

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 784 552**

51 Int. Cl.:

H04N 19/13 (2014.01)

H04N 19/157 (2014.01)

H04N 19/176 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **06.07.2012 PCT/JP2012/004407**

87 Fecha y número de publicación internacional: **17.01.2013 WO13008438**

96 Fecha de presentación y número de la solicitud europea: **06.07.2012 E 12811252 (1)**

97 Fecha y número de publicación de la concesión europea: **19.02.2020 EP 2733941**

54 Título: **Procedimiento de decodificación de imagen, aparato de decodificación de imagen**

30 Prioridad:

11.07.2011 US 201161506430 P

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

28.09.2020

73 Titular/es:

**SUN PATENT TRUST (100.0%)
450 Lexington Avenue, 38th Floor
New York, NY 10017, US**

72 Inventor/es:

**SASAI, HISAO;
NISHI, TAKAHIRO;
SHIBAHARA, YOUJI;
SUGIO, TOSHIYASU;
TANIKAWA, KYOKO y
MATSUNOBU, TORU**

74 Agente/Representante:

GONZÁLEZ PECES, Gustavo Adolfo

ES 2 784 552 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

Procedimiento de decodificación de imagen, aparato de decodificación de imagen

La presente invención se refiere a un procedimiento de decodificación de imagen y a un aparato de decodificación que usa decodificación aritmética.

- 5 Las señales de imagen natural tienen variaciones estadísticas que muestran un comportamiento no estacionario. Uno de los procedimientos de codificación de entropía que usa variaciones estadísticas no estacionarias es la Codificación Aritmética Binaria Adaptativa Basada en Contexto (CABAC) (ver NPL 1). La CABAC se emplea como el estándar ITU-T/ISO/IEC para codificación de vídeo, H.264/AVC.

El significado de los términos usados en el esquema CABAC se describirá más adelante en el presente documento.

- 10 (1) "Adaptativo Basado en el Contexto" significa adaptar los procedimientos de codificación y decodificación a las variaciones estadísticas. En otras palabras, "Adaptativo Basado en el Contexto" significa predecir una probabilidad apropiada como probabilidad de ocurrencia de un símbolo junto con un evento de condiciones circundantes, cuando el símbolo está codificado o decodificado. En la codificación, cuando se determina una probabilidad de ocurrencia $p(x)$ de cada valor de un símbolo S , se aplica una probabilidad de ocurrencia condicional usando un evento real o una secuencia de eventos $F(z)$ como una condición.
- 15 (2) "Binario" significa la representación de un símbolo usando una secuencia binaria. Un símbolo representado por un valor múltiple se asigna una vez a una secuencia binaria denominada "cadena binaria". Una probabilidad predicha (probabilidad condicional) se conmuta y se usa para cada uno de los elementos de secuencia, y la ocurrencia de uno de los eventos de los dos valores se representa mediante una secuencia de bits. Por consiguiente, la probabilidad de un valor se puede gestionar (inicializar y actualizar) usando una unidad (unidad de elemento binario) más pequeña que una unidad de un tipo de señal (ver Fig. 2 y otras de NPL 1).
- 20 (3) "Aritmética" significa que la secuencia de bits se genera no con referencia a las correspondencias en una tabla sino mediante el cálculo. En el esquema de codificación que usa las tablas de códigos de longitud variable tal como H.263, MPEG-4 y H.264, incluso cada valor de un símbolo con una probabilidad de ocurrencia mayor que 0,5 (50 %) debe asociarse con una secuencia binaria (secuencia de bits). De este modo, un valor con la mayor probabilidad debe asociarse con un bit para un símbolo como mínimo. Por el contrario, la codificación aritmética puede representar la ocurrencia de un evento con una probabilidad mayor por un número entero igual o menor que un bit. Cuando (i) hay un tipo de señal en la cual la probabilidad de ocurrencia de tener el primer valor binario como 0 excede 0,9 (90 %) y (ii) un evento que tiene el primer valor binario como 0 ocurre sucesivamente N veces, no es necesario generar datos de 1 bit N veces para cada valor de "0".
- 25 30

Se pueden encontrar ejemplos de determinación de contexto en los NPL 5 a 7.

[Referencias no de patente]

- 35 [NPL 1] Detlev Marpe, y col., "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard", IEEE Transaction on circuits and systems for video technology, Vol. 13, No.7, julio de 2003.
- [NPL 2] Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 5th Meeting: Ginebra, CH, 16-23 de marzo de 2011, JCTVC-E603, ver.7 "WD3: Working Draft 3 of High-Efficiency Video Coding", http://phenix.int-evry.fr/jct/doc_end_user/documents/5_Geneva/wg_11JCTVC-E603-v7.zip
- 40 [NPL 3] Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 4th Meeting: Daegu, KR, 20-28 de enero de 2011, "Common test conditions and software reference configurations", JCTVC-E700
- [NPL 4] Gisle Bjøntegaard, "Improvements of the BD-PSNR model", ITU-T SG16 Q.6 Document, VCEG-A111, Berlín, julio de 2008
- 45 [NPL 5] Sasai H y col.: "Modified Context Derivation for Complexity Reduction", 6. JCT-VC MEETING; 97. Reunión de MPEG; 14-7-2011-22-7-2011; TURÍN; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29 WG11 AND ITU-T SG. 16) de condición de retransmisión, n.º JCTVC F429 1 julio de 2011
- [NPL 6] Chen J y col.: "Simplified context model selection for block level syntax coding", 6. JCT-VC MEETING; 97. Reunión de MPEG; 14-7-2011-22-7-2011; TURÍN; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG. 16) de baliza recibido; n.º JCTVC-F497. 2 de julio de 2011
- 50 [NPL 7] WJ Chien y col.: "Memory and Parsing Friendly CABAC Context", 6. JCT-VC MEETING; 97. Reunión de MPEG; 14-7-2011-22-7-2011; TURÍN; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG. 16) de baliza recibido; n.º JCTVC-F606, 2 de julio de 2011

En tal procedimiento de codificación de imagen y un procedimiento de decodificación de imagen, se desea reducir el uso de memoria (capacidad de memoria a usar).

- 55 En este punto, la presente invención tiene un objeto de proporcionar un procedimiento de codificación de imagen o un procedimiento de decodificación de imagen que pueda reducir el uso de memoria. Esto se consigue mediante las características de las reivindicaciones independientes.

La presente invención puede proporcionar un procedimiento de codificación de imagen o un procedimiento de decodificación de imagen que pueda reducir el uso de memoria.

[Breve descripción de los dibujos]

- 5 [FIG. 1] La FIG. 1 es un diagrama de bloques funcional de un aparato de codificación de imagen de acuerdo con la realización 1.
 [FIG. 2] La FIG. 2 es un diagrama de bloques funcional de una unidad de codificación de longitud variable de acuerdo con la realización 1.
 [FIG. 3] La FIG. 3 es una tabla de un modelo de contexto de un parámetro de control de acuerdo con la realización 1.
- 10 [FIG. 4] La FIG. 4 es un diagrama de flujo que indica un procedimiento de codificación aritmética de acuerdo con la realización 1.
 [FIG. 5] La FIG. 5 es un diagrama de bloques funcional de un aparato de decodificación de imagen de acuerdo con la realización 2.
 [FIG. 6] La FIG. 6 es un diagrama de bloques funcional de una unidad de decodificación de longitud variable de acuerdo con la realización 2.
- 15 [FIG. 7] La FIG. 7 es un diagrama de flujo que indica un procedimiento de decodificación aritmética de acuerdo con la realización 2.
 [FIG. 8] La FIG. 8 es un diagrama de flujo que indica una modificación del procedimiento de decodificación aritmética de acuerdo con la realización 2.
- 20 [FIG. 9A] La FIG. 9A ilustra la información de mapeo y la asignación de índices de contexto de acuerdo con la realización 2.
 [FIG. 9B] La FIG. 9B ilustra bloques divididos (una estructura de árbol) de acuerdo con HEVC de acuerdo con la realización 2.
 [FIG. 10] La FIG. 10 ilustra una estructura de bloques multicapa de acuerdo con la realización 2.
- 25 [FIG. 11] La FIG. 11 ilustra un procedimiento de decodificación aritmética para split_coding_unit_flag de acuerdo con la realización 3. [FIG. 12A] La FIG. 12A es una tabla que indica el resultado de la verificación en split_coding_unit_flag de acuerdo con la realización 3.
 [FIG. 12B] La FIG. 12B es una tabla que indica el resultado de la verificación en split_coding_unit_flag de acuerdo con la realización 3.
- 30 [FIG. 13] La FIG. 13 ilustra un procedimiento de decodificación aritmética para skip_flag de acuerdo con la realización 3.
 [FIG. 14A] La FIG. 14A es una tabla que indica el resultado de la verificación en skip_flag de acuerdo con la realización 3.
 [FIG. 14B] La FIG. 14B es una tabla que indica el resultado de la verificación en skip_flag de acuerdo con la realización 3.
- 35 [FIG. 15] La FIG. 15 es una tabla que indica un procedimiento de decodificación aritmética para merge_flag de acuerdo con la realización 3.
 [FIG. 16A] La FIG. 16A es una tabla que indica el resultado de la verificación en merge_flag de acuerdo con la realización 3.
 [FIG. 16B] La FIG. 16B es una tabla que indica el resultado de la verificación en merge_flag de acuerdo con la realización 3.
- 40 [FIG. 17] La FIG. 17 es una tabla que indica un procedimiento de decodificación aritmética para ref_idx de acuerdo con la realización 3.
 [FIG. 18A] La FIG. 18A es una tabla que indica el resultado de la verificación en ref_idx de acuerdo con la realización 3.
 [FIG. 18B] La FIG. 18B es una tabla que indica el resultado de la verificación en ref_idx de acuerdo con la realización 3.
- 45 [FIG. 18C] La FIG. 18C es una tabla que indica un modelo de contexto para ref_idx de acuerdo con la realización 3.
 [FIG. 18D] La FIG. 18D es una tabla que indica un modelo de contexto para ref_idx de acuerdo con la realización 3.
- 50 [FIG. 19] La FIG. 19 es una tabla que indica un procedimiento de decodificación aritmética para inter_pred_flag de acuerdo con la realización 3.
 [FIG. 20A] La FIG. 20A es una tabla que indica el resultado de la verificación en inter_pred_flag de acuerdo con la realización 3.
- 55 [FIG. 20B] La FIG. 20B es una tabla que indica el resultado de la verificación en inter_pred_flag de acuerdo con la realización 3.
 [FIG. 21] La FIG. 21 es una tabla que indica un procedimiento de decodificación aritmética para mvd de acuerdo con la realización 3.
- 60 [FIG. 22A] La FIG. 22A es una tabla que indica el resultado de la verificación en mvd de acuerdo con la realización 3.
 [FIG. 22B] La FIG. 22B es una tabla que indica el resultado de la verificación en mvd de acuerdo con la realización 3.
 [FIG. 22C] La FIG. 22C es una tabla que indica un modelo de contexto para mvd de acuerdo con la realización 3.

[FIG. 22D] La FIG. 22D es una tabla que indica un modelo de contexto para mvd de acuerdo con la realización 3.
 [FIG. 23A] La FIG. 23A es una tabla que indica un procedimiento de decodificación aritmética para no_residual_data_flag de acuerdo con la realización 3.
 [FIG. 23B] La FIG. 23B es una tabla que indica una sintaxis para no_residual_data_flag de acuerdo con la
 5 realización 3.
 [FIG. 24A] La FIG. 24A es una tabla que indica el resultado de la verificación en no_residual_data_flag de acuerdo con la realización 3.
 [FIG. 24B] La FIG. 24B es una tabla que indica el resultado de la verificación en no_residual_data_flag de acuerdo con la realización 3.
 [FIG. 25A] La FIG. 25A es una tabla que indica un procedimiento de decodificación aritmética para intra_chroma_pred_mode de acuerdo con la realización 3.
 [FIG. 25B] La FIG. 25B es una tabla que indica un procedimiento para determinar IntraPredModeC basado en intra_chroma_pred_mode, de acuerdo con la realización 3.
 [FIG. 26A] La FIG. 26A es una tabla que indica el resultado de la verificación en intra_chroma_pred_mode de
 10 acuerdo con la realización 3.
 [FIG. 26B] La FIG. 26B es una tabla que indica el resultado de la verificación en intra_chroma_pred_mode de acuerdo con la realización 3.
 [FIG. 27] La FIG. 27 es una tabla que indica un procedimiento de decodificación aritmética para cbf_luma(cr,cb) de acuerdo con la realización 3.
 [FIG. 28A] La FIG. 28A es una tabla que indica el resultado de la verificación en cbf_luma(cr,cb) de acuerdo con la
 20 realización 3.
 [FIG. 28B] La FIG. 28B es una tabla que indica un resultado de la verificación en cbf_luma (cr, cb) de acuerdo con la realización 3.
 [FIG. 29A] La FIG. 29A es un gráfico que indica el resultado de la verificación de acuerdo con la realización 3.
 [FIG. 29B] La FIG. 29B es un gráfico que indica el resultado de la verificación de acuerdo con la realización 3.
 [FIG. 30A] La FIG. 30A es un gráfico que indica el resultado de la verificación de acuerdo con la realización 3.
 [FIG. 30B] La FIG. 30B es un gráfico que indica el resultado de la verificación de acuerdo con la realización 3.
 [FIG. 31A] La FIG. 31A es un gráfico que indica el resultado de la verificación de acuerdo con la realización 3.
 [FIG. 31B] La FIG. 31B es un gráfico que indica el resultado de la verificación de acuerdo con la realización 3.
 [FIG. 32A] La FIG. 32A es un gráfico que indica el resultado de la verificación de acuerdo con la realización 3.
 [FIG. 32B] La FIG. 32B es un gráfico que indica el resultado de la verificación de acuerdo con la realización 3.
 [FIG. 33A] La FIG. 33A es una tabla que indica un ejemplo de un conjunto de parámetros de acuerdo con la
 25 realización 3.
 [FIG. 33B] La FIG. 33B es una tabla que indica un resultado de la verificación de acuerdo con la realización 3.
 [FIG. 34A] La FIG. 34A es una tabla que indica un ejemplo de un conjunto de parámetros de acuerdo con la
 30 realización 3.
 [FIG. 34B] La FIG. 34B es una tabla que indica un resultado de la verificación de acuerdo con la realización 3.
 [FIG. 35] La FIG. 35 es un diagrama de flujo que indica un procedimiento de codificación aritmética de acuerdo con la realización 3.
 [FIG. 36] La FIG. 36 es un diagrama de flujo que indica un procedimiento de decodificación aritmética de acuerdo
 40 con la realización 3.
 [FIG. 37] La FIG. 37 ilustra modelos de contexto usando valores de parámetros de control correspondientes a dos bloques vecinos de acuerdo con las realizaciones.
 [FIG. 38] La FIG. 38 ilustra el aumento en el uso de memoria cuando se usa un bloque superior de acuerdo con las realizaciones. [FIG. 39] La FIG. 39 ilustra una configuración global de un sistema de suministro de contenido para implementar servicios de distribución de contenido.
 [FIG. 40] La FIG. 40 ilustra una configuración global de un sistema de difusión digital.
 [FIG. 41] La FIG. 41 muestra un diagrama de bloques que ilustra un ejemplo de una configuración de una televisión.
 [FIG. 42] La FIG. 42 que ilustra un diagrama de bloques que ilustra un ejemplo de una configuración de una unidad
 50 de reproducción/grabación de información que lee y escribe información desde y en un medio de grabación que es un disco óptico.
 [FIG. 43] La FIG. 43 ilustra un ejemplo de una configuración de un medio de grabación que es un disco óptico.
 [FIG. 44A] La FIG. 44A ilustra un ejemplo de un teléfono celular.
 [FIG. 44B] La FIG. 44B ilustra un diagrama de bloques que muestra un ejemplo de una configuración de un teléfono
 55 celular.
 [FIG. 45] La FIG. 45 ilustra una estructura de datos multiplexados.
 [FIG. 46] La FIG. 46 ilustra esquemáticamente cómo se multiplexa cada flujo en datos multiplexados.
 [FIG. 47] La FIG. 47 ilustra cómo se almacena un flujo de vídeo en un flujo de paquetes PES en más detalle.
 [FIG. 48] La FIG. 48 ilustra una estructura de paquetes TS y paquetes fuente en los datos multiplexados.
 [FIG. 49] La FIG. 49 ilustra una estructura de datos de una PMT.
 [FIG. 50] La FIG. 50 ilustra una estructura interna de información de datos multiplexados.
 [FIG. 51] La FIG. 51 ilustra una estructura interna de información de atributo de flujo.
 [FIG. 52] La FIG. 52 ilustra etapas para identificar datos de vídeo.
 [FIG. 53] La FIG. 53 es un diagrama de bloques que ilustra un ejemplo de una configuración de un circuito integrado para implementar el procedimiento de codificación de imagen en movimiento y el procedimiento de decodificación
 65 de imagen en movimiento de acuerdo con cada una de las realizaciones.

[FIG. 54] La FIG. 54 ilustra una configuración para conmutar entre frecuencias de accionamiento.

[FIG. 55] La FIG. 55 ilustra etapas para identificar datos de vídeo y conmutación entre frecuencias de accionamiento.

5 [FIG. 56] La FIG. 56 ilustra un ejemplo de una tabla de búsqueda en la que los estándares de datos de vídeo están asociados con las frecuencias de accionamiento.

[FIG. 57A] La FIG. 57A ilustra un ejemplo de una configuración para compartir un módulo de una unidad de procesamiento de señal.

[FIG. 57B] La FIG. 57B ilustra otro ejemplo de una configuración para compartir un módulo de la unidad de procesamiento de señal.

10 (Conocimiento en el que se basa la presente invención)

Los presentes inventores han encontrado los siguientes problemas.

En la Codificación de Vídeo de Alta Eficiencia (HEVC) que es un esquema de codificación de vídeo de siguiente generación, se está estudiando el modelo de contexto en la codificación y decodificación de diversos parámetros de control (NPL 2). El parámetro de control se incluye en un flujo de bits codificado y es un parámetro (bandera, etc.) usado en el procesamiento de codificación o decodificación. Más específicamente, el parámetro de control es un elemento de sintaxis.

20 El modelo de contexto es información que indica (i) qué condición se considera para (ii) una señal de qué unidad (cada elemento de un multivalor, un valor binario, una secuencia binaria (cadena binaria). En este punto, "qué condición" indica qué condición con el número de elementos condicionales se aplica o qué tipo de señal de un parámetro de control se considera apropiada. Como las condiciones se dividen en categorías más pequeñas, es decir, como el número de condiciones τ aumenta, disminuye el número de casos que son válidos para las condiciones. Como resultado, ya que el número de entrenamientos disminuye, la precisión de la probabilidad predicha disminuye (por ejemplo, ver "efecto de dilución" en NPL 1).

25 Adicionalmente, la disminución en el número de condiciones indica que no se considera un contexto (condiciones circundantes) y que no se adapta a las variaciones estadísticas.

Al diseñar un modelo de contexto, después de determinar una directriz para diseñar el modelo, es necesario considerar la validez del modelo realizando verificaciones especializadas para una imagen, tales como las verificaciones de variaciones estadísticas en los detalles de una imagen y en el parámetro de control para controlar la codificación y decodificación de una imagen.

