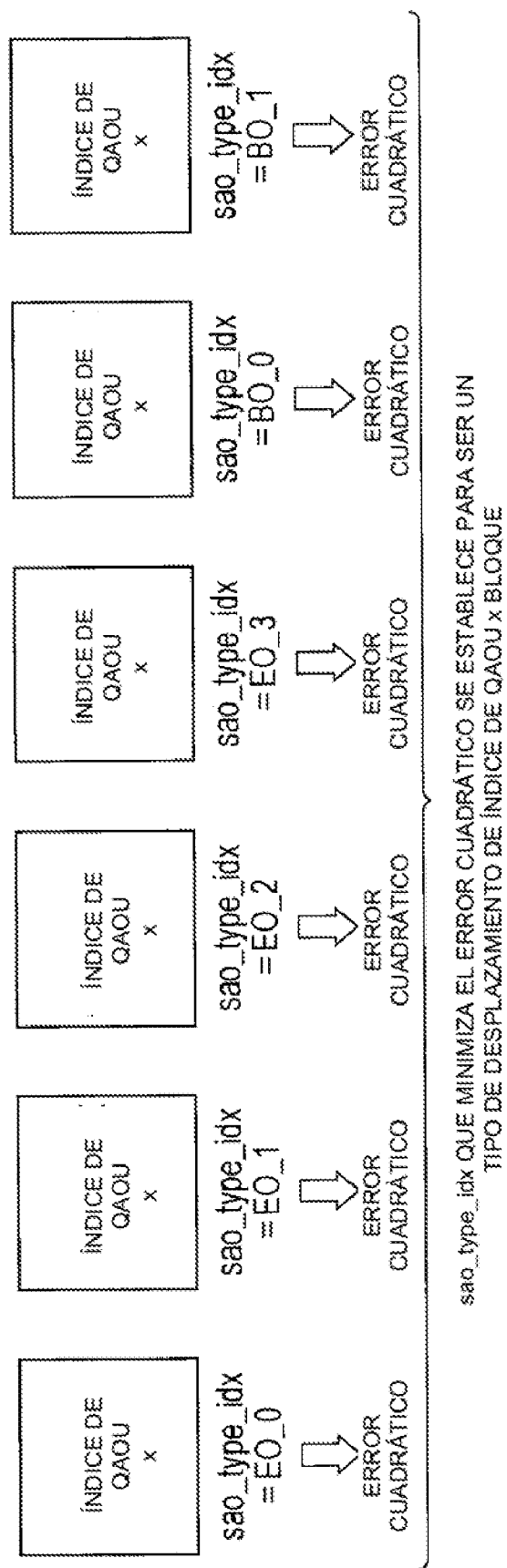


FIG. 32

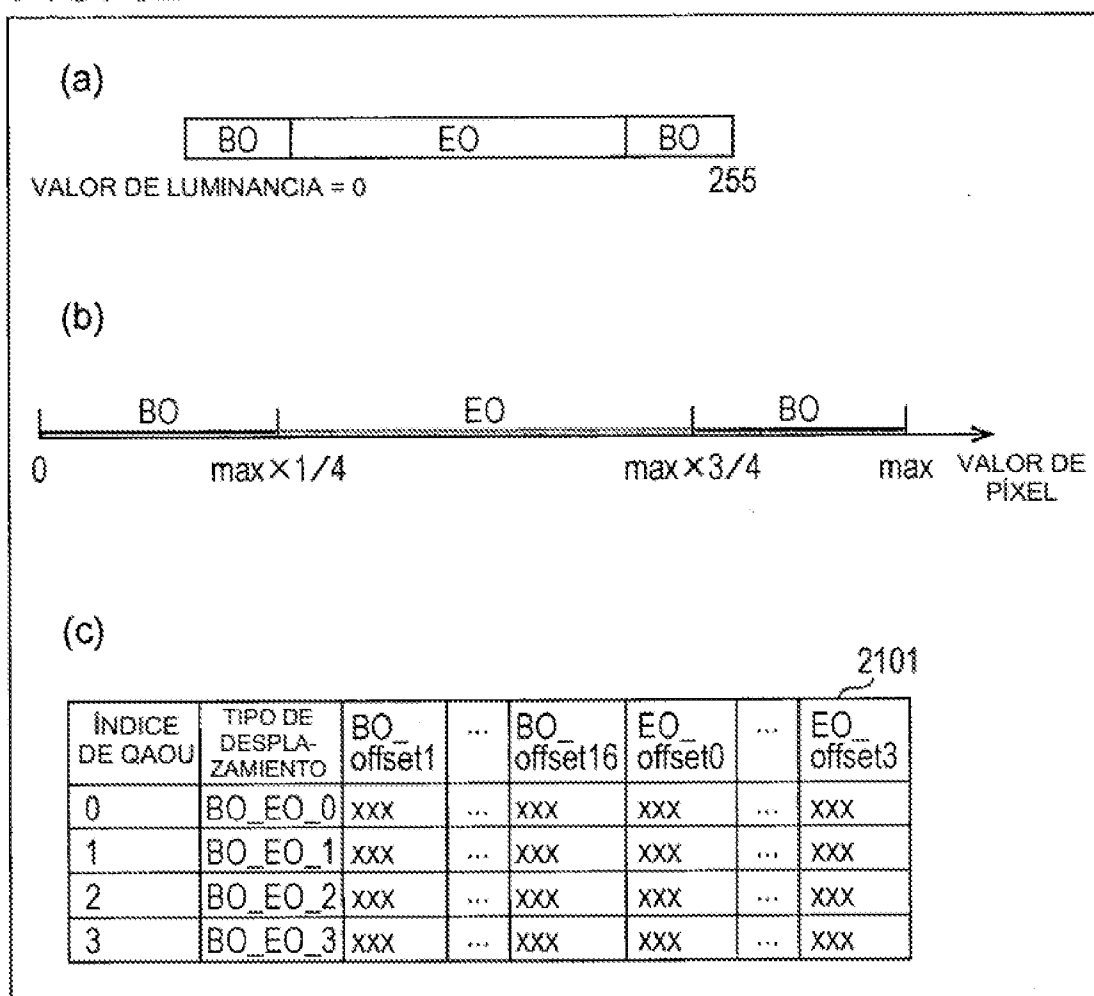


FIG. 33

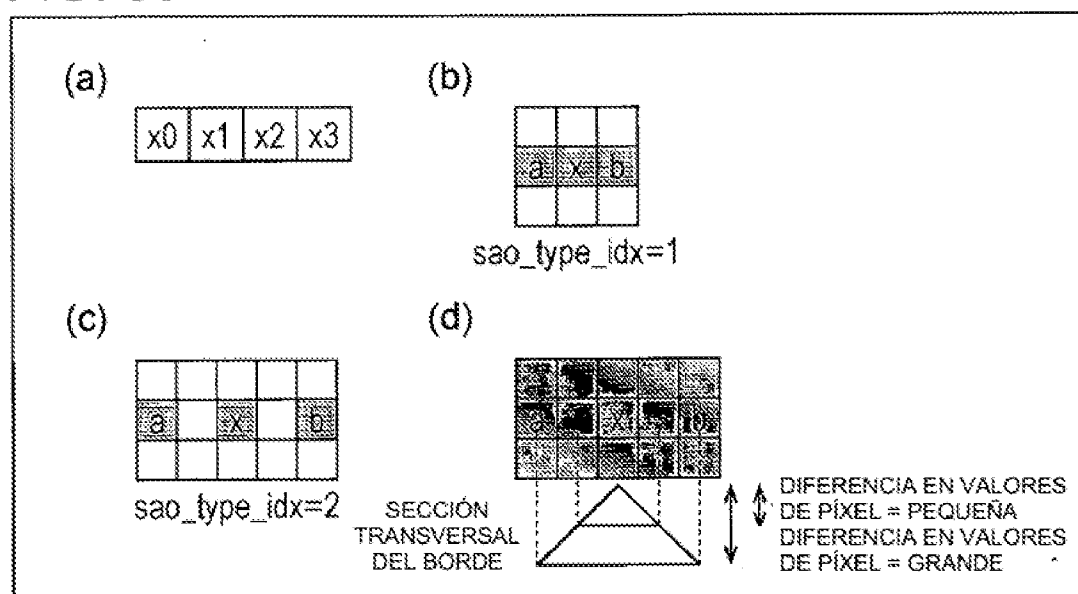


FIG. 34

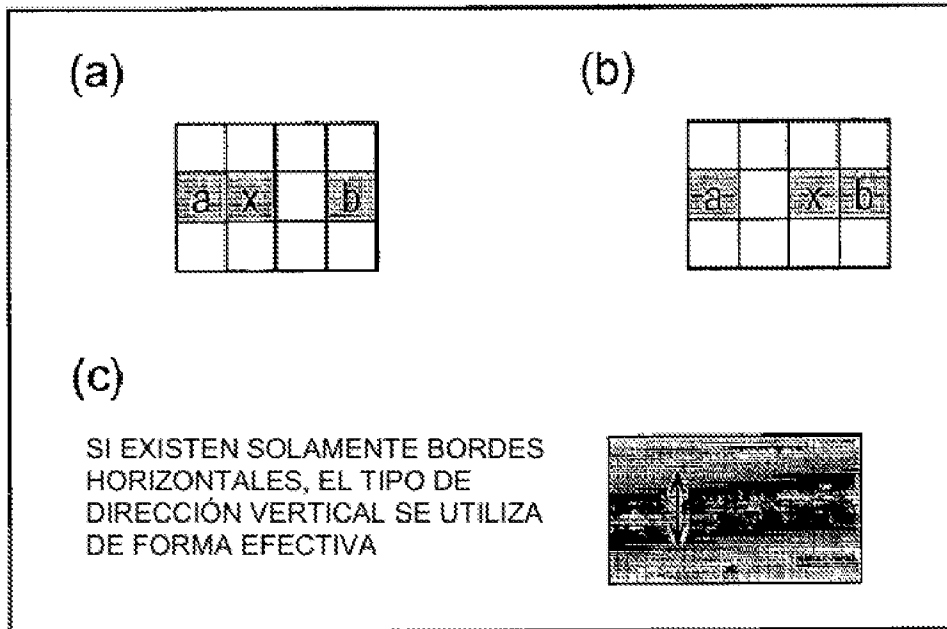


FIG. 35