30 En H.264, el uso de eventos avanzados de un número limitado para codificar un símbolo es un criterio de una regla, y los modelos de contexto se clasifican en cuatro tipos básicos de diseño.

El primer y segundo tipo se refieren a la codificación y decodificación de un parámetro de control.

35 El primer modelo de contexto usa valores codificados de hasta dos valores codificados vecinos (ver NPL 1). Aunque la definición de los dos valores codificados vecinos depende de cada tipo de señal, normalmente, se usan los valores de los parámetros de control correspondientes incluidos en los bloques vecinos a la izquierda y la parte superior del bloque actual.

El segundo tipo de modelos de contexto es un tipo para determinar un contexto basado en un árbol binario como probabilidad de ocurrencia. Más específicamente, el segundo tipo de modelos de contexto se aplica a los parámetros de control `mb_type` y `sub_mb_type`.

40 El tercer y cuarto tipo de modelos de contexto se refieren a la codificación y decodificación de valores residuales (datos residuales), tal como datos de imagen. El tercer tipo usa solo los valores codificados o decodificados anteriores en el orden de exploración de los coeficientes de frecuencia (o coeficientes cuantizados). El cuarto tipo determina un contexto de acuerdo con los valores decodificados y acumulados (niveles).

45 Las ventajas del principio de diseño y la implementación del modelo de transición de probabilidad en H.264, tal como el primer tipo, han sido estudiados durante mucho tiempo y se aplicarán a HEVC que se está estudiando (ver NPL 2). Por ejemplo, el primer tipo de contexto que usa elementos de sintaxis vecinos se está estudiando para usarse para los parámetros de control `alf_cu_flag`, `split_coding_unit_flag`, `skip_flag`, `merge_flag`, `intra_chroma_pred_mode`, `inter_pred_flag`, `ref_idx_lc`, `ref_idx_l0`, `ref_idx_l1`, `mvd_l0`, `mvd_l1`, `mvd_lc`, `no_residual_data_flag`, `cbf_luma`, `cbf_cb`, y `cbf_cr` (ver 9.3.3.1.1 de NPL 2).

50 Sin embargo, los presentes inventores han encontrado que existe un problema en el uso de la memoria al codificar usando el "modelo de contexto que usa los dos bloques vecinos" del primer tipo.

La FIG. 37 ilustra modelos de contexto usando valores de parámetros de control correspondientes a los dos bloques vecinos. Adicionalmente, La FIG. 37 ilustra los modelos de contexto usando los bloques vecinos en H. 264.

El bloque C en la FIG. 37 incluye un valor de un parámetro de control SE a codificar y decodificar actualmente. Cuando

el valor del parámetro de control SE está codificado, se usan valores de los parámetros de control SE del mismo tipo incluidos en el bloque superior A y el bloque izquierdo B que ya están codificados. Más específicamente, la probabilidad $p(x)$ que indica si el valor x del parámetro de control SE del bloque C (o el primer valor binario de la cadena binaria del parámetro de control SE) es 1 o 0 se predice basándose en una probabilidad condicional $p(x|)$ (condición A (valor del bloque superior) y condición B (valor del bloque izquierdo)) usando, como condiciones, el valor del parámetro de control SE del bloque superior A y el valor del parámetro de control SE del bloque izquierdo B.

La FIG. 38 ilustra el aumento en el uso de memoria cuando se usa un bloque superior.

En la FIG. 38, (x_P, y_P) es una posición de un píxel superior izquierdo de una unidad de predicción (PU, unidad de predicción de movimiento) que incluye el bloque C. Aquí, el bloque C es un bloque que incluye un parámetro de control (por ejemplo, `skip_flag`) actualmente a codificar. Adicionalmente, (x_P, y_A) en la FIG. 38 es una posición de un píxel que se incluye en el bloque B y se usa como condición A (valor del parámetro de control `skip_flag` del bloque superior). Adicionalmente, (x_L, y_P) en la FIG. 38 es una posición de un píxel que se incluye en el bloque A y se usa como condición B (valor del parámetro de control `skip_flag` del bloque izquierdo).

Para codificar o decodificar el valor del parámetro de control `skip_flag` del bloque C, el aparato de codificación o el aparato de decodificación necesita mantener el valor de `skip_flag` de PU (o un resultado de la determinación de una condición) correspondiente a la posición (x_P, y_A) incluido en el bloque superior B y la posición (x_L, y_P) incluido en el bloque izquierdo A. Asumiendo que la imagen tiene un ancho horizontal de 4096 píxeles, para codificar un parámetro de control `skip_flag`, es necesario mantener todos los valores de determinación incluidos en la fila superior (Línea L en la FIG. 38). En otras palabras, un parámetro de control necesita la capacidad de memoria obtenida por 4096 píxeles/tamaño de bloque.

En este punto, el bloque C a codificar tiene tamaños variables, por ejemplo, 64×64 , 16×16 , o 4×4 . Adicionalmente, el tamaño de bloque del bloque C a codificar o decodificar más tarde no se puede predecir cuando los bloques en la fila superior (Línea L) que incluyen (x_P, y_A) están codificados o decodificados. Esto se debe a que el tamaño de cada uno de los bloques en la fila inferior (fila que incluye el bloque C) no se conoce cuando la fila superior (fila que incluye el bloque A) se codifica o decodifica. De este modo, el aparato de codificación o el aparato de decodificación necesita mantener un valor de un parámetro de control (o valor de determinación) para cada tamaño mínimo de bloque, asumiendo que el tamaño de bloque más pequeño de entre todos los tamaños aplicados a los parámetros de control se usa como el tamaño de bloque de la fila inferior. En muchos tipos de señal, el tamaño de bloque más pequeño es 4×4 . Adicionalmente, en tipos de señal particulares, el tamaño de bloque más pequeño es 8×8 o 16×16 . Por ejemplo, cuando el tamaño de bloque más pequeño es 4×4 , el valor del parámetro de control que debe mantenerse es $4096 \div 4 = 1024$ bits. Las posiciones de los círculos negros en la FIG. 38 indican las condiciones que deben cumplirse, aunque los valores condicionales no son realmente necesarios cuando la fila inferior (fila que incluye el bloque C) se codifica y decodifica.

Adicionalmente, los dos bloques vecinos en la FIG. 38 (el bloque izquierdo A y el bloque superior B) siguen el concepto de los bloques vecinos en H.264, y no se introduce una nueva perspectiva sobre la división de los bloques jerárquicos. Como se describe a continuación, hay casos en que tales valores condicionales a los que se hace referencia en la FIG. 38 no siempre tiene sentido para parámetros de control adaptados a la división recursiva de cuatro árboles a introducir en HEVC, porque los parámetros de control siguen el orden de ejecución recursivo, la profundidad jerárquica, o las posiciones de los bloques.

En consecuencia, los presentes inventores han encontrado que el uso de la memoria aumenta al usar los valores condicionales de los bloques superiores al realizar la codificación o decodificación aritmética en los parámetros de control. Adicionalmente, los presentes inventores han encontrado que el uso de memoria aumenta aún más en HEVC.

Por el contrario, el procedimiento de decodificación de imagen de acuerdo con un aspecto de la presente invención es un procedimiento de decodificación de imagen que usa decodificación aritmética, y el procedimiento incluye: determinar un contexto para su uso en un bloque actual, de entre una pluralidad de contextos; realizar una decodificación aritmética en una secuencia de bits correspondiente al bloque actual, usando el contexto determinado para decodificar una secuencia binaria, siendo la secuencia de bits obtenida realizando una codificación aritmética en un parámetro de control del bloque actual; e inversamente binarizando la secuencia binaria para decodificar el parámetro de control del bloque actual, en el que la determinación de un contexto incluye: determinar un tipo de señal del parámetro de control del bloque actual; determinar el contexto bajo una primera condición en la que se usan parámetros de control decodificados de bloques vecinos del bloque actual, cuando el tipo de señal es un primer tipo, los bloques vecinos son un bloque izquierdo y un bloque superior del bloque actual; determinar el contexto bajo una segunda condición de que el parámetro de control decodificado del bloque superior no se usa, cuando el tipo de señal es un segundo tipo diferente del primer tipo; y determinar el contexto bajo una tercera condición de que el parámetro de control decodificado del bloque superior no se usa y se usa una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control del bloque actual, cuando el tipo de señal es un tercer tipo diferente del primer tipo y del segundo tipo, el primer tipo es uno de "split_coding_unit_flag" y "skip_flag", el segundo tipo es uno o varios de (i) "merge_flag", (ii) "ref_idx_l0" o "ref_idx_l1", (iii) "inter_pred_flag", (iv) "mvd_l0" o "mvd_l1", (v) "intra_chroma_pred_mode", (vi) "cbf_luma", y (vii) "cbf_cb" o "cbf_cr", y el tercer tipo es un tipo de señal de entre (i) el "merge_flag", (ii) el "ref_idx_l0" o el "ref_idx_l1", (iii) la "inter-pred-flag", (iv) el "mvd_l0" o el "mvd_l1", (v) el

"intra_chroma_pred_mode", (vi) el "cbf_luma", y (vii) el "cbf_cb" o el "cbf_cr", el tipo de señal es diferente al segundo tipo.

Con la estructura, el procedimiento de decodificación de imagen puede reducir el uso de memoria. Más específicamente, en el procedimiento de decodificación de imagen, como el parámetro de control del bloque superior no se usa para un parámetro de control del segundo tipo, no es necesario mantener el parámetro de control del segundo tipo del bloque superior. Con la estructura, en comparación con el caso en que el bloque izquierdo y el bloque superior se usan de manera uniforme "usando un modelo de contexto basado en valores de parámetros de control de bloques vecinos", el uso de memoria puede reducirse de acuerdo con el procedimiento de decodificación de imagen. Adicionalmente, el procedimiento de decodificación de imagen puede reducir adecuadamente el uso de memoria del parámetro de control del segundo tipo sin, por ejemplo, fracasando en la evaluación de una tasa BD de una imagen.

Adicionalmente, de acuerdo con el procedimiento de decodificación de imagen, se puede usar el contexto apropiado para una estructura de árbol jerárquica que es una estructura de datos que no se considera en el H.264 convencional y que es única para el nuevo estándar HEVC. Como alternativa, se puede realizar una referencia de memoria.

Adicionalmente, el procedimiento de decodificación de imagen puede seleccionar un contexto apropiado para un parámetro de control del tercer tipo mientras reduce el uso de memoria determinando el contexto usando una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control del bloque actual.

Adicionalmente, el tercer tipo puede ser el "inter_pred_flag".

Adicionalmente, el tercer tipo puede ser el "cbf_cb" y el "cbf_cr".

Adicionalmente, el tercer tipo puede ser el "cbf_luma".

Adicionalmente, el primer tipo puede ser el "split_coding_unit_flag" y el "skip_flag", el segundo tipo puede ser el "merge_flag", el "ref_idx_l0", el "ref_idx_l1", el "mvd_l0", el "mvd_l1" y el "intra_chroma_pred_mode", y el tercer tipo puede ser el "inter-pred-flag", el "cbf_luma", el "cbf_cb" y el "cbf_cr".

Adicionalmente, la segunda condición y la tercera condición pueden ser condiciones en las que no se usan los parámetros de control decodificados del bloque izquierdo y del bloque superior.

Con la estructura, el procedimiento de decodificación de imagen puede reducir el uso de memoria al no usar el parámetro de control del bloque izquierdo además del parámetro de control del bloque superior.

Adicionalmente, en la determinación de un contexto, se puede determinar un contexto predeterminado bajo la segunda condición, como el contexto para su uso en la decodificación aritmética del bloque actual, cuando el tipo de señal es el segundo tipo.

Con la estructura, el procedimiento de decodificación de imagen puede reducir la cantidad de procesamiento.

Adicionalmente, la determinación de un contexto puede incluir, además: determinar si el parámetro de control decodificado del bloque superior está disponible o no en la decodificación, basándose en una posición del bloque actual; y determinar el contexto bajo una de la segunda condición y la tercera condición, cuando el parámetro de control decodificado del bloque superior no está disponible.

Con la estructura, el procedimiento de decodificación de imagen puede reducir la cantidad de procesamiento.

Adicionalmente, en la determinación de un contexto, se puede determinar que el parámetro de control decodificado del bloque superior no está disponible, cuando el bloque actual está en un límite de corte.

Adicionalmente, en la determinación de un contexto, se puede determinar si el parámetro de control decodificado del bloque superior está o no está disponible, de acuerdo con una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control del bloque actual.

Adicionalmente, uno del segundo tipo y del tercer tipo puede ser un parámetro de control que tiene una estructura de datos predeterminada.

Adicionalmente, la determinación de un contexto puede incluir además determinar un contexto de un parámetro de control de una segunda unidad más pequeña que una primera unidad al conmutar entre la primera condición, la segunda condición y la tercera condición, basándose en un parámetro de control de la primera unidad.

Adicionalmente, "split_coding_unit_flag" puede indicar si el bloque actual está o no dividido en una pluralidad de bloques, "skip_flag" puede indicar si se debe saltar o no el bloque actual, "merge_flag" puede indicar si se usa o no un modo de fusión para el bloque actual, el "ref_idx_l0" puede indicar un índice de imagen de referencia de una lista 0 para el bloque actual, el "ref_idx_l1" puede indicar un índice de imagen de referencia de una lista 1 para el bloque actual, el "inter_pred_flag" puede indicar uno de unipredicción y bipredicción a usar para el bloque actual, el "mvd_l0" puede indicar una diferencia entre un componente de vector de movimiento de la lista 0 y un valor predicho del

componente de vector de movimiento, usándose el componente del vector de movimiento y el valor predicho para el bloque actual, el "mvd_l1" puede indicar una diferencia entre un componente de vector de movimiento de la lista 1 y un valor predicho del componente de vector de movimiento, usándose el componente del vector de movimiento y el valor predicho para el bloque actual, el "intra_chroma_pred_mode" puede indicar un modo de intrapredicción para una muestra de crominancia del bloque actual, el "cbf_luma" puede indicar si un bloque de transformación de luminancia del bloque actual contiene o no uno o varios niveles de coeficientes de transformación que no son iguales a 0, el "cbf_cb" puede indicar si un bloque de transformación Cb del bloque actual contiene o no uno o varios niveles de coeficiente de transformación que no es igual a 0, y el "cbf_cr" puede indicar si un bloque de transformación Cr del bloque actual contiene o no uno o varios niveles de coeficientes de transformación no iguales a 0.

Adicionalmente, los procedimientos de decodificación de acuerdo con un primer estándar y los procedimientos de decodificación de acuerdo con un segundo estándar pueden conmutarse de acuerdo con un identificador que indica uno del primer estándar y del segundo estándar, siendo el identificador incluido en una señal codificada y la determinación de un contexto, la ejecución y la binarización inversa se pueden realizar como los procedimientos de decodificación de acuerdo con el primer estándar, cuando el identificador indica el primer estándar.

Adicionalmente, el procedimiento de codificación de imagen de acuerdo con un aspecto de la presente invención es un procedimiento de codificación de imagen que usa codificación aritmética, y el procedimiento incluye: binarizar un parámetro de control de un bloque actual para generar una secuencia binaria; determinar un contexto para su uso en el bloque actual, de entre una pluralidad de contextos; y realizar una codificación aritmética en la secuencia binaria usando el contexto determinado para generar una secuencia de bits, en el que la determinación de un contexto incluye: determinar un tipo de señal del parámetro de control del bloque actual; determinar el contexto bajo una primera condición en la que se usan parámetros de control de bloques vecinos del bloque actual, cuando el tipo de señal es un primer tipo, los bloques vecinos son un bloque izquierdo y un bloque superior del bloque actual; determinar el contexto bajo una segunda condición de que el parámetro de control del bloque superior no se usa, cuando el tipo de señal es un segundo tipo diferente del primer tipo; y determinar el contexto bajo una tercera condición de que el parámetro de control descodificado del bloque superior no se usa y se usa una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control del bloque actual, cuando el tipo de señal es un tercer tipo diferente del primer tipo y del segundo tipo, el primer tipo es uno de "split_coding_unit_flag" y "skip_flag", y el segundo tipo es uno o varios de (i) "merge_flag", (ii) "ref_idx_l0" o "ref_idx_l1", (iii) "inter_pred_flag", (iv) "mvd_l0" o "mvd_l1", (v) "intra_chroma_pred_mode", (vi) "cbf_luma", y (vii) "cbf_cb" o "cbf_cr", y el tercer tipo es un tipo de señal de entre (i) el "merge_flag", (ii) el "ref_idx_l0" o el "ref_idx_l1", (iii) la "inter-pred-flag", (iv) el "mvd_l0" o el "mvd_l1", (v) el "intra_chroma_pred_mode", (vi) el "cbf_luma", y (vii) el "cbf_cb" o el "cbf_cr", el tipo de señal es diferente al segundo tipo.

Con la estructura, el procedimiento de codificación de imagen puede reducir el uso de memoria. Más específicamente, en el procedimiento de codificación de imagen, como el parámetro de control del bloque superior no se usa para un parámetro de control del segundo tipo, no es necesario mantener el parámetro de control del segundo tipo del bloque superior. Con la estructura, en comparación con el caso en que el bloque izquierdo y el bloque superior se usan de manera uniforme "usando un modelo de contexto basado en valores de parámetros de control de bloques vecinos", el uso de memoria puede reducirse de acuerdo con el procedimiento de codificación de imagen. Adicionalmente, el procedimiento de codificación de imagen puede reducir adecuadamente el uso de memoria del parámetro de control del segundo tipo sin, por ejemplo, fracasando en la evaluación de una tasa BD de una imagen.

Adicionalmente, de acuerdo con el procedimiento de codificación de imagen, se puede usar el contexto apropiado para una estructura de árbol jerárquica que es una estructura de datos que no se considera en el H.264 convencional y que es única para el nuevo estándar HEVC. Como alternativa, se puede realizar una referencia de memoria.

Adicionalmente, el procedimiento de codificación de imagen puede seleccionar un contexto apropiado para un parámetro de control del tercer tipo mientras reduce el uso de memoria determinando el contexto usando una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control del bloque actual.

Adicionalmente, el aparato de decodificación de imagen de acuerdo con un aspecto de la presente invención es un aparato de decodificación de imagen que usa decodificación aritmética, y el aparato incluye: una unidad de control de contexto configurada para determinar un contexto para su uso en un bloque actual, de entre una pluralidad de contextos; una unidad de decodificación aritmética configurada para realizar decodificación aritmética en una secuencia de bits correspondiente al bloque actual, usando el contexto determinado para decodificar una secuencia binaria, siendo la secuencia de bits obtenida realizando una codificación aritmética en un parámetro de control del bloque actual; y una unidad de binarización inversa configurada para binarizar inversamente la secuencia binaria para decodificar el parámetro de control del bloque actual, en el que la unidad de control de contexto está configurada para: determinar un tipo de señal del parámetro de control del bloque actual; determinar el contexto bajo una primera condición en la que se usan parámetros de control decodificados de bloques vecinos del bloque actual, cuando el tipo de señal es un primer tipo, los bloques vecinos son un bloque izquierdo y un bloque superior del bloque actual; determinar el contexto bajo una segunda condición de que el parámetro de control decodificado del bloque superior no se usa, cuando el tipo de señal es un segundo tipo diferente del primer tipo; y determinar el contexto bajo una tercera condición de que el parámetro de control descodificado del bloque superior no se usa y se usa una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control del bloque actual, cuando el tipo de

señal es un tercer tipo diferente del primer tipo y del segundo tipo, el primer tipo es uno de "split_coding_unit_flag" y "skip_flag", y el segundo tipo es uno o varios de (i) "merge_flag", (ii) "ref_idx_l0" o "ref_idx_l1", (iii) "inter_pred_flag", (iv) "mvd_l0" o "mvd_l1", (v) "intra_chroma_pred_mode", (vi) "cbf_luma", y (vii) "cbf_cb" o "cbf_cr", y el tercer tipo es un tipo de señal de entre (i) el "merge_flag", (ii) el "ref_idx_l0" o el "ref_idx_l1", (iii) la "inter-pred-flag", (iv) el "mvd_l0" o el "mvd_l1", (v) el "intra_chroma_pred_mode", (vi) el "cbf_luma", y (vii) el "cbf_cb" o el "cbf_cr", el tipo de señal es diferente al segundo tipo.

Con la configuración, el aparato de decodificación de imagen puede reducir el uso de memoria.

Adicionalmente, el aparato de codificación de imagen de acuerdo con un aspecto de la presente invención es un aparato de codificación de imagen que usa codificación aritmética, y el aparato incluye: una unidad de binarización configurada para binarizar un parámetro de control de un bloque actual para generar una secuencia binaria; una unidad de control de contexto configurada para determinar un contexto para su uso en el bloque actual, de entre una pluralidad de contextos; y una unidad de codificación aritmética configurada para realizar codificación aritmética en la secuencia binaria usando el contexto determinado para generar una secuencia de bits, en el que la unidad de control de contexto está configurada para: determinar un tipo de señal del parámetro de control del bloque actual; determinar el contexto bajo una primera condición en la que se usan parámetros de control de bloques vecinos del bloque actual, cuando el tipo de señal es un primer tipo, los bloques vecinos son un bloque izquierdo y un bloque superior del bloque actual; determinar el contexto bajo una segunda condición de que el parámetro de control del bloque superior no se usa, cuando el tipo de señal es un segundo tipo diferente del primer tipo; y determinar el contexto bajo una tercera condición de que el parámetro de control descodificado del bloque superior no se usa y se usa una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control del bloque actual, cuando el tipo de señal es un tercer tipo diferente del primer tipo y del segundo tipo, el primer tipo es uno de "split_coding_unit_flag" y "skip_flag", y el segundo tipo es uno o varios de (i) "merge_flag", (ii) "ref_idx_l0" o "ref_idx_l1", (iii) "inter_pred_flag", (iv) "mvd_l0" o "mvd_l1", (v) "intra_chroma_pred_mode", (vi) "cbf_luma", y (vii) "cbf_cb" o "cbf_cr", y el tercer tipo es un tipo de señal de entre (i)

EP91521AKlpp el "merge_flag", (ii) el "ref_idx_l0" o el "ref_idx_l1", (iii) la "inter-pred-flag", (iv) el "mvd_l0" o el "mvd_l1", (v) el "intra_chroma_pred_mode", (vi) el "cbf_luma", y (vii) el "cbf_cb" o el "cbf_cr", el tipo de señal es diferente al segundo tipo.

Con la configuración, el aparato de codificación de imagen puede reducir el uso de memoria.

Adicionalmente, el aparato de codificación y decodificación de imagen de acuerdo con un aspecto de la presente invención es un aparato de codificación y decodificación de imagen que incluye el aparato de decodificación de imágenes y el aparato de codificación de imágenes.

Los aspectos generales o específicos pueden ser implementados por un sistema, un procedimiento, un circuito integrado, un programa informático, o un medio de grabación, o por una combinación arbitraria del sistema, el procedimiento, el circuito integrado, el programa informático y el medio de grabación.

El aparato de decodificación de imagen y el aparato de codificación de imagen de acuerdo con un aspecto de la presente invención se describirán específicamente con referencia a los dibujos.

Las realizaciones descritas a continuación indican ejemplos específicos de la presente invención. Los valores, formas, materiales, elementos constituyentes, posiciones y conexiones de los elementos constituyentes, etapas y órdenes de las etapas indicados en las realizaciones son ejemplos, y no limitan la presente invención. La invención se define por las reivindicaciones anexas a la misma.

(Realización 1)

Se describirá un aparato de codificación de imagen de acuerdo con la realización 1 de la presente invención. El aparato de codificación de imagen de acuerdo con la realización 1 determina un contexto cambiando entre (1) usando el bloque superior y (2) sin usar el bloque superior, de acuerdo con un tipo de señal de un parámetro de control en codificación aritmética. Con la estructura, se puede suprimir el deterioro de la calidad de la imagen y reducir el uso de memoria.

En primer lugar, se describirá una configuración del aparato de codificación de imagen de acuerdo con la realización 1.

La FIG. 1 es un diagrama de bloques que ilustra un aparato de codificación de imagen 100 de acuerdo con la realización 1.

El aparato de codificación de imagen 100 en la FIG. 1 es un aparato de codificación de imagen que usa codificación aritmética, y codifica una señal de imagen de entrada 121 para generar un flujo de bits 124. El aparato de codificación de imagen 100 incluye una unidad de control 101, una unidad de sustracción 102, una unidad de transformación y cuantización 103, una unidad de codificación de longitud variable 104, una unidad de cuantización inversa y transformación inversa 105, una unidad de suma 106, una unidad de intrapredicción 107, una unidad de interpredicción 108 y un conmutador 109.

- La unidad de control 101 calcula un parámetro de control 130 basándose en la señal de imagen de entrada 121 a codificar. Por ejemplo, el parámetro de control 130 incluye información sobre un tipo de imagen de la señal de imagen de entrada 121 a codificar, un tamaño de una unidad de predicción de movimiento (unidad de predicción, PU) del bloque actual a codificar, e información de control de la unidad de predicción de movimiento. En este punto, el parámetro de control 130 (datos de control) en sí mismo debe codificarse. De este modo, la unidad de control 101 genera el parámetro de control 130 a la unidad de codificación de longitud variable 104.
- La unidad de sustracción 102 calcula una señal residual 122 que es una diferencia (valor residual) entre la señal de imagen de entrada 121 y una señal de predicción de imagen 129 basándose en la unidad de bloque.
- La unidad de transformación y cuantización 103 transforma la señal residual 122 en valores de coeficiente de frecuencia y cuantiza los valores de coeficiente de frecuencia obtenidos en coeficientes de transformación cuantizados 123 (datos residuales).
- La unidad 105 de cuantización inversa y transformación inversa cuantiza inversamente los coeficientes de transformación cuantizados 123 en los valores del coeficiente de frecuencia y transforma inversamente los valores del coeficiente de frecuencia obtenidos en una señal residual decodificada 125.
- La unidad de adición 106 añade la señal residual 125 a la señal de predicción de imagen 129, y emite una señal de imagen decodificada 126.
- La unidad 107 de intrapredicción realiza intrapredicción usando la señal de imagen decodificada 126 para generar una señal de predicción de imagen 127. La unidad 108 de interpredicción realiza interpredicción usando la señal de imagen decodificada 126 para generar una señal de predicción de imagen 128.
- El conmutador 109 selecciona una de la señal de predicción de imagen 127 y la señal de predicción de imagen 128, y genera la señal seleccionada como la señal de predicción de imagen 129.
- La unidad de codificación de longitud variable 104 codifica, usando el CABAC, los coeficientes de transformación cuantizados 123 y el parámetro de control 130 para cada bloque de entrada para generar el flujo de bits 124.
- A continuación, se describirá la configuración de la unidad de codificación de longitud variable 104.
- La FIG. 2 es un diagrama de bloques funcional de la unidad de codificación de longitud variable 104. La unidad de codificación de longitud variable 104 incluye una unidad de binarización 141, una unidad de control de contexto 142 y una unidad de codificación aritmética binaria 143. A continuación, se describe el procedimiento de codificación de longitud variable en el parámetro de control 130. Aunque se omite la descripción sobre el procedimiento de codificación de longitud variable en los coeficientes de transformación cuantizados 123, el procedimiento puede implementarse, por ejemplo, utilizando una técnica conocida.
- La unidad de binarización 141 binariza el parámetro de control 130 para generar una secuencia binaria 151. Más específicamente, la unidad de binarización 141 es una unidad de procesamiento que realiza "II.1) procesamiento de binarización" de acuerdo con NPL 1. La unidad de binarización 141 transforma el parámetro de control 130 en la secuencia binaria 151 denominada "cadena de bin" para cada tipo de señal, de acuerdo con un procedimiento de binarización predeterminado. La correspondencia entre los tipos de señal y los procedimientos de binarización se describirán más adelante. Cuando el parámetro de control de entrada 130 es un valor binario, tal como una bandera, la unidad de binarización 141 genera el parámetro de control 130 como la secuencia binaria 151 tal como es.
- La unidad de control de contexto 142 determina un contexto para su uso en la codificación aritmética del parámetro de control 130 incluido en un bloque actual, de entre una pluralidad de contextos (una tabla de estado de probabilidad). Adicionalmente, la unidad de control de contexto 142 genera un índice de contexto 152 que especifica el contexto determinado a la unidad de codificación aritmética binaria 143.
- Más específicamente, la unidad de control de contexto 142 es una unidad de procesamiento que realiza "2) modelado de contexto" de acuerdo con NPL 1. La unidad de control de contexto 142 recibe secuencialmente una pluralidad de elementos incluidos en la secuencia binaria 151 que sale de la unidad de codificación aritmética binaria 143. La unidad de control de contexto 142 selecciona uno de los contextos a utilizar para el binario del parámetro de control 130, de acuerdo con el tipo de señal del parámetro de control 130 y una posición del elemento del binario en la secuencia binaria 151, y genera, a la unidad de codificación aritmética binaria 143, el índice de contexto 152 que es un índice que indica el contexto seleccionado.
- Adicionalmente, la unidad de control de contexto 142 sostiene la tabla de valores de estado de probabilidad (valores de índice de contexto) obtenidos dividiendo los elementos en la secuencia binaria del parámetro de control 130 en condiciones de probabilidades condicionales, como estados del contexto, e inicializa y actualiza la tabla de estado de probabilidad.
- Adicionalmente, la unidad de control de contexto 142 sostiene un estado (índice de estado de probabilidad) para cada vez que está presente esta condición τ (para cada contexto), como una división adicional de un tipo de señal (para

cada número de elemento en la secuencia binaria del parámetro de control 130 cuando el número de elementos en la secuencia binaria es dos o más; lo mismo se aplicará de aquí en adelante). El estado se representa mediante el valor total de 7 bits combinando la probabilidad de ocurrencia P (relación interna, típicamente, un valor de 6 bits) que es la probabilidad más baja de uno de los dos valores 0 y 1, y un valor de 1 bit que indica cuál de los valores tiene la probabilidad más alta. Adicionalmente, mantener un estado significa inicializar y actualizar el estado. Por ejemplo, la actualización corresponde a cambiar la indexación que indica un estado de probabilidad actual (es decir, una probabilidad) como una transición entre 64 estados finitos como en H.264.

Cuando ocurre un evento X en el lado más probable que tiene la mayor probabilidad entre los dos valores, se incrementa ligeramente la relación de la probabilidad en el lado más probable. Por ejemplo, la unidad de control de contexto 142 puede aumentar ligeramente la relación de probabilidad en el lado más probable aumentando o disminuyendo, por 1, el valor del índice de estado de probabilidad correspondiente a 64 tablas. Por otro lado, cuando ocurre un evento No X que tiene la probabilidad más baja (contra la probabilidad pronosticada), la unidad de control de contexto 142 disminuye en gran medida la relación de la probabilidad más probable mantenida basándose en un coeficiente de escala predeterminado α (por ejemplo, $\approx 0,95$) (ver Fig. 6 de NPL 1). La unidad de control de contexto 142 de acuerdo con la realización 1 transita y sostiene un estado, basado en un valor de cambio de índice de tabla correspondiente para asociarse con el cambio en consideración de a como en H.264.

La unidad de codificación aritmética binaria 143 realiza la codificación aritmética en la secuencia binaria 151 usando el contexto determinado por la unidad de control de contexto 142 para generar el flujo de bits 124 (secuencia de bits).

Más específicamente, la unidad de codificación aritmética binaria 143 es una unidad de procesamiento que realiza "3) codificación aritmética binaria" de acuerdo con NPL 1. La unidad de codificación aritmética binaria 143 realiza la codificación aritmética en la secuencia binaria 151 usando el contexto específico por el índice de control de contexto 152 para generar el flujo de bits 124. En este punto, la codificación aritmética es manejar eventos que ocurren para los parámetros de control 130 de diversos tipos de señales como una suma acumulativa de probabilidades, y determinar correspondencias entre los eventos al reducir el rango a un rango predeterminado en una línea numérica.

En primer lugar, la unidad de codificación aritmética binaria 143 divide la línea de un número en dos medias secciones, de acuerdo con las probabilidades de ocurrencia de dos valores posibles del binario, dados desde la unidad de control de contexto 142. Cuando el valor real que ocurre para el binario (por ejemplo, 0) es un valor con una probabilidad más alta (que excede 0,5 (por ejemplo, 0,75)), la unidad de codificación aritmética binaria 143 mantiene el límite inferior "Bajo" en el rango en la línea numérica sin cambio, y establece un valor correspondiente al resultado de multiplicar una vez un coeficiente de escala 0,95 por la probabilidad 0,75 esta vez, a un nuevo rango. Por otro lado, cuando el valor binario realmente generado es un valor predicho con una probabilidad menor, la unidad de codificación aritmética binaria 143 desplaza el límite inferior "Bajo" por la probabilidad más alta, y cambia el rango de acuerdo con la probabilidad más baja. Las secciones se mantienen de acuerdo con una suma acumulativa de resultados de multiplicaciones de los rangos de probabilidad. Cuando se produce sucesivamente un valor con una probabilidad menor, la precisión de la longitud del rango pronto se vuelve más baja que la precisión que puede garantizarse mediante un cálculo. En este punto, la unidad de codificación aritmética binaria 143 amplía (renormaliza) el rango para mantener la precisión, y genera la secuencia de bits que indica el rango actual. En cambio, cuando un valor con una probabilidad más alta (0,95, etc.) ocurre sucesivamente, los valores de probabilidad pueden soportar varios cálculos (transiciones de estado en el caso de implementación por una tabla) hasta que la longitud del rango se acorte a una longitud predeterminada incluso con la multiplicación de los valores. De este modo, el número de símbolos que se pueden acumular hasta que se genera el bit son muchos.

La FIG. 3 es una tabla en la que los parámetros de control 130, cada uno de los cuales usa un modelo de contexto basado en un valor del parámetro de control 130 de un bloque vecino, se clasifican.

El significado de cada columna se describirá desde la izquierda de la tabla.

(c2) El tipo de señal (elemento de sintaxis) indica un nombre específico de un tipo de señal del parámetro de control 130. El significado de cada uno de los tipos de señal se describirá más adelante.

(c3) El esquema de binarización indica un esquema de binarización que se aplicará al parámetro de control 130 (SE) especificado en la columna inmediatamente a la izquierda. La unidad de binarización 141 realiza el procedimiento de binarización. En la columna, "Longitud fija" significa que la unidad de binarización 141 genera el valor del parámetro de control 130 en la sección inmediatamente a la izquierda como una secuencia binaria (cadena binaria) de una longitud fija. En HEVC, un tipo de señal del parámetro de control 130 cuyo nombre termina con "bandera" es un valor binario de 0 o 1. De este modo, la unidad de binarización 141 emite solo el primer elemento ($\text{binIdx} = 0$) como el elemento de la secuencia binaria 151, y no emite los elementos después del segundo elemento ($\text{binIdx} > 1$). En otras palabras, la unidad de binarización 141 genera el valor del parámetro de control 130 como la secuencia binaria 151 tal como es.

Adicionalmente, "Longitud variable" en la columna significa que la unidad de binarización 141 mapea, a una secuencia binaria, el valor del parámetro de control 130 usando secuencias binarias con longitudes variables respectivas cuyos valores están asociados para tener longitudes binarias en orden ascendente de las frecuencias de ocurrencia (cadena binaria o secuencias binarias cada una con el número de elementos ≥ 1), y genera la secuencia binaria. Por ejemplo,

la unidad de binarización 141 emplea y genera un esquema de acuerdo con el tipo de señal, tal como un esquema unario (truncado) y una combinación de los esquemas de Golomb unarios y otros exponenciales (ver "A. Binarización de NPL 1). En el caso de "Longitud variable", el número de elementos de la secuencia binaria 151 a veces se limita a 1, o es igual o mayor que 2. Una unidad de binarización inversa en un aparato de decodificación de imagen que se describirá más adelante realiza una transformación inversa al esquema de binarización para decodificar la secuencia binaria de entrada en un multivalor o un valor de bandera.

En cuanto al índice de contexto (c4) del primer elemento (binIdx = 0), la unidad de control de contexto 142 indica la elección de un índice de contexto (incremento) a aplicar al primer elemento incluido en una secuencia binaria generada de acuerdo con el esquema de binarización especificado en la columna de c3. En la columna, "0, 1, 2" indica que la unidad de control de contexto 142 selecciona y aplica una de las tres tablas de estado de probabilidad (contextos). Por ejemplo, se preparan tres índices de contexto con condiciones detalladas para el tipo de señal "skip_flag", es decir, se preparan tres contextos y la codificación aritmética se realiza en los índices de contexto.

De manera similar, "0, 1, 2, 3" en la columna c4 indica que el contexto que se aplicará al primer elemento (binIdx = 0) incluido en la secuencia binaria 151 se selecciona entre uno de los cuatro valores, o 0, 1, 2 o 3. La secuencia binaria 151 se obtiene mediante mapeo, a una secuencia binaria, el valor del parámetro de control 130 del tipo de señal especificado en la columna de c2, de acuerdo con el esquema de binarización en la columna de c3. Las expresiones condicionales en la columna se describirán más adelante.

Con respecto a (c5) Condición del bloque izquierdo L (condl), la unidad de control de contexto 142 indica la condición del bloque izquierdo para seleccionar uno de 0, 1 y 2 en la columna c4. La condición de bloque izquierdo L tiene un valor de verdadero o falso determinado de acuerdo con el valor del parámetro de control del bloque izquierdo correspondiente al parámetro de control a codificar (o decodificar).

Por ejemplo, en el caso donde el parámetro de control (SE) es skip_flag, la condición de bloque izquierdo L tiene el valor verdadero cuando skip_flag [xL] [yL] indica verdadero (por ejemplo, 1), y tiene el valor de falso cuando indica falso (por ejemplo, 0).

Con respecto a (c6) Condición del bloque superior A, la unidad de control de contexto 142 indica la condición de bloque superior para seleccionar uno de 0, 1 y 2 en elementos de codificación o decodificación de una secuencia especificada en la columna c4. La condición de bloque superior A tiene un valor de verdadero o falso determinado de acuerdo con el valor del parámetro de control del bloque superior correspondiente al parámetro de control a codificar (o decodificar). Por ejemplo, en el caso donde el parámetro de control (SE) es skip_flag, la condición de bloque superior A tiene el valor verdadero cuando skip_flag[xA][yA] indica verdadero (por ejemplo, 1), y tiene el valor de falso cuando indica falso (por ejemplo, 0).