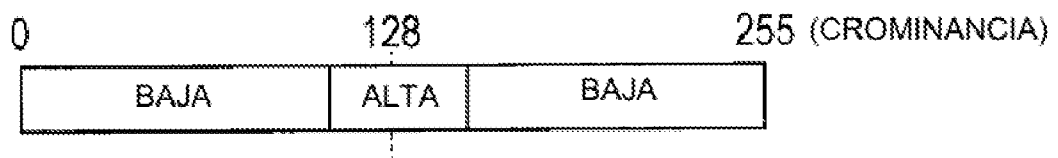


FIG. 36

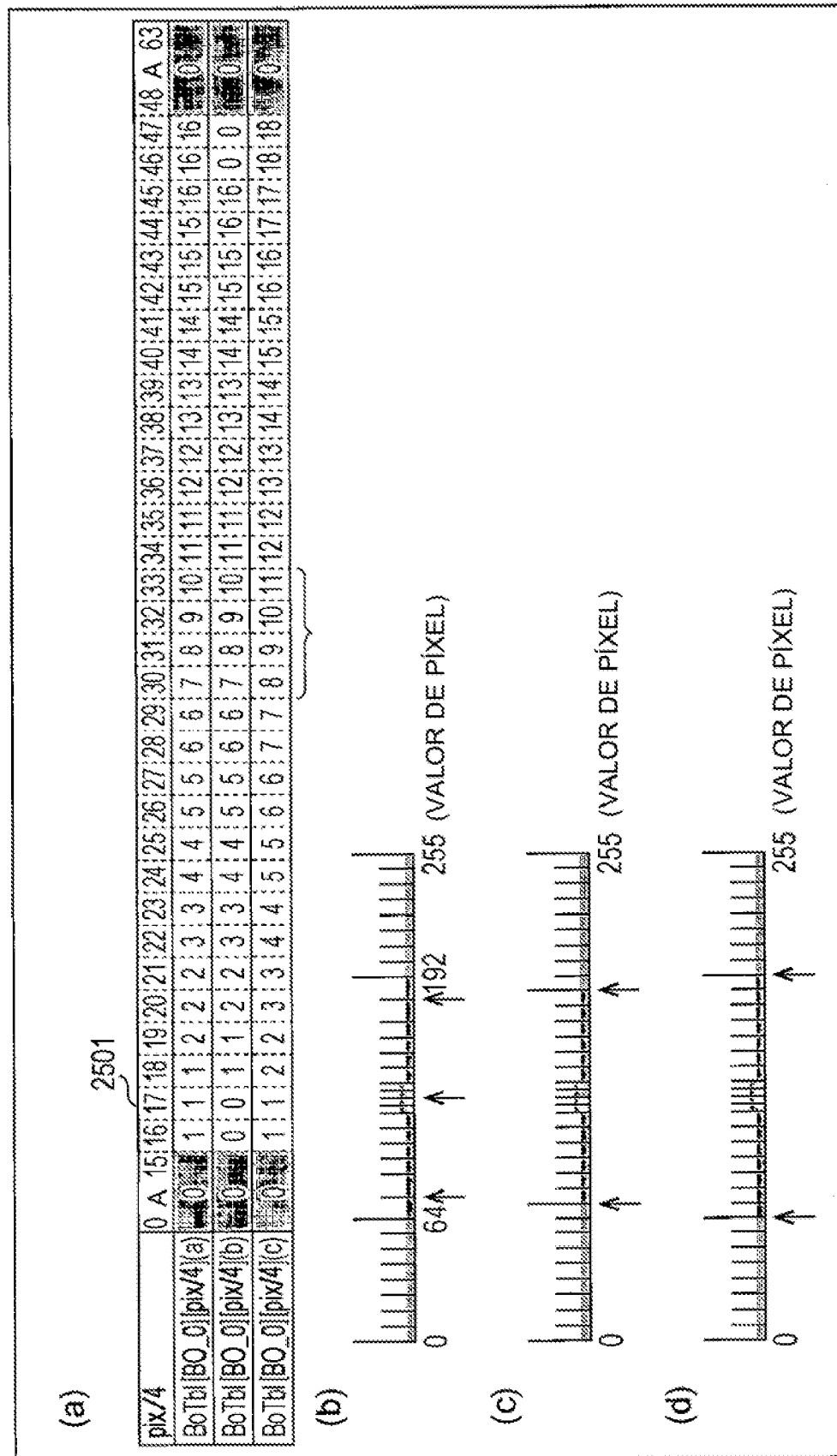


FIG. 37

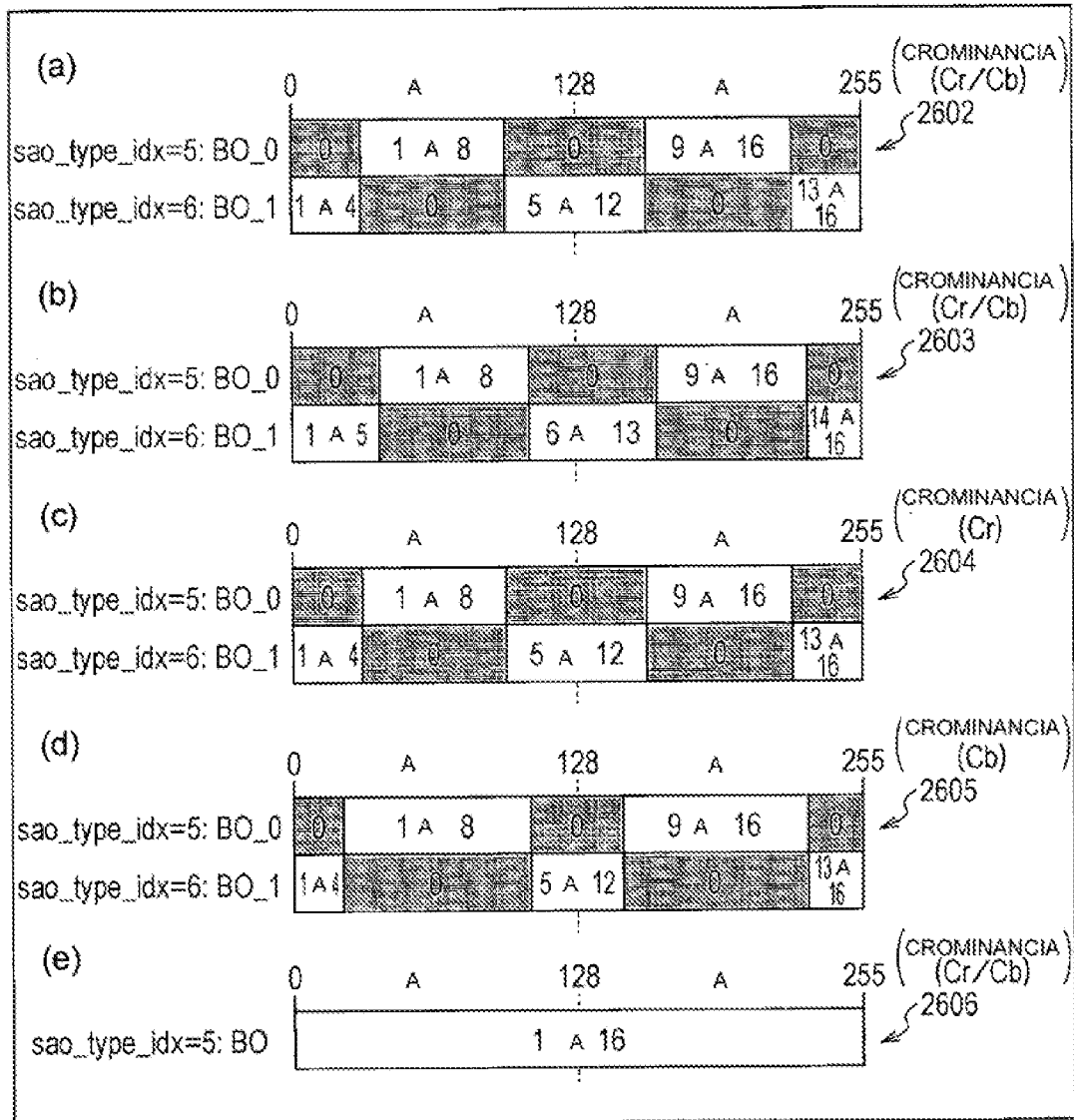


FIG. 38

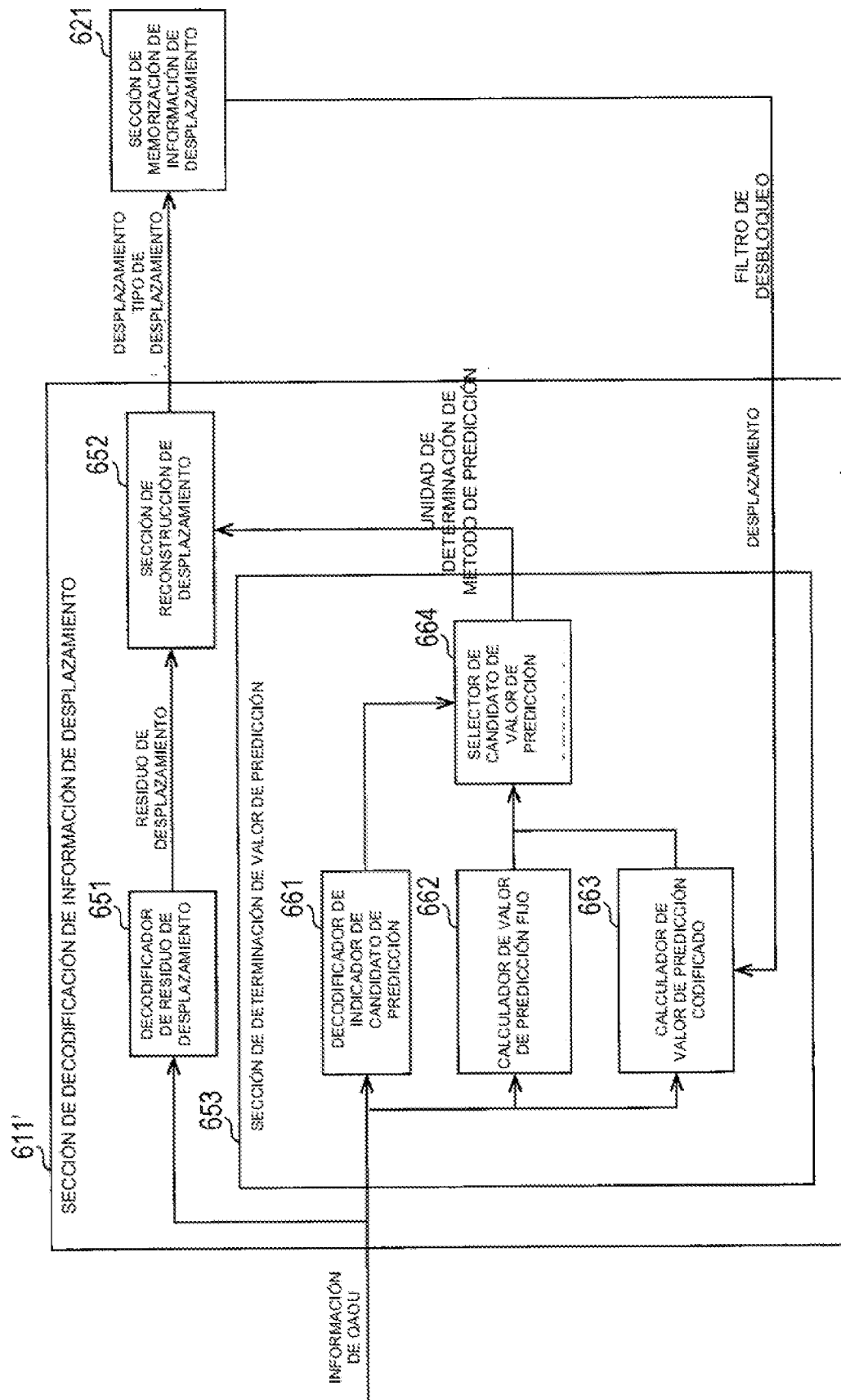


FIG. 39

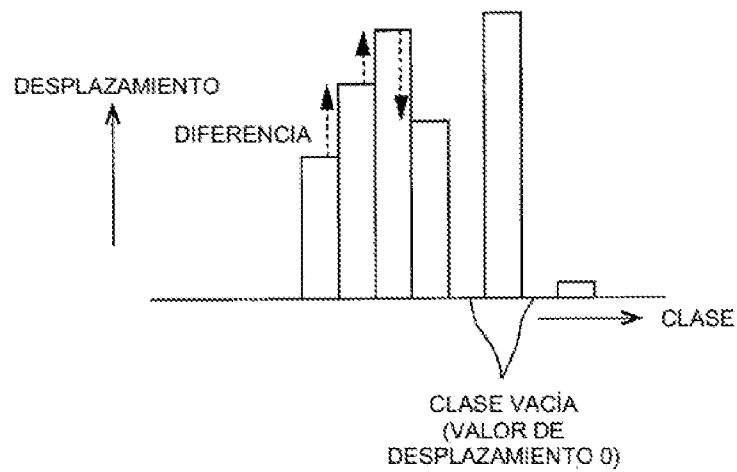


FIG. 40

2901

sao_offset_param(xs, ys, sao_curr_depth) {	C	
if (sao_split_flag [sao_curr_depth][ys][xs]) {		
sao_offset_param(xs + 0, ys + 0, sao_curr_depth + 1)		
sao_offset_param(xs + 1, ys + 0, sao_curr_depth + 1)		
sao_offset_param(xs + 0, ys + 1, sao_curr_depth + 1)		
sao_offset_param(xs + 1, ys + 1, sao_curr_depth + 1)		
} else {		
sao_type_idx [sao_curr_depth][ys][xs]	2	ue(v)
curr_num = sao_offset_num [sao_type_idx [sao_curr_depth][ys][xs]]		
if (sao_type_idx [sao_curr_depth][ys][xs] != 0) {		
for (i = 0; i < curr_num; i++) {		
if (i > 0)		
sao_pred_frag		u(i)
sao_offset_delta [sao_curr_depth][ys][xs][i]	2	se(v)
}		
}		
}		
}		