Aunque no se ilustra, el tipo de señal de más de dos bits está asociado con "(c7) Incremento de contexto que se aplicará a binIdx >= 1". Esto (c7) indica el modelo de contexto aplicado por la unidad de control de contexto 142 a un binario después del segundo elemento en la secuencia binaria (valor binario de un elemento de secuencia binaria que incluye un valor de índice de binIdx >= 1).

En el procedimiento de codificación de la realización 1, las siguientes operaciones se conmutan de acuerdo con el tipo de señal del parámetro de control 130 para la condición de bloque izquierdo L y la condición de bloque superior A (operado usando diferentes patrones):

- (Patrón 1) Usar dos bloques vecinos (un valor de determinación de la condición de bloque izquierdo L y un valor de determinación de la condición de bloque superior A);
- (Patrón 2) Usar un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L); y
- (Patrón 3) Usar cero bloques vecinos (sin usar un valor de determinación de la condición de bloque izquierdo L ni un valor de determinación de la condición de bloque superior A).

La FIG. 4 es un diagrama de flujo que indica un procedimiento de codificación de imagen de acuerdo con la realización 1 que se realiza mediante la unidad de codificación de longitud variable 104 en la FIG. 2.

En primer lugar, la unidad de binarización 141 mapea el valor del parámetro de control 130 a una secuencia binaria de acuerdo con un esquema correspondiente al tipo de señal del parámetro de control 130 (S101).

A continuación, la unidad de control de contexto 142 obtiene un valor básico de un contexto para su uso en la codificación aritmética del parámetro de control 130 (S102). Por ejemplo, la unidad de control de contexto 142 determina el valor básico de acuerdo con el tipo de imagen (I, P o B).

A continuación, la unidad de control de contexto 142 determina un valor de contexto usando uno de los patrones 1 a 3, basado en el tipo de señal del parámetro de control 130 (S103). En este punto, determinar un valor de contexto es equivalente a determinar un valor de ajuste (valor de incremento CtxldxInc) para el valor básico del contexto.

En primer lugar, la unidad de control de contexto 142 determina el tipo de señal del parámetro de control 130 (S103). Cuando el tipo de señal del parámetro de control 130 es el primer tipo correspondiente al patrón 1 (el primer tipo en

S104), la unidad de control de contexto 142 determina un valor de contexto usando un valor de determinación derivado de los valores de los parámetros de control de dos bloques vecinos (bloque A y bloque B) (S105). En otras palabras, la unidad de control de contexto 142 determina un contexto a condición de que se usen los parámetros de control de los dos bloques vecinos del bloque izquierdo y el bloque superior. En este punto, la unidad de control de contexto 142 usa tanto el resultado de la determinación de (c5) condI como un resultado de la determinación de (c6) condA en la FIG. 3. Por consiguiente, los datos de una fila de imágenes se mantienen para los parámetros de control del primer tipo.

Por otro lado, cuando el tipo de señal del parámetro de control 130 es el segundo tipo correspondiente al patrón 2 (el segundo tipo en S104), la unidad de control de contexto 142 determina un valor de contexto usando un valor de un parámetro de control de un bloque vecino (un bloque inmediatamente adyacente en orden de codificación) (S106). En otras palabras, la unidad de control de contexto 142 determina el valor de contexto a condición de que el parámetro de control del bloque superior no se use.

Por otro lado, cuando el tipo de señal del parámetro de control 130 es el tercer tipo correspondiente al patrón 3 (el tercer tipo en S104), la unidad de control de contexto 142 determina fijamente un valor de contexto sin usar los dos parámetros de control del bloque superior y el bloque izquierdo (S107).

A continuación, la unidad de control de contexto 142 agrega el incremento determinado en la Etapa S103 al valor básico del índice de contexto determinado en la Etapa S102 para determinar un valor de índice de contexto (S108).

Finalmente, la unidad de codificación aritmética binaria 143 realiza la codificación aritmética en el valor binario del primer elemento usando el valor de contexto especificado por el valor de índice de contexto determinado en la Etapa S108 para generar la secuencia de bits (flujo de bits 124) (S109).

A continuación, cuando los procedimientos de las Etapas S102 a S109 no se ejecutan en todos los elementos incluidos en la secuencia binaria (No en S110), la unidad de codificación de longitud variable 104 realiza los procedimientos desde las Etapas S102 a S109 en el siguiente elemento incluido en la secuencia binaria. Por otro lado, cuando los procedimientos de las Etapas S102 a S109 se completan en todos los elementos incluidos en la secuencia binaria (Sí en S110), la unidad de codificación de longitud variable 104 finaliza el procesamiento de codificación en el parámetro de control del bloque actual.

Como se ha descrito anteriormente, el aparato de codificación de imagen 100 de acuerdo con la realización 1 determina un contexto usando el bloque superior para realizar la codificación aritmética en el parámetro de control del primer tipo, y determina un contexto sin usar el bloque superior para los parámetros de control del segundo y tercer tipo.

En comparación con el caso en que el bloque izquierdo y el bloque superior se usan de manera uniforme "usando un modelo de contexto basado en valores de parámetros de control de bloques vecinos", el aparato de codificación de imagen 100 puede reducir el uso de memoria con la configuración. De este modo, el aparato de codificación de imagen 100 puede suprimir el deterioro de la calidad de imagen y reducir el uso de memoria.

(Realización 2)

La realización 2 describirá un aparato de decodificación de imagen que decodifica el flujo de bits 124 generado por el aparato de codificación de imágenes 100.

La FIG. 5 es un diagrama de bloques que ilustra un aparato de decodificación de imagen 200 de acuerdo con la realización 2. El aparato de decodificación de imagen 200 es un aparato de decodificación de imagen que usa decodificación aritmética, y decodifica el flujo de bits 124 para generar una señal de imagen 229. En este punto, el flujo de bits 124 es, por ejemplo, generado por el aparato de codificación de imagen 100.

El aparato de decodificación de imagen 200 incluye una unidad de control 201, una unidad de decodificación de longitud variable 202, una unidad de cuantización inversa 204, una unidad de transformación inversa 205, una unidad de suma 206, una unidad de intrapredicción 207, y una unidad de interpredicción 208.

El aparato de decodificación de imagen 200 realiza el procesamiento de decodificación para cada flujo de bits de una unidad de procesamiento predeterminada. La unidad de procesamiento es, por ejemplo, una unidad de corte o una unidad de bloque.

La unidad de decodificación de longitud variable 202 realiza la decodificación aritmética en el flujo de bits 124 para generar un parámetro de control 230 (elemento de sintaxis de datos de control) y coeficientes de transformación cuantizados 223 (valores de elementos de sintaxis de datos residuales). La unidad de control 201 recibe el parámetro de control generado 230.

La unidad de control 201 controla cada una de las unidades de procesamiento incluidas en el aparato de decodificación de imagen 200, de acuerdo con el parámetro de control 230.

La unidad de cuantización inversa 204 cuantiza inversamente los coeficientes de transformación cuantizados 223 en

coeficientes de transformación ortogonales 224.

La unidad de transformación inversa 205 transforma inversamente los coeficientes de transformación ortogonal 224 para decodificar una señal residual 225. La unidad de adición 206 agrega la señal residual 225 a una señal de predicción de imagen (señal de imagen 229) para generar una señal de imagen decodificada 226.

- 5 La unidad 207 de intrapredicción realiza intrapredicción usando la señal de imagen decodificada 226 para generar una señal de predicción de imagen 227. La unidad 208 de interpredicción realiza interpredicción usando la señal de imagen decodificada 226 para generar una señal de predicción de imagen 228.

El conmutador 209 selecciona una de la señal de predicción de imagen 227 y la señal de predicción de imagen 228, y genera la señal seleccionada como la señal de imagen 229 (señal de predicción de imagen).

- 10 A continuación, se describirá la configuración de la unidad de decodificación de longitud variable 202.

La FIG. 6 es un diagrama de bloques funcional que ilustra una configuración de la unidad de decodificación de longitud variable 202. La unidad de decodificación de longitud variable 202 incluye una unidad de decodificación aritmética binaria 243, una unidad de control de contexto 242 y una unidad de binarización inversa 241. A continuación, se describe el procedimiento de decodificación de longitud variable en el parámetro de control 230. Aunque se omite la descripción sobre el procedimiento de decodificación de longitud variable en los coeficientes de transformación cuantizados 223, el procedimiento puede implementarse, por ejemplo, utilizando una técnica conocida.

- 15 La unidad de control de contexto 242 determina un contexto para su uso en la decodificación aritmética del parámetro de control 230 del bloque actual, de entre una pluralidad de contextos. Adicionalmente, la unidad de control de contexto 242 genera un índice de contexto 252 que especifica el contexto determinado a la unidad de decodificación aritmética binaria 243.

- 20 Más específicamente, la unidad de control de contexto 242 usa el mismo modelo de contexto que el de la unidad de control de contexto 142 en la FIG. 2 como modelo de transición de probabilidad sostenida. Cuando la unidad de codificación aritmética 143 usa 64 estados de probabilidad, la unidad de decodificación aritmética binaria 243 también sostiene los 64 estados de probabilidad. Esto se debe a que tanto el codificador como el decodificador necesitan interpretar un rango de la línea numérica que se codificará exactamente de la misma manera. De este modo, el decodificador usa el mismo patrón que el patrón seleccionado por el codificador entre los tres patrones 1 a 3.

- 25 La unidad de decodificación aritmética 243 realiza la decodificación aritmética en la secuencia de bits (flujo de bits 124) usando el contexto determinado por la unidad de control de contexto 242 para decodificar la secuencia binaria 251. Más específicamente, la unidad de decodificación aritmética 243 decodifica la secuencia de bits de entrada en la secuencia binaria 251, de acuerdo con el contexto (tabla de estado de probabilidad) especificado por el índice de contexto dado desde la unidad de control de contexto 242.

- 30 La unidad de binarización inversa 241 decodifica la secuencia binaria 251 en un parámetro de control 230 si es necesario a través del procedimiento de binarización inversa. En consecuencia, la unidad de control de contexto 142 incluida en el aparato de codificación de imagen 100 y la unidad de control de contexto 242 incluida en el aparato de decodificación de imagen 200 usan el mismo modelo de contexto tanto en la codificación aritmética como en la decodificación aritmética de un parámetro de control de un cierto tipo de señal.

La FIG. 7 es un diagrama de flujo que indica un procedimiento de decodificación de imagen de acuerdo con la realización 2 que se realiza mediante la unidad de decodificación de longitud variable 202.

En primer lugar, la unidad de decodificación de longitud variable 202 obtiene el flujo de bits 124 (S201).

- 40 A continuación, la unidad de control de contexto 242 determina un tipo de señal de un parámetro de control a decodificar, de acuerdo con la estructura de datos del flujo de bits 124 (S202).

A continuación, la unidad de control de contexto 242 determina un valor básico de un contexto para su uso en la decodificación aritmética del parámetro de control a decodificar (S203). Por ejemplo, la unidad de control de contexto 242 determina el valor básico de acuerdo con el tipo de imagen (I, P o B).

- 45 A continuación, la unidad de control de contexto 242 determina un valor de contexto usando uno de los patrones 1 a 3, basado en el tipo de señal del parámetro de control (S204). En este punto, determinar un valor de contexto es equivalente a determinar un valor de ajuste (valor de incremento CtxIdxInc) para el valor básico del contexto. Por ejemplo, la unidad de control de contexto 242 determina estáticamente uno de los patrones 1 a 3 basándose en el tipo de señal del parámetro de control siguiendo una tabla predeterminada.

- 50 La unidad de control de contexto 242 cambia entre bloques vecinos para su uso en la determinación de un contexto para obtener un valor binario del primer elemento incluido en la secuencia binaria 251 usando la decodificación aritmética, de acuerdo con el tipo de señal del parámetro de control.

En primer lugar, la unidad de control de contexto 242 determina el tipo de señal del parámetro de control 230 (S205).

Cuando el tipo de señal es el primer tipo correspondiente al patrón 1 (el primer tipo en S205), la unidad de control de contexto 242 determina un valor de contexto usando parámetros de control de dos bloques vecinos (S206). En otras palabras, la unidad de control de contexto 242 determina el valor de contexto a condición de que se usen los parámetros decodificados de control de los dos bloques vecinos del bloque izquierdo y el bloque superior.

5 Por otro lado, cuando el tipo de señal es el segundo tipo correspondiente al patrón 2 (el segundo tipo en S205), la unidad de control de contexto 242 determina un valor de contexto usando un valor de un parámetro de control de un bloque vecino (un bloque inmediatamente adyacente en orden de codificación) (S207). En otras palabras, la unidad de control de contexto 242 determina el valor de contexto a condición de que el parámetro de control del bloque decodificado superior no se use.

10 Por otro lado, cuando el tipo de señal es el tercer tipo correspondiente al patrón 3 (el tercer tipo en S205), la unidad de control de contexto 242 determina fijamente un valor de contexto (S208). En otras palabras, la unidad de control de contexto 242 determina el valor de contexto a condición de que los parámetros de control decodificados del bloque superior y el bloque izquierdo no se usen.

15 A continuación, la unidad de control de contexto 242 agrega el incremento determinado en la Etapa S204 al valor básico del índice de contexto determinado en la Etapa S203 para determinar un valor de índice de contexto (S209).

A continuación, la unidad de decodificación aritmética binaria 243 determina uno de los elementos de la secuencia binaria a través de la decodificación usando el valor de contexto indicado por el valor de índice de contexto dado desde la unidad de control de contexto 242 (S210).

20 A continuación, cuando los procedimientos de las Etapas S203 a S210 no se ejecutan en todos los elementos incluidos en la secuencia binaria (No en S211), la unidad de decodificación de longitud variable 202 realiza los procedimientos desde las Etapas S203 a S210 en el siguiente elemento incluido en la secuencia binaria.

25 Por otro lado, cuando los procedimientos de las Etapas S203 a S210 se completan en todos los elementos incluidos en la secuencia binaria (Sí en S211), la unidad de binarización inversa 241 cambia uno o varios de los elementos de la secuencia binaria 251 obtenida repitiendo los procedimientos desde las Etapas S203 a S210 más de una vez para generar el parámetro de control 230 (S212).

Como se ha descrito anteriormente, el aparato de decodificación de imagen 200 de acuerdo con la realización 2 determina un contexto usando el bloque superior para realizar la decodificación aritmética en el parámetro de control del primer tipo, y determina un contexto sin usar el bloque superior para los parámetros de control del segundo y tercer tipo.

30 En comparación con el caso en que el bloque izquierdo y el bloque superior se usan de manera uniforme "usando un modelo de contexto basado en valores de parámetros de control de bloques vecinos", el aparato de decodificación de imagen 200 puede reducir el uso de memoria con la configuración. De este modo, el aparato de decodificación de imagen 200 puede suprimir el deterioro de la calidad de imagen y reducir el uso de memoria.

35 Por ejemplo, cuando la secuencia binaria 251 es una bandera y tiene solo un elemento, es decir, la secuencia binaria 251 se compone de 1 binario, la unidad de binarización inversa 241 puede generar la secuencia binaria 251 tal como es.

Además de la descripción anterior, la unidad de control 101 o 201 puede controlar cada una de las unidades de procesamiento o hacer referencia a un valor de una memoria, a través de una línea de señal que no se ilustra.

40 Aunque la unidad de control de contexto 142 o 242 cambia entre los tres patrones 1 a 3 de acuerdo con un tipo de señal de un parámetro de control en la descripción anterior, puede conmutar entre dos de los patrones 1 a 3 de acuerdo con el tipo de señal. En otras palabras, la unidad de control de contexto 142 o 242 puede conmutar entre usar y no usar la condición de bloque superior, de acuerdo con un tipo de señal de un parámetro de control.

45 Adicionalmente, la unidad de control de contexto 142 o 242 puede cambiar un procedimiento para conmutar entre los modelos de contexto seleccionados de tal manera (incluyendo un caso en el que se cambia el incremento del modelo de contexto; lo mismo se aplicará de aquí en adelante) de acuerdo con la información de imagen predeterminada. Por ejemplo, la unidad de control de contexto 142 o 242 puede conmutar aún más la política de cambio en sí misma, de acuerdo con la cantidad de memoria, o el tamaño del ancho horizontal o un formato de muestreo de una imagen que afecta el número de entrenamientos de cada contexto.

50 Aunque la unidad de control de contexto 142 o 242 cambia entre usar y no usar la condición de bloque superior como la descripción simplificada, la unidad de control de contexto 142 o 242 puede combinar un caso en el que el bloque superior no está disponible para la conmutación y aplicar el caso combinado. Por ejemplo, la unidad de control de contexto 142 o 242 puede cambiar la política de cambio en sí misma, de acuerdo con si un corte a procesar es o no un corte de entropía (entropy_slice_flag indica 1 o 0). De manera similar, cuando no se puede garantizar la disponibilidad del bloque vecino superior, la unidad de control de contexto 142 o 242 puede cambiar la política de conmutación para no usar el bloque superior.

55

Por ejemplo, como se ilustra en la FIG. 8, la unidad de control de contexto 142 o 242 puede conmutar la política de determinación del modelo de contexto entre el primer criterio de determinación (S302) y el segundo criterio de determinación (S303), de acuerdo con un valor de un parámetro de una unidad predeterminada. En este punto, "de acuerdo con un valor de un parámetro de una unidad predeterminada" significa de acuerdo con si un corte es un corte de entropía como se describió anteriormente. Adicionalmente, el primer criterio de determinación es un criterio basado en el cual los procedimientos de la FIG. 7 se realizan. El segundo criterio de determinación es un criterio que excluye la Etapa S204 en la FIG. 7, y es, por ejemplo, un criterio convencional. Esto es equivalente a determinar el incremento del índice de contexto, usando un parámetro de una unidad local predeterminada y un valor de un parámetro de una unidad mayor que la unidad local predeterminada.

En otras palabras, la unidad de control de contexto 142 o 242 puede conmutar de un criterio de determinación que se aplicará a una unidad más pequeña que la primera unidad, a otro criterio de determinación basado en un valor de un parámetro de control de la primera unidad.

FIG. 9A ilustra la información de mapeo y la asignación (atribución) de índices de contexto. La FIG. 9A indica un ejemplo de una señal `mvd_I0`, `I1`, `Ic`. Lo mismo se aplica a los otros tipos de señal.

La Asignación 901B en la FIG. 9A es la asignación de índices de contexto usados en NPL 1. Los 14 valores de desplazamiento de 0 a 13 se asignan a imágenes P. Adicionalmente, los 14 valores de desplazamiento de 14 a 27 se asignan a imágenes B. En este punto, cada uno de `mvd_I0[ ][0]` and `mvd_I0[ ][1]` es un valor de componente (direcciones horizontal y vertical) de una diferencia entre vectores. En HEVC que se encuentra actualmente en estudio, tres valores de desplazamiento 0 a 2, tres valores de desplazamiento 7 a 9, tres valores de desplazamiento 14 a 16, y tres valores de desplazamiento 21 a 23 se asignan como valores condicionales de contexto (condiciones que se detallarán de acuerdo con `condA` y `condI`) para calcular un binario del primer elemento (`binIdx = 0`) de una secuencia binaria. Adicionalmente, la relación entre los tipos de señal y los índices de contexto se fija independientemente de varias secuencias de imagen.

Cada una de las asignaciones 902B a 904B de índices de contexto en la FIG. 9A es la asignación de índices de contexto de acuerdo con la realización 2.

La asignación 902B indica la asignación de índices de contexto cuando se usa el patrón 2 (sin usar el bloque superior). En este punto, no es necesario asignar los tres índices de contexto de 0 a 2 y otros como valores condicionales, pero dos índices de contexto de 0 a 1 son suficientes. Esto se debe a que `condA` no se usa. De este modo, no es necesario asignar los índices de contexto a las porciones sombreadas en la FIG. 9A. De este modo, incluso cuando se asignan cuatro índices de contexto de la misma manera a un `binIdx > 0` como la asignación 901B, 24 índices de contexto de 0 a 23 son suficientes en total. De este modo, se pueden reducir al menos cuatro contextos.

La asignación 903B indica la asignación de índices de contexto cuando se usa el patrón 3 (sin usar el bloque superior ni el bloque izquierdo). En este punto, no es necesario asignar los tres índices de contexto de 0 a 2 y otros como valores condicionales, pero solo un índice de contexto 0 es suficiente. Esto se debe a que no se utiliza `condA` ni `condI`. De este modo, no es necesario asignar los índices de contexto a las porciones sombreadas en la FIG. 9A. De este modo, 20 índices de contexto de 0 a 19 son suficientes en total. De este modo, se pueden reducir al menos ocho contextos.

La asignación 904B indica un ejemplo de asignación de índices de contexto cuando se construye una secuencia de imágenes sin incluir ninguna imagen B o cuando solo se usa la referencia directa como una unidad más grande que una unidad de un bloque del tipo de señal. En este punto, "cuando solo se usa la referencia directa" significa un caso en el que una secuencia de imágenes incluye solo imágenes I y P, o un caso en el que solo se usa referencia directa para imágenes B incluidas en la secuencia de imágenes. En tal caso, el índice de contexto para las imágenes B no tiene que usarse en primer lugar. Esto se debe a que no es necesario distinguir un contexto para las imágenes P de un contexto para las imágenes B cuando solo se utiliza la referencia directa.

De este modo, 10 índices de contexto (valores relativos) de 0 a 9 son suficientes como se ilustra en la FIG. 9A. De este modo, se pueden reducir al menos 18 contextos. En otras palabras, se reduce el número de contextos a inicializar o actualizar.

Aunque `mvd` se describe como un ejemplo en el presente documento, lo mismo es válido para la asignación de contextos para los otros tipos de señal en los que se utiliza el contexto para imágenes B y el contexto para imágenes P.

Conmutar el criterio como se describe para la FIG. 8 puede implicar la conmutación de una de las asignaciones 901B a 904B a usar, de acuerdo con el tipo de parámetro para toda o una parte de la secuencia de imágenes (unidad predeterminada).

De acuerdo con la realización 2, los índices de contexto asignados de acuerdo con un criterio estático (usando los bloques superior e izquierdo) como se usan convencionalmente se pueden cambiar de acuerdo con los criterios. De este modo, además de la reducción en el uso de memoria, la política de asignación de los contextos puede conmutarse de acuerdo con las características de la unidad predeterminada según sea necesario.

Adicionalmente, la unidad de control de contexto 142 o 242 puede cambiar el criterio de determinación a usar, de acuerdo con las características de un sistema de imagen. Por ejemplo, la unidad de control de contexto 142 o 242 puede cambiar el criterio de determinación a usar, de acuerdo con los intervalos de las imágenes I (valores de configuración de IntraPeriod).

- 5 Aunque la unidad de control de contexto 142 o 242 conmuta entre los criterios de determinación de acuerdo con las condiciones anteriores, puede conmutar si se usa o no el bloque superior.

- Adicionalmente, la unidad de control de contexto 142 o 242 puede determinar si se usa o no un parámetro de control del bloque superior, de acuerdo con si el parámetro de control del bloque superior está disponible en la codificación o decodificación basándose en una posición del parámetro de control. En otras palabras, la unidad de control de contexto 142 o 242 puede determinar si el parámetro de control del bloque superior está disponible o no en la decodificación, basándose en una posición del bloque actual y determina un contexto usando uno de los patrones 2 y 3 cuando el parámetro de control del bloque superior no está disponible. Adicionalmente, la unidad de control de contexto 142 o 242 puede determinar si un valor de referencia del bloque superior está disponible o no basado en una estructura de árbol para dividir bloques TU, CU o PU. En otras palabras, la unidad de control de contexto 142 o 242 puede determinar si el parámetro de control del bloque superior está disponible o no en la decodificación, de acuerdo con la profundidad jerárquica de una unidad de datos a la que pertenece cada uno de los parámetros de control a procesar.

- La FIG. 9B ilustra una relación entre una imagen, cortes y bloques de acuerdo con el estándar HEVC. Una imagen se divide en uno o varios cortes. En el ejemplo de la FIG. 9B, la imagen se divide en dos cortes (CORTE 1 y CORTE 2). Uno de los cortes incluye los bloques 301 (por ejemplo, bloques de árboles). En este punto, el bloque 301 es la unidad más grande como una determinada unidad de control cuando un corte se divide en un tamaño predeterminado, y tiene un tamaño de raíz cuando la unidad está en la raíz en la estructura dividida jerárquicamente.

En el ejemplo de la FIG. 9B, el CORTE 2 comienza desde un bloque 301A, y se compone de una secuencia que incluye bloques en la esquina inferior derecha de la imagen a través de los bloques sombreados 301B y 301C. Uno de los bloques sombreados de la FIG. 9B es un bloque (TreeBlock) a procesar actualmente.

- 25 Cada uno de los bloques 301 incluye N x M píxeles. Uno de los bloques 301 está dividido recursivamente dentro (típicamente en cuatro). En otras palabras, un TreeBlock conceptualmente compone un árbol cuádruple. En el bloque 301B en la FIG. 9B, el bloque superior derecho obtenido al dividir el bloque sombreado 301B en cuatro se divide recursivamente en cuatro bloques dos veces. En otras palabras, el bloque 301B incluye 10 unidades lógicas desde la unidad cero-superior hasta la novena unidad inferior derecha que se dividen con una cierta perspectiva.
- 30 En este punto, la perspectiva indica el concepto de una pluralidad de árboles que tienen diferentes profundidades con una raíz como punto base, tal como un árbol con respecto a una unidad de codificación (CU) y un árbol con respecto a residual_data. En este punto, un valor de cada parámetro de control pertenece a uno de los nodos hoja.

- En este punto, si un valor de un parámetro de control de cierto tipo de señal incluido en un bloque superior está realmente disponible depende de un tipo de árbol al que pertenece el parámetro de control. De este modo, la unidad de control de contexto 142 o 242 puede cambiar un criterio de determinación de acuerdo con un tipo de árbol al que pertenece el parámetro de control. Este cambio es equivalente al cambio en una unidad de sintaxis. Por ejemplo, la unidad de control de contexto 142 o 242 puede usar el patrón 2 o 3 en el que el bloque superior no se usa para datos de un filtro adaptativo con una estructura de datos tal como alf_param, mientras que puede usar la política del modelo de contexto (patrón 1) para las otras sintaxis como se usa convencionalmente. En otras palabras, el segundo tipo o el tercer tipo puede ser un parámetro de control que tiene una estructura de datos predeterminada. Adicionalmente, esto significa que el criterio de determinación puede cambiarse de acuerdo con el tipo de árbol de un bloque vecino.

- Adicionalmente, si el valor del parámetro de control se puede usar realmente o no, o si la ventaja de reducir el uso de memoria difiere dependiendo de la posición de un bloque en la relación jerárquica. En otras palabras, la unidad de control de contexto 142 o 242 puede conmutar entre usar o no usar el bloque superior, de acuerdo con una profundidad de un bloque y una posición jerárquica del bloque.

Por ejemplo, en la FIG. 9B, los números del 0 al 9 en el bloque 301B están en orden de decodificación. En este caso, los parámetros de control de los bloques 1 y 2 están disponibles cuando el bloque 4 se codifica o decodifica.

- Adicionalmente, para reducir el uso de memoria, la unidad de control de contexto 142 o 242 puede seleccionar el patrón 1 usando el bloque superior, cuando el bloque no está en una profundidad 0 y la posición propia es uno de los segundos a los elementos subsiguientes en la partición vertical. En este punto, "profundidad" indica la profundidad desde la raíz. En otras palabras, cuando un determinado bloque se define como bloque [xn],[y0][depth], el criterio de determinación puede cambiarse de acuerdo con si el bloque actual satisface o no el bloque[xn][(y0) + 1][depth]. En otras palabras, los bloques superiores se usan para los bloques 4 a 9 en la FIG. 9B. Cuando el árbol se codifica o decodifica en el orden numerado (comenzando desde 0 y terminando en 9), está claro que los bloques 4 a 9 pueden usar los parámetros de control incluidos en los bloques superiores. Adicionalmente, hay una ventaja de que estos bloques solo tienen que contener temporalmente los datos. Adicionalmente, esto indica que el valor del contexto se determina de acuerdo con la posición 3D, incluida la profundidad además de las coordenadas x e y. En este punto, el valor de contexto puede determinarse de acuerdo con al menos una posición entre las coordenadas X e Y y una

profundidad.

Asimismo, un valor condicional de un bloque en la capa superior puede usarse (seguido) como un valor condicional de un bloque en la capa inferior. De manera similar, un valor condicional de un bloque en la capa inferior puede usarse (seguido) como un valor condicional de un bloque en la capa superior.

- 5 Como una extensión de esto, se puede usar el siguiente procedimiento, en lugar de mantener un valor para una fila inferior por unidad mínima (por ejemplo, 4 x 4) como se describe para la FIG. 38. Por ejemplo, un valor condicional o un valor para derivar el valor condicional se mantienen en este procedimiento por unidad más grande que la unidad mínima, tal como 32 x 32 o 64 x 64. En este procedimiento, el valor de la fila inferior se calcula usando valores de dos bloques en la fila superior con interpolación o división interior de un valor por encima del bloque actual, usando la
- 10 relación de posición entre los dos bloques y el bloque actual. Por ejemplo, en este procedimiento, los valores de muestra se mantienen para la fila superior por unidad, por encima de la unidad mínima (por ejemplo, 4 x 4). Los valores de muestra para la fila superior se derivan con la división interior, de acuerdo con la posición de un bloque (4 x 4 a 64 x 64, etc.) en la fila inferior. Aunque la cantidad de cálculo aumenta, el uso de memoria se puede reducir con esta estructura. En consecuencia, en comparación con el caso donde los valores de la muestra se mantienen por unidad
- 15 mínima para el peor de los casos, el uso de memoria se puede reducir para la fila superior en este procedimiento. Por ejemplo, cuando los valores de la muestra se mantienen en una unidad por unidad de 32 x 32, el uso de memoria se reduce a una octava parte de eso cuando se usa la unidad mínima (4 x 4).

- Adicionalmente, por ejemplo, `split_coding_unit_flag` es una bandera que indica si un bloque tiene cuatro particiones o no. El contexto para este indicador (`split_condig_unit_flag`) puede determinarse de acuerdo con una profundidad
- 20 jerárquica del bloque actual y manteniendo el valor condicional `condI` para el bloque izquierdo, sin usar el valor condicional (`condA`) para el bloque superior. Adicionalmente, cuando un factor dominante de un tipo de señal no es el bloque superior o el bloque izquierdo sino una profundidad, la unidad de control de contexto 142 o 242 puede usar la profundidad sola como condición para determinar un contexto. En otras palabras, la unidad de control de contexto 142 o 242 puede determinar un contexto para los otros tipos de señal usando una profundidad, sin referirse al bloque superior y al bloque izquierdo.
- 25

Adicionalmente, la unidad de control de contexto 142 o 242 puede cambiar estos criterios en consideración de la relación de posición entre el bloque actual y los otros cortes. De aquí en adelante, los tres bloques tramados 301A, 301B y 301C en la FIG. 9B se describirá.

- En este punto, el bloque 301A es un bloque de inicio, y tanto el bloque izquierdo como el bloque superior del bloque 301A están incluidos en otro corte. El bloque superior del bloque 301B está incluido en otro corte. Tanto el bloque izquierdo como el bloque superior del bloque 301C están incluidos en el mismo corte que incluye el bloque 301C. La unidad de control de contexto 142 o 242 puede conmutar el criterio de acuerdo con tal condición. En otras palabras, la unidad de control de contexto 142 o 242 puede conmutar el criterio (i) de acuerdo con si el bloque superior está
- 30 incluido o no en otro corte, (ii) de acuerdo con si el bloque izquierdo está incluido o no en otro corte, o (iii) de acuerdo con ambos (i) y (ii). En otras palabras, la unidad de control de contexto 142 o 242 puede determinar que el parámetro de control del bloque superior no está disponible en la decodificación cuando el bloque actual está en el límite del corte. Por consiguiente, cuando no se completa el procesamiento de decodificación en el CORTE 1 superior, por ejemplo, es posible realizar el procesamiento de decodificación considerando si CORTE 2 puede o no obtener información por sí mismo.
- 35

- 40 A continuación, se describirá la unidad de procesamiento jerárquico (estructura de bloques multicapa). La FIG. 10 ilustra la unidad de procesamiento jerárquico (estructura de bloque multicapa).

- El aparato de codificación de imagen 100 codifica imágenes en movimiento en una por unidad de procesamiento, y el aparato de codificación de imagen 200 decodifica un flujo codificado en una por unidad de procesamiento. La unidad de procesamiento se estratifica dividiendo la unidad de procesamiento en unidades de procesamiento pequeñas y dividiendo aún más las unidades de procesamiento pequeñas en unidades de procesamiento más pequeñas. Como
- 45 la unidad de procesamiento es más pequeña, la profundidad de la unidad de procesamiento es mayor y es jerárquicamente menor, y el valor que indica la profundidad es mayor. En cambio, como la unidad de procesamiento es más grande, la profundidad de la unidad de procesamiento es menor y es jerárquicamente superior, y el valor que indica la profundidad es menor.

- 50 La unidad de procesamiento incluye una unidad de codificación (CU), una unidad de predicción (PU) y una unidad de transformación (TU). Un CU es un bloque de 128 x 128 píxeles como máximo, y es una unidad correspondiente a un macrobloque convencional. Un PU es una unidad básica para la interpretación. Una TU es una unidad básica para la transformación ortogonal, y tiene un tamaño idéntico al de la PU o mucho más pequeño que la PU. Una CU es, por ejemplo, dividida en 4 subCU, y una de las subCU incluye una PU y una TU que tienen el mismo tamaño que el de la sub-CU (aquí, PU y TU se superponen entre sí). Por ejemplo, la PU se divide en 4 subPU y la TU se divide más en 4 subCU. Cuando la unidad de procesamiento se divide en unidades de procesamiento más pequeñas, cada una de las unidades de procesamiento más pequeñas se denomina unidad de subprocesamiento. Por ejemplo, cuando la unidad de procesamiento es una CU, la unidad de subprocesamiento es una subCU. Cuando la unidad de procesamiento es una PU, la unidad de subprocesamiento es una subPU. Adicionalmente, cuando la unidad de procesamiento es una
- 55

TU, la unidad de subprocesamiento es una subTU.

Más específicamente, a continuación, se indican los detalles.

Una imagen se divide en uno o varios cortes. Un corte es una secuencia de la unidad de codificación más grande. La posición de la unidad de codificación más grande se indica mediante una dirección de la unidad de codificación más grande lcuAddr.

Cada una de las unidades de codificación, incluidas las unidades de codificación más grandes respectivas, se divide en cuatro unidades de codificación. Como resultado, se construye un árbol cuádruple que tiene el tamaño de una CU. La posición de la CU se indica mediante un índice de unidad de codificación culdx que tiene una muestra (píxel o coeficientes) en la esquina superior izquierda de la unidad de codificación más grande como punto de partida.

Cuando no se permite la partición de una CU, la CU se maneja como una PU. De manera similar a la CU, la posición de un PU se indica mediante un índice de unidad de predicción puldx que tiene una muestra en la esquina superior izquierda de la unidad de codificación más grande como punto de partida.

La PU puede incluir particiones (particiones PU o subPU). La partición PU se indica mediante un índice de partición de unidad de predicción puPartIdx que tiene una muestra en la esquina superior izquierda de la PU como punto de partida.

La PU puede incluir TU. De manera similar a la CU, la TU puede dividirse en cuatro TU más pequeñas (subTU). Esto indica el permiso de la división del árbol cuádruple de una señal residual. La posición de la TU se indica mediante un índice de unidad de transformación tuldx que tiene una muestra en la esquina superior izquierda de la PU como punto de partida.

En este punto, la definición de cada una de las unidades de procesamiento es la siguiente:

CTB (bloque de árbol de codificación): Unidad básica para identificar la división del árbol cuádruple de una región cuadrada. Tener diversos tamaños cuadrados;

LCTB (bloque de árbol de codificación más grande): El CTB más grande permitido en un corte. Un corte incluye una pluralidad de LCTB que no se superponen entre sí;

SCTB (bloque de árbol de codificación más pequeño): El CTB más pequeño permitido en un corte. No se permite la división de un SCTB en CTB más pequeños;

PU (predicción unidad): Unidad básica para identificar el procesamiento de predicciones. Una PU es tan grande como una CU en la que no se permite la división. Aunque se permite dividir una CU en cuatro regiones cuadradas, una PU puede dividirse en una pluralidad de divisiones que tienen cualquier forma;

TU (unidad de transformación): Unidad básica para identificar transformación y cuantización;

CU (unidad de codificación): Igual que CTB;

LCU (unidad de codificación más grande): Igual que la CTB más grande; y

SCU (unidad de codificación más pequeña): Igual que el CTB más pequeño.

Adicionalmente, los parámetros de cuantización incluyen al menos uno de un parámetro de escala de cuantificación delta (delta QP o QP delta), un parámetro de desplazamiento de cuantización, un índice (Q matrix select idc) y un parámetro de desplazamiento de zona muerta de cuantización. El índice es para seleccionar una de las matrices de escala cuantizadas.

El parámetro de escala de cuantización delta (delta QP o QP delta) es una diferencia entre un parámetro de escala de cuantización que se aplicará a los coeficientes de transformación y un parámetro de escala de cuantización especificado por un encabezado de secuencia o un encabezado de corte (o un parámetro de escala de cuantización inmediatamente antes en orden de exploración Z).

El parámetro de desplazamiento de cuantización también se conoce como compensación de cuantización, y es un valor de ajuste (valor de desplazamiento) para redondear una señal al realizar la cuantización. De este modo, cuando el aparato de codificación de imagen 100 realiza la cuantización, codifica el desplazamiento de cuantización. Seguidamente, el aparato de decodificación de imagen 200 decodifica el desplazamiento de cuantización codificado. A continuación, el aparato de decodificación de imagen 200 realiza la corrección usando el desplazamiento de cuantización cuando cuantiza inversamente los coeficientes de transformación.

Un índice (Qmatrix select idc) se denomina matriz de cuantización adaptativa e indica qué matriz de escala de cuantización se usa entre una pluralidad de matrices de escala de cuantización. Adicionalmente, cuando solo hay una matriz de escala de cuantización, Qmatrix select idc indica si se usa o no la matriz de escala de cuantización. La matriz de cuantización adaptativa se puede controlar por unidad de bloque (unidad de procesamiento).