FIG. 41

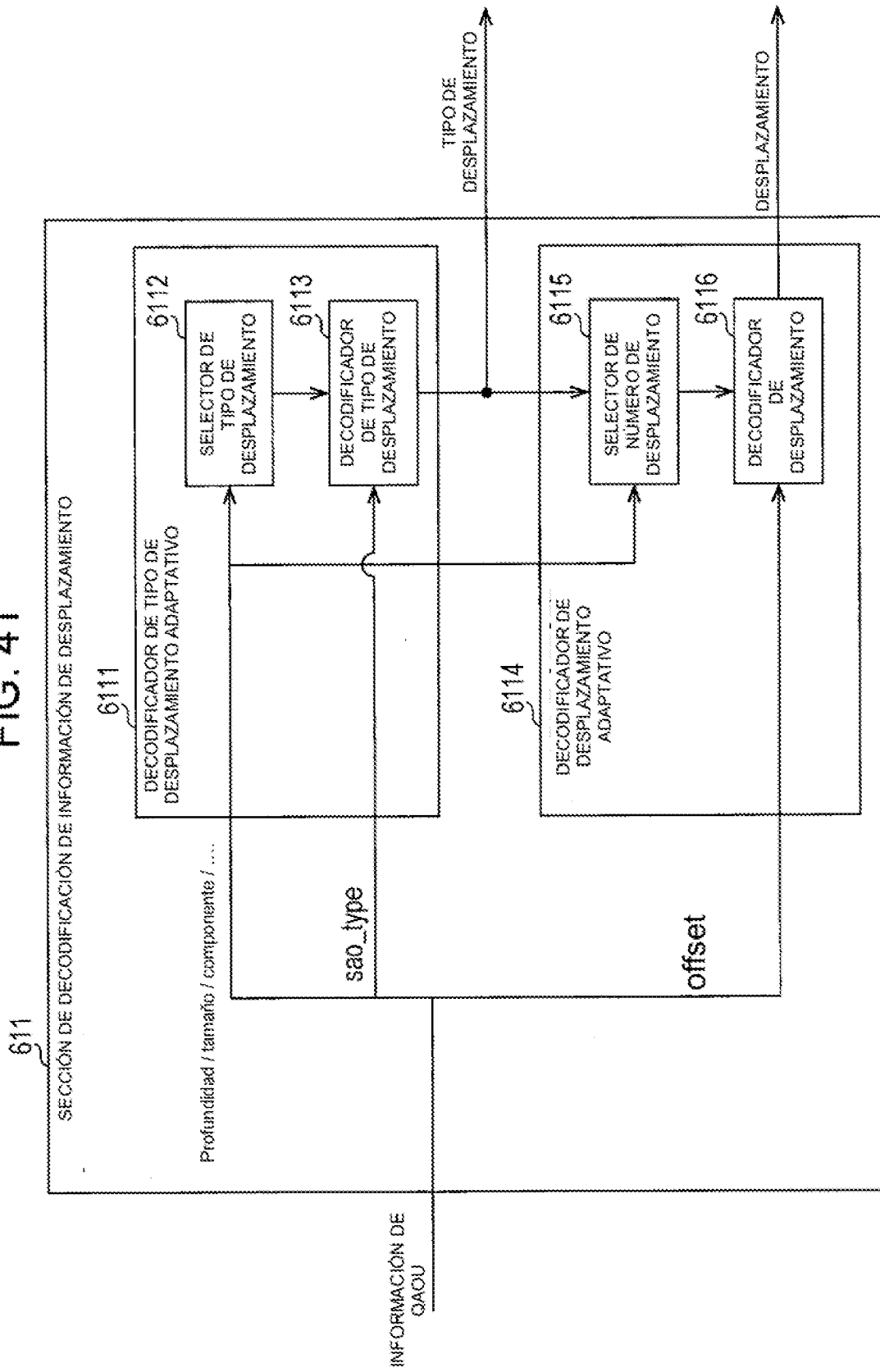


FIG. 42

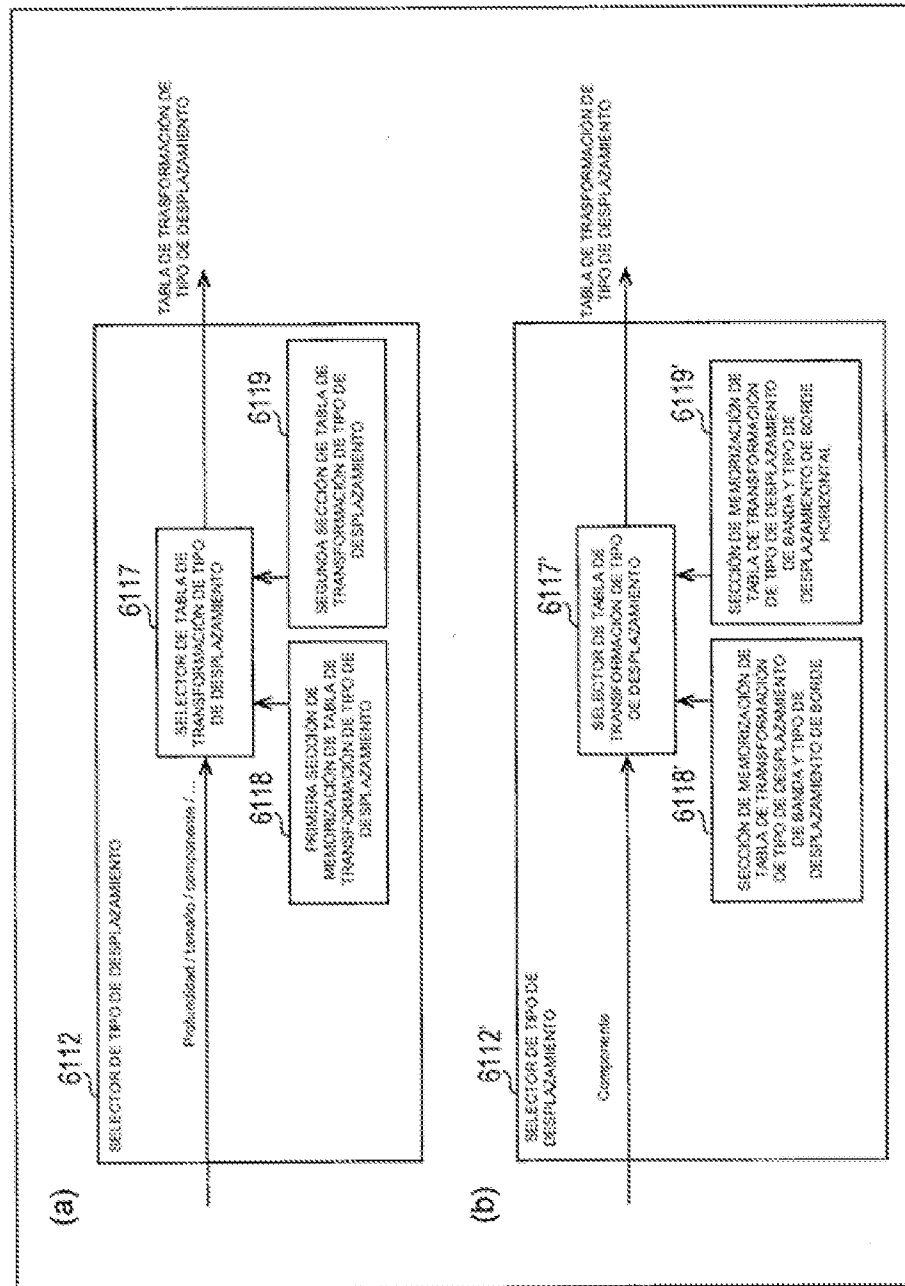


FIG. 43

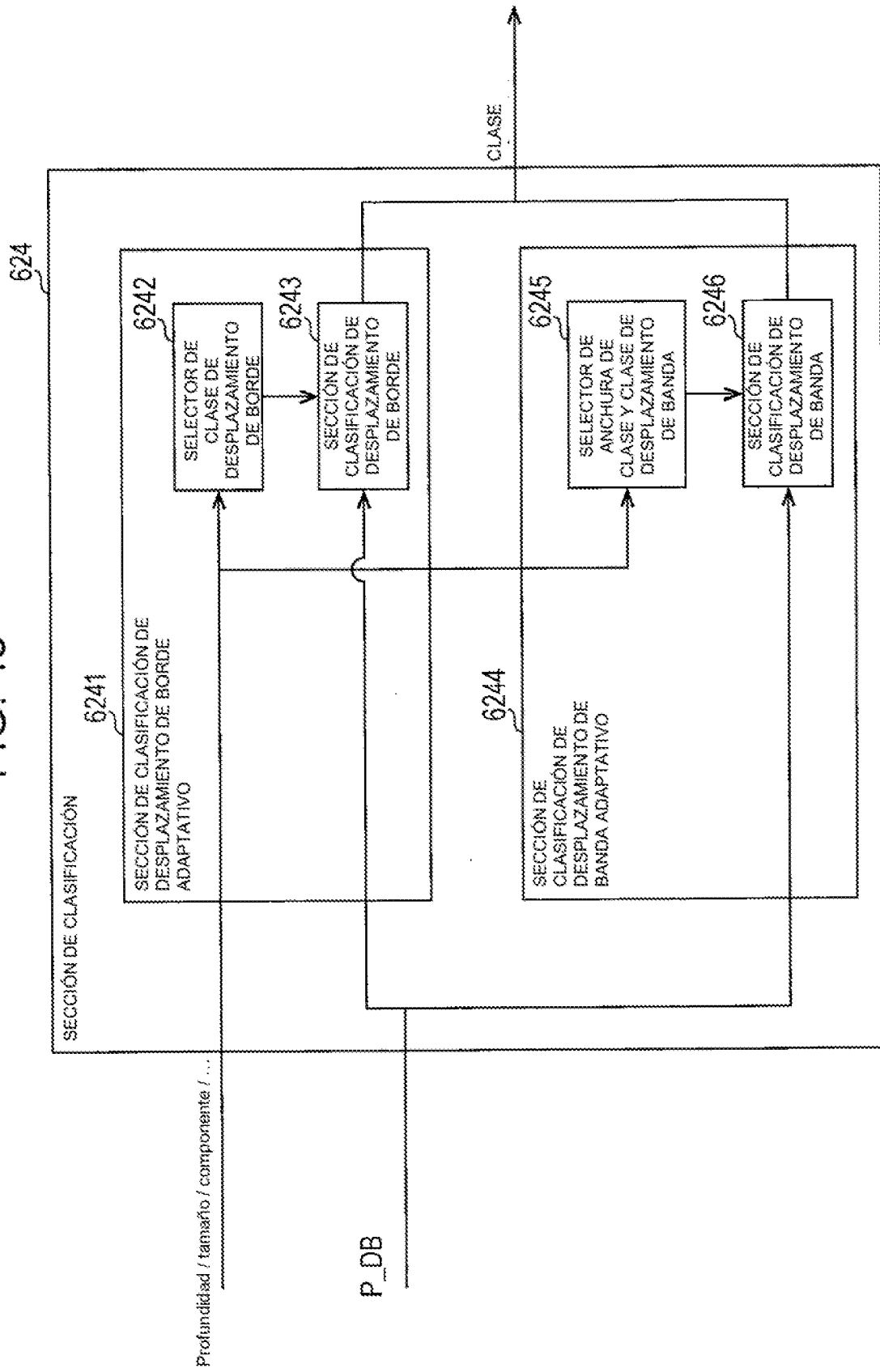


FIG. 44

351A	(a)	sao_offset_param(x, y, sao_curr_depth, component) {	C	Descriptor
		if (sao_split_flag[sao_curr_depth][y][x][component]) {		
		sao_offset_param(x+0, y+0, sao_curr_depth+1, component)		
		sao_offset_param(x+1, y+0, sao_curr_depth+1, component)		
		sao_offset_param(x+0, y+1, sao_curr_depth+1, component)		
		sao_offset_param(x+1, y+1, sao_curr_depth+1, component)		
		else {		
		sao_type_idx[sao_curr_depth][y][x][component]	2	u(v)[ae(v)]