El parámetro de desplazamiento de zona muerta de cuantización se denomina zona muerta adaptativa y es información de control para cambiar adaptativamente una zona muerta por bloque. La zona muerta es un ancho cuyos coeficientes de frecuencia se convierten en 0 por cuantización (último ancho que se convierte en +1 o -1 después de la cuantización).

Aunque un caso en el que el patrón 3 con el que se usa un valor fijo predeterminado como valor de contexto se describe aquí anteriormente, el caso puede realizarse a condición de que los parámetros de control del bloque superior y el bloque izquierdo no se usen, y además a condición de no usar los parámetros de control del bloque superior y el bloque izquierdo como patrón 3. Por ejemplo, la unidad de control de contexto 142 o 242 puede determinar un contexto de acuerdo con la profundidad jerárquica de una unidad de datos a la que pertenece cada uno de los parámetros de control, como el patrón 3.

(Realización 3)

La realización 3 describirá qué tipo de señal debe usarse como el primer tipo y el segundo tipo (o el tercer tipo).

Más específicamente, los presentes inventores han verificado cada uno de los tipos de señal a continuación entre los tipos de señal indicados en la FIG. 3 (Sección 9.3.3.1.1.1 de NPL 2). Cada uno de los tipos de señal ha sido verificado, porque hay diversos parámetros y es difícil predecir si cada patrón de los otros tipos de señal satisface o no la validez, basándose en un resultado de la verificación en uno de los tipos de señal (cuál de los patrones 1 a 3 es apropiado).

La verificación está en conformidad con la estructura (parámetro de configuración y versión de software HM3.0) descrita en JCTVC-E700, "Condiciones de prueba comunes y configuraciones de referencia de software" (ver NPL 3). Adicionalmente, cada una de las imágenes de prueba tiene una longitud limitada a 49 tramas.

El procedimiento de codificación de imagen y el procedimiento de decodificación de imagen de acuerdo con la realización 3 se refieren a CABAC. De este modo, la verificación se realizó usando los siguientes cuatro patrones de prueba que son un conjunto de valores de configuración, cada uno de los cuales indica 1 como el valor del Modo de Símbolo (#0: LCEC, 1: CABAC):

- 4.1 Intra, ajuste de alta eficiencia;
- 4.3 Acceso aleatorio, ajuste de alta eficiencia;
- 4.5 Bajo retraso, ajuste de alta eficiencia; y
- 4.7 Bajo retraso, ajuste de alta eficiencia (P cortes solo).

La evaluación se realiza basándose en un valor de evaluación llamado "tasa BD" que se usa como un estándar de evaluación usado de manera uniforme para una evaluación de implementación en HEVC. Tasa BD Y, tasa BD U y tasa BD V son tasas BD para un espacio de color YUV y son valores estándar de evaluación. De acuerdo con VCEG-A111 (NPL 4), la tasa BD es un valor de evaluación obtenido al integrar dos pares de cantidades de código con un resultado de PSNR, y que representa la eficiencia de codificación de acuerdo con la relación de área. Adicionalmente, la tasa BD que indica un valor negativo significa que se ha mejorado la eficiencia de codificación. Los criterios de comparación se basan en un resultado de la salida de un programa de referencia que implementa el patrón 1. Los resultados de los patrones 2 y 3 se muestran con respecto al resultado del patrón 1.

A continuación, se describe el resultado de la verificación en cada uno de los tipos de señal:

- (Primera verificación) split_coding_unit_flag;
- (Segunda verificación) skip_flag;
- (Tercera verificación) merge_flag;
- (Cuarta verificación) ref_idx_l0 (l1, lc);
- (Quinta verificación) inter_pred_flag;
- (Sexta verificación) mvd_l0 (l1, lc);
- (Séptima verificación) no_residual_data_flag;
- (Ochava verificación) intra_chroma_pred_mode; y
- (Novena verificación) cbf_luma, cbf_cr, cbf_cb.

(Primera verificación) split_coding_unit_flag

La FIG. 11 ilustra un procedimiento de decodificación aritmética para split_coding_unit_flag.

La verificación se realiza cambiando el modelo de contexto del patrón 1 al patrón 2 o 3 solo para que se verifique un tipo de señal, sin cambiar el modelo de contexto para los otros tipos de señal y el parámetro de verificación especificado en NPL 3. En la columna de la FIG. 11, el valor de "Fijado" indica que la condición (la condición de bloque izquierdo o la condición de bloque superior) de la columna especificada por "Fijado" no se usa cuando se deriva un valor de contexto (o incremento). En otras palabras, cuando solo una de las condiciones del bloque izquierdo y la condición del bloque superior es "Fijado", solo se usa la otra condición. Adicionalmente, cuando tanto la condición del bloque izquierdo como la condición del bloque superior son "Fijadas", un valor predeterminado (por ejemplo, 0) se usa como un valor de contexto (o incremento).

El significado del tipo de señal "split_coding_unit_flag" se define de la siguiente manera.

split_coding_unit_flag[x0][y0] especifica si una unidad de codificación se divide en unidades de codificación con tamaño medio horizontal y vertical. Los índices de matriz x0, y0 especifica la ubicación (x0, y0) de la muestra de

luminancia superior izquierda del bloque de codificación considerado en relación con la muestra de luminancia superior izquierda de la imagen. En otras palabras, "split_coding_unit_flag" indica si la CU de destino se divide o no en cuatro. Más específicamente, la CU de destino se divide cuando split_coding_unit_flag indica 1, mientras que la CU de destino no está dividida cuando split_coding_unit_flag indica 0.

- 5 Los datos de split_coding_unit_flag se estructuran en una sintaxis de árbol de codificación como sintaxis. El aparato de decodificación de imagen analiza una secuencia de bits de acuerdo con la sintaxis de esta estructura de datos.

Las figs. 12A y 12B son tablas para describir los resultados de la verificación en split_coding_unit_flag.

- 10 La FIG. 12A indica el resultado de la verificación usando un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L) del patrón 2. La FIG. 12B indica el resultado de la verificación usando un bloque vecino cero (sin usar la condición de bloque superior L ni la condición de bloque izquierdo L) del patrón 3.

El resultado de la verificación en cada una de las Figs. 12A y 12B indican el incremento y la disminución de la tasa BD de acuerdo con los cuatro patrones de prueba.

- 15 Adicionalmente, el valor de evaluación está representado por el estándar de evaluación que indica un valor relativo a un valor de evaluación en el caso del patrón 1 en el que se usan tanto el bloque izquierdo como el bloque superior. Más específicamente, cuando el valor de evaluación es positivo, el resultado es inferior al valor de evaluación (tasa BD) en el caso del patrón 1. Adicionalmente, cuando el valor de evaluación es negativo, el resultado es más mejorado que el valor de evaluación en el caso del patrón 1.

El resultado aclara que el patrón 1 es superior como patrón de un modelo de contexto para split_coding_unit_flag. En otras palabras, los valores de evaluación obtenidos por los patrones 2 y 3 son inferiores a los del patrón 1.

- 20 De este modo, cuando el tipo de señal de un parámetro de control es split_coding_unit_flag, la unidad de control de contexto 142 o 242 determina un valor de contexto usando el patrón 1 que es un patrón convencional de un modelo de contexto, en términos de la tasa BD.

(Segunda verificación) skip_flag

- 25 La FIG. 13 ilustra un procedimiento de decodificación aritmética para skip_flag. En este punto, el procedimiento de verificación es el mismo que en la primera verificación.

El significado del tipo de señal "skip_flag" se define de la siguiente manera.

- 30 skip_flag[x0][y0] igual a 1 especifica que, para la unidad de codificación actual, al decodificar un corte P o B, no se analizan más elementos de sintaxis, excepto los índices de predicción del vector de movimiento después de skip_flag[x0][y0]. skip_flag [x0][y0] igual a 1 especifica que la unidad de codificación no se debe saltar. Los índices de matriz x0, y0 especifica la ubicación (x0, y0) de la muestra de luminancia superior izquierda del bloque de codificación considerado en relación con la muestra de luminancia superior izquierda de la imagen. En otras palabras, skip_flag indica si se debe saltar o no la CU de destino (se maneja como un bloque saltado).

- 35 Los datos de skip_flag se estructuran en una sintaxis de unidad de codificación como sintaxis. En otras palabras, skip_flag se establece para cada CU. El aparato de decodificación de imagen analiza una secuencia de bits de acuerdo con la sintaxis de esta estructura de datos.

Las figs. 14A y 14B son tablas para describir los resultados de la verificación en skip_flag.

La FIG. 14A indica el resultado de la verificación usando un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L) del patrón 2. La FIG. 14B indica el resultado de la verificación usando un bloque vecino cero (sin usar la condición de bloque superior L ni la condición de bloque izquierdo L) del patrón 3.

- 40 El resultado de la verificación en cada una de las Figs. 14A y 14B indican el incremento y la disminución de la tasa BD de acuerdo con los cuatro patrones de prueba como se describe para la primera verificación. Adicionalmente, el significado del valor de evaluación es el mismo que el de la primera verificación.

El resultado aclara que el patrón 1 es superior como patrón de un modelo de contexto para "skip_flag". En otras palabras, los valores de evaluación obtenidos por los patrones 2 y 3 son inferiores a los del patrón 1.

- 45 De este modo, cuando el tipo de señal de un parámetro de control es "skip_flag", la unidad de control de contexto 142 o 242 determina un valor de contexto usando el patrón 1 que es un patrón convencional de un modelo de contexto, en términos de la tasa BD.

(Tercera verificación) "merge_flag"

- 50 La FIG. 15 es una tabla que indica un procedimiento de decodificación aritmética para merge_flag. En este punto, el procedimiento de verificación es el mismo que el de la primera verificación y el de la segunda verificación.

El significado del tipo de señal "merge_flag" se define de la siguiente manera.

merge_flag[x0][y0] especifica si los parámetros de interpredicción para la unidad de predicción actual se infieren de una división de interpredicción vecina. Los índices de matriz x0, y0 especifica la ubicación (x0, y0) de la muestra de luminancia superior izquierda del bloque de predicción considerado en relación con la muestra de luminancia superior izquierda de la imagen. Cuando merge_flag[x0][y0] no está presente (InferredMergeFlag es igual a 1), se infiere que es igual a 1. En otras palabras, merge_flag[x0][y0] indica si se usa o no un modo de fusión. En este punto, el modo de fusión es un modo en el que un vector de movimiento y un índice de imagen de referencia se copian de un bloque vecino del bloque actual a codificar y el bloque actual se codifica.

Los datos de merge_flag se estructuran en una unidad de predicción como una sintaxis. En otras palabras, merge_flag se establece para cada PU. El aparato de decodificación de imagen analiza una secuencia de bits de acuerdo con la sintaxis de esta estructura de datos.

Las figs. 16A y 16B son tablas para describir los resultados de la verificación en merge_flag.

La FIG. 16A indica el resultado de la verificación usando un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L) del patrón 2. La FIG. 16B indica el resultado de la verificación usando un bloque vecino cero (sin usar la condición de bloque superior L ni la condición de bloque izquierdo L) del patrón 3.

El resultado de la verificación en cada una de las Figs. 16A y 16B indican el incremento y la disminución de la tasa BD de acuerdo con los cuatro patrones de prueba como se describe para la primera verificación. Adicionalmente, el significado del valor de evaluación es el mismo que el de la primera verificación.

El resultado es diferente de los de la primera verificación de split_coding_unit_flag y la segunda verificación de skip_flag. No hay una diferencia significativa en la tasa BD entre los patrones 1 y 2 o 3 como un patrón de un modelo de contexto para merge_flag.

De este modo, bajo un entorno mixto con una pluralidad de parámetros de control de tipos de señal, la unidad de control de contexto 142 o 242 determina un valor de contexto sin usar el bloque superior como un bloque vecino, particularmente cuando el tipo de señal del parámetro de control es merge_flag. En otras palabras, la unidad de control de contexto 142 o 242 determina un valor de contexto usando el patrón 2 o 3 cuando el tipo de señal del parámetro de control es merge_flag. En otras palabras, el primer tipo incluye "split_coding_unit_flag" o "skip_flag", y el segundo tipo o el tercer tipo incluye "merge_flag". Por consiguiente, el aparato de codificación de imagen y el aparato de decodificación de imagen de acuerdo con la realización 3 pueden reducir el uso de memoria mientras suprimen la disminución de la tasa BD.

Cuando el patrón 2 se compara con el patrón 3 para merge_flag, estas tasas de BD no tienen una diferencia significativa. De este modo, se prefiere usar el patrón 3 para merge_flag. Por consiguiente, es posible reducir aún más el uso de memoria y la cantidad de procesamiento.

En este punto, en comparación con merge_flag y skip_flag, aunque los datos residuales de un vector de movimiento no se transmiten en un modo de salto, los datos residuales del vector de movimiento se transmiten en modo de fusión. Por consiguiente, incluso cuando el contexto a usar temporalmente no sea óptimo para merge_flag, el deterioro de la calidad de imagen causado por no usar el contexto óptimo puede compensarse en cierta medida con el procesamiento utilizando los datos residuales. Por consiguiente, se suprime el deterioro de la calidad de la imagen.

(Cuarta verificación) "ref_idx_l0(l1,lc)"

La FIG. 17 es una tabla que indica un procedimiento de decodificación aritmética para ref_idx. En este punto, el procedimiento de verificación es el mismo que el de la primera verificación y el de la segunda verificación.

El significado del tipo de señal "ref_idx" se define de la siguiente manera.

ref_idx_l0[x0][y0] especifica el índice de imagen de referencia de la lista 0 para la unidad de predicción actual. Los índices de matriz x0, y0 especifica la ubicación (x0, y0) de la muestra de luminancia superior izquierda del bloque de predicción considerado en relación con la muestra de luminancia superior izquierda de la imagen.

Adicionalmente, ref_idx_l1[x0][y0] tiene la misma semántica que l0 y la lista 0 reemplazada por l1 y la lista 1 en ref_idx_l0. En otras palabras, ref_idx_l1 indica un índice de imagen de referencia para la lista 1 de la PU actual.

La presencia o ausencia de ref_idx_l1 se puede determinar basándose en, por ejemplo, un tipo de imagen.

Adicionalmente, ref_idx_lc[x0][y0] tiene la misma semántica que ref_idx_l0, con l0 y la lista 0 reemplazada por lc y combinación de lista, respectivamente. El ref_idx_lc es un parámetro de control agregado en HAVC. Adicionalmente, la combinación de la lista se obtiene combinando (fusionando) la lista 0 y la lista 1. Normalmente, un flujo de bits incluye solo uno de ref_idx_l0, ref_idx_l1 y ref_idx_lc. Hay casos en los que el flujo de bits incluye solo uno o ambos de ref_idx_l0 y ref_idx_l1.

Los datos de `ref_idx_l0(l1, lc)` se estructuran en una unidad de predicción como una sintaxis. En otras palabras, `merge_flag` se establece para cada PU. El aparato de decodificación de imagen analiza una secuencia de bits de acuerdo con la sintaxis de esta estructura de datos.

Las figs. 18A y 18B son tablas para describir los resultados de la verificación en `ref_idx`.

- 5 La FIG. 18A indica el resultado de la verificación usando un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L) del patrón 2. La FIG. 18B indica el resultado de la verificación usando un bloque vecino cero (sin usar la condición de bloque superior L ni la condición de bloque izquierdo L) del patrón 3.

10 El resultado de la verificación en cada una de las Figs. 18A y 18B indican el incremento y la disminución de la tasa BD de acuerdo con los cuatro patrones de prueba como se describe para la primera verificación. Adicionalmente, el significado del valor de evaluación es el mismo que el de la primera verificación.

El resultado es diferente de los de la primera verificación de `split_coding_unit_flag` y la segunda verificación de `skip_flag`. No hay una diferencia significativa en la tasa BD entre los patrones 1 y 2 o 3 como un patrón de un modelo de contexto para `ref_idx`.

15 De este modo, bajo un entorno mixto con una pluralidad de parámetros de control de tipos de señal, la unidad de control de contexto 142 o 242 determina un valor de contexto sin usar el bloque superior como un bloque vecino, particularmente cuando el tipo de señal del parámetro de control es `ref_idx_l0(l1,lc)`. En otras palabras, la unidad de control de contexto 142 o 242 determina un valor de contexto usando el patrón 2 o 3 cuando el tipo de señal del parámetro de control es `ref_idx_l0(l1,lc)`. En otras palabras, el primer tipo incluye "split_coding_unit_flag" o "skip_flag", y el segundo tipo o el tercer tipo incluye "ref_idx_l0(l1,lc)". Por consiguiente, el aparato de codificación de imagen y el
20 aparato de decodificación de imagen de acuerdo con la realización 3 pueden reducir el uso de memoria mientras suprimen la disminución de la tasa BD.

En este punto, el segundo tipo o el tercer tipo solo debe incluir al menos uno de `ref_idx_l0`, `ref_idx_l1` y `ref_idx_lc`. Por ejemplo, el segundo tipo o el tercer tipo pueden incluir `ref_idx_l0` y `ref_idx_l1` sin incluir `ref_idx_lc`.

25 Cuando el patrón 2 se compara con el patrón 3 para `ref_idx`, estas tasas de BD no tienen una diferencia significativa. De este modo, se prefiere usar el patrón 3 para `ref_idx_l0(l1, lc)`. Por consiguiente, es posible reducir aún más el uso de memoria y la cantidad de procesamiento.

30 En este punto, `ref_idx` se usa en el modo de interpredicción normal que no es un modo de salto ni un modo de fusión. Aunque el mismo vector de movimiento para el bloque superior y el bloque izquierdo se utiliza como vector de movimiento del bloque actual en el modo de salto y el modo de fusión, un vector de movimiento diferente al del bloque superior y el bloque izquierdo se usa como un vector de movimiento del bloque actual en el modo de interpredicción normal diferente del modo de salto y el modo de fusión. Por consiguiente, la eficiencia de codificación disminuye usando el bloque superior y el bloque izquierdo para `ref_idx` como en el patrón 1. En otras palabras, es posible mejorar la eficiencia de codificación usando el patrón 2 o 3 para `ref_idx`.

35 En este punto, la unidad de control de contexto 142 o 242 puede usar un valor derivado de un resultado de `condl` y un resultado de `condA` para `ref_idx_l0[xP][yP]` del bloque actual, en lugar de usar `ref_idx_lc[xL][yP]` del bloque vecino A o `ref_idx_lc[xP][yA]` del bloque vecino B, al determinar una condición A (o L) para `ref_idx_lc[xP][yP]`. En otras palabras,

la unidad de control de contexto 142 o 242 puede derivar un valor resultante de la condición como un valor dependiente de los valores de los bloques actuales `l0` y `l1`.

40 Un aparato de codificación o un aparato de grabación genera `ref_idx_lc` combinando `ref_idx_l0` y `ref_idx_l1` en la grabación o codificación de un flujo. En otras palabras, `ref_idx_l0` y `ref_idx_l1` se usan para cada determinación dentro de estos aparatos. De este modo, (1) cuando una condición `condA` o L $\{(ref_idx_l0 \text{ está disponible}) \text{ y } (verdadero \text{ se mantiene si } ref_idx_l0 > 0)\}$ y (2) la condición `condA` o L $\{(ref_idx_l1 \text{ está disponible}) \text{ y } (verdadero \text{ se mantiene si } ref_idx_l1 > 0)\}$ se mantiene en el bloque actual, la condición `condA` o L $\{(ref_idx_lc \text{ está disponible}) \text{ y } (true \text{ se mantiene si } ref_idx_lc > 0)\}$ se mantiene.

45 De este modo, se puede realizar lo siguiente. La FIG. 18C es una tabla que indica la condición A y la condición L para `ref_idx` como se indica en NPL 2. La FIG. 18D es una tabla que indica la condición A y la condición L para `ref_idx` de acuerdo con la realización 3.

50 Como se indica en la FIG. 18D, las unidades de control de contexto 142 y 242 pueden derivar los valores condicionales de `condl` y `condA` para `ref_idx_lc` de al menos uno de los valores condicionales de `ref_idx_l0` y `ref_idx_l1` en el mismo bloque. En otras palabras, la unidad de control de contexto 142 y 242 puede hacer que los valores condicionales de `condl` y `condA` para `ref_idx_lc` dependan linealmente de los valores condicionales de `ref_idx_l0` y `ref_idx_l1` en el mismo bloque.

Por consiguiente, `ref_idx_lc` no necesita referencia de memoria. En otras palabras, el valor condicional para `ref_idx_lc` se puede derivar sin hacer referencia al valor de `ref_idx_lc` para el bloque superior.

(Quinta verificación) "inter_pred_flag"

La FIG. 19 es una tabla que indica un procedimiento de decodificación aritmética para inter_pred_flag.

El significado del tipo de señal "inter_pred_flag" se define de la siguiente manera.

5 inter_pred_flag[x0][y0] especifica si se usa unipredicción o bipredicción para la unidad de predicción actual de acuerdo con la Tabla 7 11. Los índices de matriz x0, y0 especifica la ubicación (x0, y0) de la muestra de luminancia superior izquierda del bloque de predicción considerado en relación con la muestra de luminancia superior izquierda de la imagen. En este punto, unipredicción es predicción usando lc (combinación de listas), y bipredicción es predicción usando la lista 0 y la lista 1. Adicionalmente, la combinación de la lista se obtiene combinando (fusionando) la lista 0 y la lista 1. Adicionalmente, inter_pred_flag se usa solo cuando el corte actual es un corte B.

10 Los datos de inter_pred_flag se estructuran en una unidad de predicción como una sintaxis. El aparato de decodificación de imagen analiza una secuencia de bits de acuerdo con la sintaxis de esta estructura de datos.

Las figs. 20A y 20B son tablas para describir los resultados de la verificación en inter_pred_flag.

15 La FIG. 20A indica el resultado de la verificación usando un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L) del patrón 2. La FIG. 20B indica el resultado de la verificación usando un bloque vecino cero (sin usar la condición de bloque superior L ni la condición de bloque izquierdo L) del patrón 3.

El resultado de la verificación en cada una de las Figs. 20A y 20B indican el incremento y la disminución de la tasa BD de acuerdo con los cuatro patrones de prueba como se describe para la primera verificación. Adicionalmente, el significado del valor de evaluación es el mismo que el de la primera verificación.

20 El resultado es diferente de los de la primera verificación de split_coding_unit_flag y la segunda verificación de skip_flag. No hay una diferencia significativa en la tasa BD entre los patrones 1 y 2 o 3 como un patrón de un modelo de contexto para inter_pred_flag.

25 De este modo, bajo un entorno mixto con una pluralidad de parámetros de control de tipos de señal, la unidad de control de contexto 142 o 242 determina un valor de contexto sin usar el bloque superior como un bloque vecino, particularmente cuando el tipo de señal del parámetro de control es inter_pred_flag. En otras palabras, la unidad de control de contexto 142 o 242 determina un valor de contexto usando el patrón 2 o 3 cuando el tipo de señal del parámetro de control es inter_pred_flag. En otras palabras, el primer tipo incluye "split_coding_unit_flag" o "skip_flag", y el segundo tipo o el tercer tipo incluye "inter_pred_flag". Por consiguiente, el aparato de codificación de imagen y el aparato de decodificación de imagen de acuerdo con la realización 3 pueden reducir el uso de memoria mientras suprimen la disminución de la tasa BD.

30 Cuando el patrón 2 se compara con el patrón 3 para inter_pred_flag, estas tasas de BD no tienen una diferencia significativa. De este modo, se prefiere usar el patrón 3 para inter_pred_flag. Por consiguiente, es posible reducir aún más el uso de memoria y la cantidad de procesamiento.

35 En este punto, inter_pred_flag se usa en el modo de interpredicción normal que no es un modo de salto ni un modo de fusión. Aunque el mismo vector de movimiento para el bloque superior y el bloque izquierdo se utiliza como vector de movimiento del bloque actual en el modo de salto y el modo de fusión, un vector de movimiento diferente al del bloque superior y el bloque izquierdo se usa como un vector de movimiento del bloque actual en el modo de interpredicción normal diferente del modo de salto y el modo de fusión. Por consiguiente, la eficiencia de codificación disminuye usando el bloque superior y el bloque izquierdo para inter_pred_flag como en el patrón 1. En otras palabras, es posible mejorar la eficiencia de codificación usando el patrón 2 o 3 para inter_pred_flag.

40 Adicionalmente, como se ha descrito anteriormente, es posible mejorar aún más la eficiencia de codificación determinando un valor de contexto de acuerdo con la profundidad del bloque actual para inter_pred_flag.

(Sexta verificación) "mvd_l0 (l1,lc)"

La FIG. 21 es una tabla que indica un procedimiento de decodificación aritmética para mvd_l0(l1,lc). En este punto, el procedimiento de verificación es el mismo que el de la primera verificación y el de la segunda verificación.

45 El significado del tipo de señal "mvd_l0 (l1, lc)" se define de la siguiente manera.

50 mvd_l0[x0][y0][complx] especifica la diferencia entre un componente de vector de lista 0 a usar y su predicción. Los índices de matriz x0, y0 especifica la ubicación (x0, y0) de la muestra de luminancia superior izquierda del bloque de predicción considerado en relación con la muestra de luminancia superior izquierda de la imagen. La diferencia del componente del vector de movimiento horizontal se asigna compldx = 0 y el componente del vector de movimiento vertical se asigna compldx = 1. Cuando alguno de los dos componentes no está presente, el valor inferido es 0. En otras palabras, mvd_l0 representa una diferencia entre un vector de movimiento en una posición PU (xP, yP) y el vector predicho, usando un primer componente (componente horizontal compldx = 0) y un segundo componente (componente vertical compldx = 1).

`mvd_l1[x0][y0][complx]` tiene la misma semántica que `l0` y la lista 0 reemplazada por `l1` y la lista 1 en `mvd_l0`, respectivamente. La presencia o ausencia de `mvd_l1` se puede determinar basándose en un tipo de imagen y otros.

Adicionalmente, `mvd_lc[x0][y0][complx]` tiene la misma semántica que `mvd_l0`, con `l0` y lista 0 reemplazado por `lc` y combinación de lista, respectivamente. En otras palabras, `mvd_lc` se genera combinando `mvd_l0` y `mvd_l1`.

- 5 El término "mvd" incluye al menos `mvd_l0` e incluye al menos uno de `mvd_l1` y `mvd_lc` de acuerdo con una condición de una imagen.

Los datos de mvd se estructuran en una unidad de predicción como una sintaxis. El aparato de decodificación de imagen analiza una secuencia de bits de acuerdo con la sintaxis de esta estructura de datos.

Las figs. 22A y 22B son tablas para describir los resultados de la verificación en mvd.

- 10 La FIG. 22A indica el resultado de la verificación usando un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L) del patrón 2. La FIG. 22B indica el resultado de la verificación usando un bloque vecino cero (sin usar la condición de bloque superior L ni la condición de bloque izquierdo L) del patrón 3.

El resultado de la verificación en cada una de las Figs. 22A y 22B indican el incremento y la disminución de la tasa BD de acuerdo con los cuatro patrones de prueba como se describe para la primera verificación. Adicionalmente, el significado del valor de evaluación es el mismo que el de la primera verificación.

- 15 El resultado es diferente de los de la primera verificación de `split_coding_unit_flag` y la segunda verificación de `skip_flag`. No hay una diferencia significativa en la tasa BD entre los patrones 1 y 2 o 3 como un patrón de un modelo de contexto para mvd.

De este modo, bajo un entorno mixto con una pluralidad de parámetros de control de tipos de señal, la unidad de control de contexto 142 o 242 determina un valor de contexto sin usar el bloque superior como un bloque vecino, particularmente cuando el tipo de señal del parámetro de control es `mvd_l0(l1,lc)`. En otras palabras, la unidad de control de contexto 142 o 242 determina un valor de contexto usando el patrón 2 o 3 cuando el tipo de señal del parámetro de control es `mvd_l0(l1,lc)`. En otras palabras, el primer tipo incluye "split_coding_unit_flag" o "skip_flag", y el segundo tipo o el tercer tipo incluye `mvd_l0`, `mvd_l1` o `mvd_lc`. Por consiguiente, el aparato de codificación de imagen y el aparato de decodificación de imagen de acuerdo con la realización 3 pueden reducir el uso de memoria mientras suprimen la disminución de la tasa BD.

- 20
- 25

En otras palabras, el segundo tipo o el tercer tipo solo debe incluir al menos uno de `mvd_l0`, `mvd_l1` y `mvd_lc`. Por ejemplo, el segundo tipo o el tercer tipo pueden incluir `mvd_l0` y `mvd_l1` sin incluir `mvd_lc`.

Cuando el patrón 2 se compara con el patrón 3 para mvd, estas tasas de BD no tienen una diferencia significativa. De este modo, se prefiere usar el patrón 3 para `mvd_l0(l1,lc)`. Por consiguiente, es posible reducir aún más el uso de memoria y la cantidad de procesamiento.

- 30

En este punto, aunque los datos residuales (mvd) de un vector de movimiento no se transmiten en un modo de salto, los datos residuales (mvd) del vector de movimiento se transmiten en modo de fusión. Por consiguiente, incluso cuando el contexto a usar temporalmente no sea óptimo en el modo de fusión, el deterioro de la calidad de imagen causado por no usar el contexto óptimo puede compensarse en cierta medida con el procesamiento utilizando los mvd. En consecuencia, el deterioro de la calidad de la imagen se suprime cuando no se usa el bloque circundante para mvd.

- 35

Cuando el valor condicional del bloque superior o el bloque izquierdo (`condA` o `condl`) se usa de acuerdo con la condición predeterminada, se aplica la siguiente modificación.

La primera modificación es un procedimiento que usa una dependencia entre `mvd_l0`, `mvd_l1` y `mvd_lc`.

- 40 Más específicamente, la unidad de control de contexto 142 o 242 puede derivar un valor condicional de otro tipo de señal que tenga un valor condicional dependiente de los valores condicionales (`condl` o `condA`) de dos tipos de señal entre los tres tipos de señal de `mvd_l0`, `mvd_l1` y `mvd_lc`, usando los valores condicionales.

Por ejemplo, cuando un valor de `condA` para `mvd_lc` depende de los valores condicionales (un valor de `condA` para `l0` y un valor de `condA` para `l1`) de los dos tipos de señal de `mvd_l0` y `lvd_l1`, la unidad de control de contexto 142 o 242 no necesita referirse al valor de `condA` para `mvd_lc`.

- 45

La FIG. 22C es una tabla que indica la condición A y la condición L para mvd como se indica en NPL 2. La FIG. 22D es una tabla que indica la condición A y la condición L para mvd de acuerdo con la realización 3.

Como se indica en la FIG. 22D, las unidades de control de contexto 142 y 242 pueden derivar los valores condicionales de `condl` y `condA` para `mvd_lc` de al menos uno de los valores condicionales de `mvd_l0` y `mvd_l1` en el mismo bloque.

- 50 En este punto, la unidad de control de contexto 142 y 242 puede usar estas relaciones con una o ambas de la dirección horizontal (`complx = 0`) y la dirección vertical (`complx = 1`).

Adicionalmente, la unidad de control de contexto 142 y 242 puede usar la dependencia entre `complx = 0` y 1. En otras palabras, las unidades de control de contexto 142 y 242 pueden hacer que un resultado de uno de los dos valores condicionales de la dirección horizontal `mvd_l0[ ][ ][0]` y la dirección vertical `mvd_l0[ ][ ][1]` dependan del otro. En otras palabras, las unidades de control de contexto 142 y 242 pueden derivar los valores condicionales `condl` y `conda` de una de las direcciones horizontal y la dirección vertical para `mvd`, del otro de los valores condicionales para `mvd`. En este punto, de acuerdo con NPL 2, se establece un índice de contexto (incremento del índice + valor de referencia) en cada una de las direcciones horizontales `mvd_l0[ ][ ][0]`, `mvd_l1[ ][ ][0]`, y `mvd_lc[ ][ ][0]`, y las direcciones verticales `mvd_l0[ ][ ][1]`, `mvd_l1[ ][ ][1]` y `mvd_lc[ ][ ][1]`. De este modo, es posible reducir los residuos utilizando la dependencia. En otras palabras, el número de índices de contexto puede reducirse.

En este punto, los valores condicionales del bloque superior y el bloque izquierdo se usan solo para el primer bit de `mvd` de acuerdo con NPL 2. En otras palabras, la unidad de control de contexto 142 y 242 puede usar el patrón 2 o 3 para el primer bit de `mvd`. En otras palabras, la unidad de control de contexto 142 y 242 puede usar el patrón 2 o 3 para `abs_mvd_greater0_flag[complx]` que indica si una diferencia entre un vector de movimiento y el vector predicho es igual o mayor que 0.

(Séptima verificación) "no_residual_data_flag"

La FIG. 23A es una tabla que indica un procedimiento de decodificación aritmética para `no_residual_data_flag`.

El significado del tipo de señal "no_residual_data_flag" se define de la siguiente manera.

`no_residual_data_flag` igual a 1 especifica que no hay datos residuales presentes para la unidad de codificación actual. `no_residual_data_flag` igual a 0 especifica que los datos residuales están presentes para la unidad de codificación actual. Cuando `no_residual_data_flag` no está presente, se deducirá que su valor es igual a 0.

Los datos de `no_residual_data_flag` se estructuran en un árbol de transformación entre tipos de árboles. La FIG. 23B es una tabla que indica una sintaxis de árbol de transformación. El aparato de decodificación de imagen analiza una secuencia de bits de acuerdo con la sintaxis de esta estructura de datos.

Las figs. 24A y 24B son tablas para describir los resultados de la verificación en `no_residual_data_flag`.

La FIG. 24A indica el resultado de la verificación usando un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L) del patrón 2. La FIG. 24B indica el resultado de la verificación usando un bloque vecino cero (sin usar la condición de bloque superior L ni la condición de bloque izquierdo L) del patrón 3.

El resultado de la verificación en cada una de las Figs. 24A y 24B indican el incremento y la disminución de la tasa BD de acuerdo con los cuatro patrones de prueba como se describe para la primera verificación. Adicionalmente, el significado del valor de evaluación es el mismo que el de la primera verificación.

El resultado es diferente de los de la primera verificación de `split_coding_unit_flag` y la segunda verificación de `skip_flag`. No hay una diferencia significativa en la tasa BD entre los patrones 1 y 2 o 3 como un patrón de un modelo de contexto para `no_residual_data_flag`.

De este modo, bajo un entorno mixto con una pluralidad de parámetros de control de tipos de señal, la unidad de control de contexto 142 o 242 determina un valor de contexto sin usar el bloque superior como un bloque vecino, particularmente cuando el tipo de señal del parámetro de control es `no_residual_data_flag`. En otras palabras, la unidad de control de contexto 142 o 242 determina un valor de contexto usando el patrón 2 o 3 cuando el tipo de señal del parámetro de control es `no_residual_data_flag`. En otras palabras, el primer tipo incluye "split_coding_unit_flag" o "skip_flag", y el segundo tipo o el tercer tipo incluye "no_residual_data_flag". Por consiguiente, el aparato de codificación de imagen y el aparato de decodificación de imagen de acuerdo con la realización 3 pueden reducir el uso de memoria mientras suprimen la disminución de la tasa BD.

Cuando el patrón 2 se compara con el patrón 3 para `no_residual_data_flag`, estas tasas de BD no tienen una diferencia significativa. De este modo, se prefiere usar el patrón 3 para `no_residual_data_flag`. Por consiguiente, es posible reducir aún más el uso de memoria y la cantidad de procesamiento.

En este punto, `no_residual_data_flag` indica la presencia o ausencia de coeficientes (datos residuales) de luminancia y crominancia. Adicionalmente, los datos residuales aumentan en la intrapredicción y disminuyen en la interpredicción. De este modo, la eficiencia de codificación disminuye cuando un modo de un bloque circundante es diferente de un modo del bloque actual (el bloque circundante tiene características diferentes de las del bloque actual). Por ejemplo, cuando la intrapredicción se realiza en el bloque actual y la interpredicción se realiza en el bloque circundante, los datos residuales del bloque actual disminuyen y los datos residuales del bloque circundante aumentan. Por consiguiente, la eficiencia de codificación disminuye cuando se usa el contexto del bloque circundante. De este modo, las unidades de control de contexto 142 y 242 pueden mejorar la eficiencia de codificación usando el contexto del bloque actual sin depender del bloque circundante.

(Ochava verificación) "intra_chroma_pred_mode"

La FIG. 25A es una tabla que indica un procedimiento de decodificación aritmética para `intra_chroma_pred_mode`.

Los datos de `intra_chroma_pred_mode` se estructuran en una unidad de predicción como una sintaxis. El aparato de decodificación de imagen analiza una secuencia de bits de acuerdo con la sintaxis de esta estructura de datos.

El significado del tipo de señal "`intra_chroma_pred_mode`" se define de la siguiente manera.

- 5 `intra_chroma_pred_mode[x0][y0]` especifica el modo de intrapredicción para muestras de crominancia. Los índices de matriz `x0`, `y0` especifica la ubicación (`x0`, `y0`) de la muestra de luminancia superior izquierda del bloque de predicción considerado en relación con la muestra de luminancia superior izquierda de la imagen.

- 10 El valor del modo de intrapredicción crominancia "modo de intrapredicción crominancia" (`IntraPredModeC`) se determina combinando el valor de `intra_chroma_pred_mode` (valor entre 0 y 4 inclusive) y el `IntraPredMode [xP] [yB]` del bloque actual. En este punto, las coordenadas del bloque actual son `[xB][yB]`. `[xB][yB]` indica la misma posición que `[xP][yP]`. Adicionalmente, `IntraPredMode` indica un valor de modo de predicción para luminancia.

La FIG. 25B es una tabla que indica un procedimiento de derivación de `IntraPredModeC`, basado en `intra_chroma_pred_mode` e `IntraPredMode` que se describen en NPL 2.

- 15 Adicionalmente, `IntraPredMode` (variable que comienza en mayúscula) no es un valor codificado directamente en una secuencia de bits codificada, sino una secuencia de valores decodificados por un decodificador. Cuando se deriva `IntraPredMode [xP][yP]` del bloque actual, `IntraPredMode [xP] [yA]` y otros de un bloque vecino se usan si están disponibles.

Las figs. 26A y 26B son tablas para describir los resultados de la verificación en `intra_chroma_pred_mode`.