		curr_num = sao_offset_num[sao_type_idx[sao_curr_depth][y][x][component]]		
		if (sao_type_idx[sao_curr_depth][y][x][component] != 0) {		
		for (i = 0; i < curr_num; i++) {		
		sao_offset[sao_curr_depth][y][x][i][component]	2	se(v)[ae(v)]
		}		
		}		
		}		
351B	(b)	sao_split_param(x, y, sao_curr_depth, component) {	C	Descriptor
		if (sao_curr_depth < SaoMaxDepth)		
		sao_split_flag[sao_curr_depth][x][y][component]	2	u(l)[ae(v)]
		else		
		sao_split_flag[sao_curr_depth][x][y][component] = 0		
		if (sao_split_flag[sao_curr_depth][y][x][component]) {		
		sao_split_param(x+0, y+0, sao_curr_depth+1, component)		
		sao_split_param(x+1, y+0, sao_curr_depth+1, component)		
		sao_split_param(x+0, y+1, sao_curr_depth+1, component)		
		sao_split_param(x+1, y+1, sao_curr_depth+1, component)		
		}		
		}		
		}		
		}		
		}		
351C	(c)	sao_param() {	C	Descriptor
		sao_flag	2	u(l)[ae(v)]
		if (sao_flag) {		
		sao_split_param(0, 0, 0, 0)		
		sao_offset_param(0, 0, 0, 0)		
		sao_flag_cb	2	u(l)[ae(v)]
		if (sao_flag_cb) {		
		sao_split_param(0, 0, 0, 1)		
		sao_offset_param(0, 0, 0, 1)		
		}		
		sao_flag_or	2	u(l)[ae(v)]
		if (sao_flag_or) {		
		sao_split_param(0, 0, 0, 2)		
		sao_offset_param(0, 0, 0, 2)		
		}		
		}		

FIG. 45

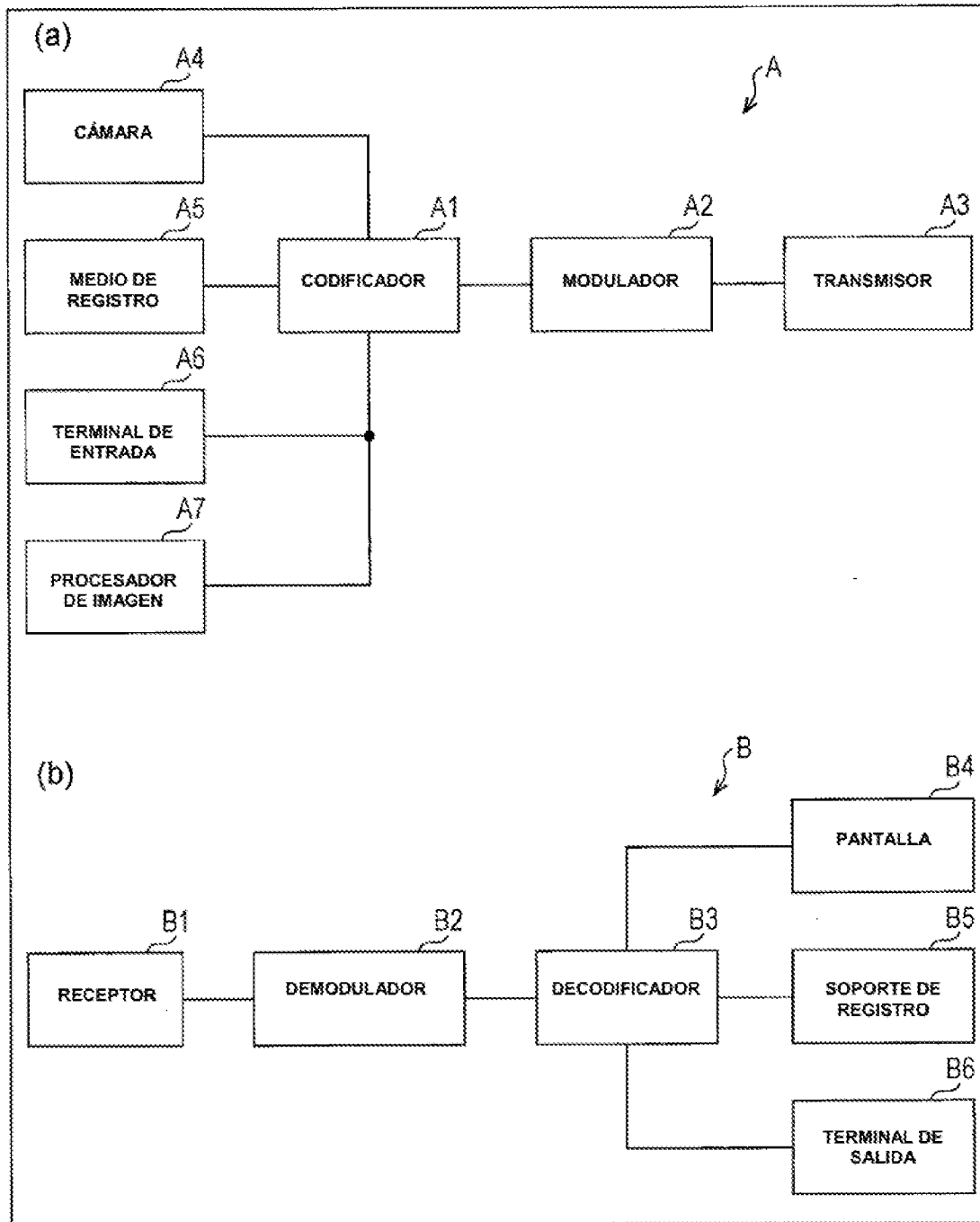


FIG. 46