- 20 La FIG. 26A indica el resultado de la verificación usando un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L) del patrón 2. La FIG. 26B indica el resultado de la verificación usando un bloque vecino cero (sin usar la condición de bloque superior L ni la condición de bloque izquierdo L) del patrón 3.

El resultado de la verificación en cada una de las Figs. 26A y 26B indican el incremento y la disminución de la tasa BD de acuerdo con los cuatro patrones de prueba como se describe para la primera verificación. Adicionalmente, el significado del valor de evaluación es el mismo que el de la primera verificación.

- 25 El resultado es diferente de los de la primera verificación de `split_coding_unit_flag` y la segunda verificación de `skip_flag`. No hay una diferencia significativa en la tasa BD entre los patrones 1 y 2 o 3 como un patrón de un modelo de contexto para `intra_chroma_pred_mode`.

- 30 De este modo, bajo un entorno mixto con una pluralidad de parámetros de control de tipos de señal, la unidad de control de contexto 142 o 242 determina un valor de contexto sin usar el bloque superior como un bloque vecino, particularmente cuando el tipo de señal del parámetro de control es `intra_chroma_pred_mode`. En otras palabras, la unidad de control de contexto 142 o 242 determina un valor de contexto usando el patrón 2 o 3 cuando el tipo de señal del parámetro de control es `intra_chroma_pred_mode`. En otras palabras, el primer tipo incluye "`split_coding_unit_flag`" o "`skip_flag`", y el segundo tipo o el tercer tipo incluye "`intra_chroma_pred_mode`". Por consiguiente, el aparato de codificación de imagen y el aparato de decodificación de imagen de acuerdo con la realización 3 pueden reducir el uso de memoria mientras suprimen la disminución de la tasa BD.

Cuando el patrón 2 se compara con el patrón 3 para `intra_chroma_pred_mode`, estas tasas de BD no tienen una diferencia significativa. De este modo, se prefiere usar el patrón 3 para `intra_chroma_pred_mode`. Por consiguiente, es posible reducir aún más el uso de memoria y la cantidad de procesamiento.

- 40 En este punto, `intra_chroma_pred_mode` es información de un total de 4 bits e indica si el primer 1 bit indica si el modo de intrapredicción para luminancia se utiliza o no como el modo de intrapredicción para crominancia. En este punto, las unidades de control de contexto 142 y 242 usan el patrón 2 o 3 para el primer 1 bit. Más específicamente, `intra_chroma_pred_mode` indica 0 cuando se usa el mismo modo que el de luminancia para el crominancia. Cuando se usa un modo diferente del de luminancia para el crominancia, `intra_chroma_pred_mode` indica 1, y los 3 bits restantes indican un modo utilizado para crominancia.

- 45 En la intrapredicción, la correlación entre el bloque superior, el bloque izquierdo y se usa el bloque actual. En otras palabras, dado que la información de correlación se usa en la intrapredicción, usar el mismo modo que el de luminancia para crominancia es eficiente. En otras palabras, aunque el modo diferente del de luminancia se puede usar para crominancia para tener diversos modos, un caso en el que se usa el modo diferente del de luminancia para el crominancia y se usa el contexto circundante es raro. En otras palabras, hay muchos casos en los que `intra_chroma_pred_mode` en el que se usa el mismo modo que el de luminancia se establece en 0. Por consiguiente, hay un pequeño mérito en usar el contexto circundante, el uso del contexto del bloque actual permite reducir la cantidad de procesamiento mientras se mantiene la eficiencia de codificación.

Adicionalmente, determinar si el bloque superior está disponible o no en la decodificación es extremadamente difícil. La matriz de `IntraPredMode` que se deriva en un procedimiento de decodificación y se asigna a una secuencia se

describirá con referencia a la FIG. 38.

IntraPredMode de una fila superior (Línea L) que incluye el bloque actual está representado por IntraPredMode [n-ésimo bloque en una dirección horizontal] [una fila superior (Línea L)]. Adicionalmente, IntraPredMode de la fila actual, incluido el bloque actual, está representado por IntraPredMode [k-ésimo bloque en una dirección horizontal] [fila actual]. En este punto, una señal en la que se va a realizar actualmente la decodificación aritmética es intra_chroma_pred_mode [i-ésimo bloque en una dirección horizontal] [fila actual].

En primer lugar, no hay garantía de que el bloque n-ésimo en una fila superior en una dirección horizontal esté asociado con el bloque k-ésimo en la fila actual en la dirección horizontal. Como se describe para la FIG. 38, esto se debe a que el tamaño de un bloque PU varía para cada bloque. De este modo, no hay manera, excepto proporcionar una cierta tabla de correspondencia para administrar estos bloques u obtener todos los IntraPredModes como la unidad mínima como se describe para la FIG. 38.

Adicionalmente, aparte de intra_chroma_pred_mode para decodificar, IntraPredMode de la fila superior no es un tipo de señal capaz de ser obtenido a través del análisis por la unidad de decodificación aritmética, sino un valor (variable de H.264, etc. a partir de un capital) derivado de otro procedimiento de decodificación. De este modo, obtener la disponibilidad de este valor únicamente por la unidad de decodificación aritmética es oneroso.

De este modo, en el modelo de contexto en el que intra_chroma_pred_mode usa bloques vecinos, no usar CondA que es un valor condicional del bloque superior (en particular, un valor de determinación de condición de IntraPredMode [posición horizontal correspondiente] [una fila superior]) es útil en términos del uso de memoria.

(Novena verificación) "cbf_luma, cbf_cr, cbf_cb"

La FIG. 27 es una tabla que indica un procedimiento de decodificación aritmética para cbf_luma, cbf_cr y cbf_cb.

El significado del tipo de señal "cbf_luma" se define de la siguiente manera.

cbf_luma[x0][y0][trafoDepth] igual a 1 especifica que el bloque de transformación luminancia contiene uno o varios niveles de coeficientes de transformación que no son iguales a 0. Los índices de matriz x0, y0 especifica la ubicación (x0, y0) de la muestra de luminancia superior izquierda del bloque de transformación considerado en relación con la muestra de luminancia superior izquierda de la imagen. El índice de matriz trafoDepth especifica el nivel de subdivisión actual de una unidad de codificación en bloques con el fin de transformar la codificación. trafoDepth es igual a 0 para bloques que corresponden a unidades de codificación.

En otras palabras, la posición del bloque de transformación luminancia se define como un valor de una matriz tridimensional que incluye una profundidad jerárquica (trafoDepth) en relación con una capa de una CU, además de los elementos verticales y horizontales.

Adicionalmente, un tipo de señal "cbf_cb" define cbf_luma con respecto a luminancia para crominancia (cb). El significado del tipo de señal "cbf_cb" se define de la siguiente manera.

cbf_cb[x0][y0][trafoDepth] igual a 1 especifica que el bloque de transformación luminancia contiene uno o varios niveles de coeficientes de transformación que no son iguales a 0. Los índices de matriz x0, y0 especifica la ubicación (x0, y0) de la muestra de luminancia superior izquierda del bloque de transformación considerado en relación con la muestra de luminancia superior izquierda de la imagen. El índice de matriz trafoDepth especifica el nivel de subdivisión actual de una unidad de codificación en bloques con el fin de transformar la codificación. trafoDepth es igual a 0 para bloques que corresponden a unidades de codificación. Cuando cbf_cb[x0][y0][trafoDepth] no está presente y PredMode no es igual a MODE_INTRA, se infiere que el valor de cbf_cb[x0][y0][trafoDepth] es igual a 0.

En otras palabras, la posición del bloque de transformación Cb se define como un valor de una matriz tridimensional que incluye una profundidad jerárquica (trafoDepth) en relación con una capa de una CU, además de los elementos verticales y horizontales.

Adicionalmente, un tipo de señal "cbf_cr" define cbf_luma con respecto a luminancia para crominancia (cr). El significado del tipo de señal "cbf_cr" se define de la siguiente manera.

cbf_cr[x0][y0][trafoDepth] igual a 1 especifica que el bloque de transformación luminancia contiene uno o varios niveles de coeficientes de transformación que no son iguales a 0. Los índices de matriz x0, y0 especifica la ubicación (x0, y0) de la muestra de luminancia superior izquierda del bloque de transformación considerado en relación con la muestra de luminancia superior izquierda de la imagen. El índice de matriz trafoDepth especifica el nivel de subdivisión actual de una unidad de codificación en bloques con el fin de transformar la codificación. trafoDepth es igual a 0 para bloques que corresponden a unidades de codificación. Cuando cbf_cr[x0][y0][trafoDepth] no está presente y PredMode no es igual a MODE_INTRA, se infiere que el valor de cbf_cr[x0][y0][trafoDepth] es igual a 0.

En otras palabras, la posición del bloque de transformación Cb se define como un valor de una matriz tridimensional que incluye una profundidad jerárquica (trafoDepth) en relación con una capa de una CU, además de los elementos verticales y horizontales.

Las FIGS. 16A y 16B son tablas para describir los resultados de la verificación en cbf_luma, cbf_cr y cbf_cb.

La FIG. 28A indica el resultado de la verificación usando un bloque vecino (solo un valor de determinación de la condición de bloque izquierdo L) del patrón 2. La FIG. 28B indica el resultado de la verificación usando un bloque vecino cero (sin usar la condición de bloque superior L ni la condición de bloque izquierdo L) del patrón 3.

- 5 El resultado de la verificación en cada una de las Figs. 28A y 28B indican el incremento y la disminución de la tasa BD de acuerdo con los cuatro patrones de prueba como se describe para la primera verificación. Adicionalmente, el significado del valor de evaluación es el mismo que el de la primera verificación.

10 El resultado es diferente de los de la primera verificación de split_coding_unit_flag y la segunda verificación de skip_flag. No hay una diferencia significativa en la tasa BD entre los patrones 1 y 2 o 3 como un patrón de un modelo de contexto para cbf_luma, cbf_cr y cbf_cb.

15 De este modo, bajo un entorno mixto con una pluralidad de parámetros de control de tipos de señal, la unidad de control de contexto 142 o 242 determina un valor de contexto sin usar el bloque superior como un bloque vecino, particularmente cuando el tipo de señal del parámetro de control es cbf_luma, cbf_cr y cbf_cb. En otras palabras, la unidad de control de contexto 142 o 242 determina un valor de contexto usando el patrón 2 o 3 cuando el tipo de señal del parámetro de control es cbf_luma, cbf_cr y cbf_cb. En otras palabras, el primer tipo incluye "split_coding_unit_flag" o "skip_flag", y el segundo tipo o el tercer tipo incluye "cbf_luma, cbf_cr y cbf_cb". Por consiguiente, el aparato de codificación de imagen y el aparato de decodificación de imagen de acuerdo con la realización 3 pueden reducir el uso de memoria mientras suprimen la disminución de la tasa BD.

20 Cuando el patrón 2 se compara con el patrón 3 para cbf_luma, cbf_cr y cbf_cb, estas tasas de BD no tienen una diferencia significativa. De este modo, se prefiere usar el patrón 3 para cbf_luma, cbf_cr y cbf_cb. Por consiguiente, es posible reducir aún más el uso de memoria y la cantidad de procesamiento.

25 Adicionalmente, cbf_luma, cbf_cb y cbf_cr son valores de una matriz tridimensional que tiene una profundidad. De este modo, como se describe para la FIG. 9B de acuerdo con la realización 2, las distancias (lejanía) entre los valores de referencia vecinos en el orden de decodificación (incluido el orden de ejecución recursiva y será el mismo en lo sucesivo) son diferentes de acuerdo con las profundidades jerárquicas respectivas. Por consiguiente, si el valor del parámetro de control está disponible realmente o no, o si la ventaja para reducir el uso de memoria difiere dependiendo de la posición de un bloque en la relación jerárquica.

30 De este modo, las unidades de control de contexto 142 y 242 pueden cambiar un criterio de determinación de un valor condicional de acuerdo con la profundidad jerárquica. Por ejemplo, la unidad de control de contexto 142 y 242 puede usar (seguir) un valor condicional de un bloque en la capa superior como un valor condicional de un bloque en la capa inferior.

Adicionalmente, la unidad de control de contexto 142 y 242 puede cambiar estos criterios en consideración de la relación de posición con otro corte, además de lo anterior o individualmente.

35 Como resultado de las verificaciones anteriores, la siguiente modificación detallada es concebible. Si el efecto de dilución ocurre o no depende del número de entrenamientos a los que se aplica la condición. Generalmente, Y que representa la luminancia en Y, U y V, tal como el formato 4:2:0, es mayor en número de muestras que los otros dos ejes (U y V). De este modo, un límite que debe distinguirse es entre (a) luminancia y (b) un par de cb y cr.

40 Por ejemplo, el patrón 3 puede aplicarse a (a) cbf_luma, y el otro patrón 2 o 1 puede aplicarse a (b) cbf_cb y cbf_cr. En otras palabras, la unidad de control de contexto 142 y 242 puede determinar un contexto usando una condición diferente para cada caso donde el tipo de señal es el "cbf_luma", o el tipo de señal es uno de los "cbf_cb" y el "cbf_cr". Adicionalmente, la unidad de control de contexto 142 y 242 puede aplicar el patrón 1 a (a) cbf_luma, y otro patrón 2 o 3 a (b) cbf_cb y cbf_cr.

Adicionalmente, la unidad de control de contexto 142 y 242 puede determinar un contexto de acuerdo con una profundidad jerárquica para (b) cbf_cb y cbf_cr.

45 El número de entrenamientos para cada una de las condiciones (condiciones M fijas ($M < N$)) de (b) cbf_cb y cbf_cr que tienen el menor número de muestras se puede aproximar al número de entrenamientos de cada una de las condiciones (N condiciones, por ejemplo, cuatro condiciones de 0, 1, 2 y 3) de (a) cbf_luma que tiene el mayor número de muestras, reduciendo en gran medida el número de condiciones (condiciones de contexto) para cbf_cb y cbf_cr a menos que el de (a) cbf_luma.

50 En este punto, la unidad de control de contexto 142 y 242 puede determinar un contexto de acuerdo con una profundidad jerárquica para (a) cbf_luma.

En este punto, cuando el número de entrenamientos es suficiente, se prefiere que las unidades de control de contexto 142 y 242 aumenten el número de condiciones (de contexto) para la precisión. Adicionalmente, se prefiere que las unidades de control de contexto 142 y 242 disminuyan el número de condiciones de contexto cuando el número de

entrenamientos es menor. De este modo, la unidad de control de contexto 142 y 242 puede conmutar entre estas condiciones de acuerdo con la resolución. Adicionalmente, la unidad de control de contexto 142 y 242 puede conmutar entre estas condiciones de acuerdo con el formato (4:2:0) y otros.

Adicionalmente, `cbf_luma`, `cbf_cr` y `cbf_cb` indican la presencia o ausencia de coeficientes de luminancia o crominancia. En otras palabras, `cbf_luma`, `cbf_cr` y `cbf_cb` son jerárquicamente más bajos que `no_residual_data_flag`, lo que indica la presencia o ausencia de datos residuales. En este punto, `no_residual_data_flag` se usa para la TU más grande que se puede seleccionar para el tamaño de una CU en una relación de $CU \geq PU \geq TU$. Más específicamente, `no_residual_data_flag` se usa en la capa superior de una TU. Por otro lado, `cbf_luma`, `cbf_cr` y `cbf_cb` se usan en una capa inferior a la capa de `no_residual_data_flag`. La presencia de `no_residual_data_flag` indica que los bloques posteriores no tienen datos residuales. Adicionalmente, a medida que la profundidad es mayor, los datos residuales probablemente estén presentes. De este modo, la unidad de control de contexto 142 y 242 puede mejorar la eficiencia de codificación usando la información jerárquica para `cbf_luma`, `cbf_cr` y `cbf_cb`. En otras palabras, la unidad de control de contexto 142 y 242 puede determinar un contexto de acuerdo con una profundidad jerárquica de una unidad de datos a la que el parámetro de control (`cbf_luma`, `cbf_cr`, y `cbf_cb`) del bloque actual pertenece. Por otro lado, como `no_residual_data_flag` es una bandera que depende de un modo de predicción, se prefiere que se use un valor fijo que no dependa de una profundidad como valor de contexto.

El sumario del resultado de las verificaciones en todos los tipos de señal se describirá a continuación.

Las FIGS. 29A y 29B son gráficos que indican cada uno un resultado de "4.1 Configuración intra alta eficiencia" (todos los tipos de señal).

Las FIGS. 30A y 30B son gráficos que indican cada uno un resultado de "4.3 Configuración de acceso aleatorio de alta frecuencia" (todos los tipos de señales).

Las FIGS. 31A y 31B son gráficos que indican cada uno un resultado de "4.5 Configuración intra alta eficiencia" (todos los tipos de señal).

Las FIGS. 32A y 32B son gráficos que indican "4.7 Configuración de baja demora y alta eficiencia (solo cortes P)".

La FIG. 33A es una tabla que indica un conjunto de parámetros en el que uno de los patrones 1 a 3 se aplica a cada uno de los parámetros de control. En el ejemplo de la FIG. 33A, el patrón 1 (que usa el bloque superior y el bloque izquierdo) se aplica a `split_coding_unit_flag` y `skip_flag`, y el patrón 3 (que no usa ni el bloque superior ni el bloque izquierdo) se aplica a `merge_flag`, `ref_idx`, `inter_pred_flag`, `mvd_l0`, `mvd_l1`, `mvd_lc`, `no_residual_data_flag`, `intra_chroma_pred_mode`, `cbf_luma`, `cbf_cb` y `cbf_cr`.

La FIG. 33B es una tabla que indica un resultado de verificación cuando el conjunto de parámetros indicado en la FIG. 33A se usa. Como se indica en la FIG. 33B, usando el conjunto de parámetros en la FIG. 33A puede reducir el uso de memoria mientras suprime la disminución de la eficiencia de codificación.

La FIG. 34A es una tabla que indica un ejemplo de otro conjunto de parámetros. En el ejemplo de la FIG. 34A, el patrón 1 (que usa tanto el bloque superior como el bloque izquierdo) se aplica a `split_coding_unit_flag` y `skip_flag`, el patrón 2 (usando solo el bloque izquierdo) se aplica a `intra_chroma_pred_mode`, `cbf_luma`, `cbf_cb` y `cbf_cr`, y el patrón 3 (sin usar el bloque superior ni el bloque izquierdo) se aplica a `merge_flag`, `ref_idx`, `inter_pred_flag`, `mvd_l0`, `mvd_l1`, `mvd_lc` y `no_residual_data_flag`.

La FIG. 34B es una tabla que indica un resultado de verificación cuando el conjunto de parámetros indicado en la FIG. 34A se usa. Como se indica en la FIG. 34B, usando el conjunto de parámetros en la FIG. 34A puede reducir el uso de memoria mientras suprime la disminución de la eficiencia de codificación.

Adicionalmente, la unidad de control de contexto 142 o 242 puede determinar un contexto usando una profundidad como se describe anteriormente.

La FIG. 35 es un diagrama de flujo que indica un procedimiento de codificación aritmética ejecutado por la unidad de codificación de longitud variable 104. Los procedimientos de la FIG. 35 se obtienen agregando las Etapas S111 y S112 a los procedimientos en la FIG. 4.

Cuando el tipo de señal es el cuarto tipo correspondiente al patrón 4 (el cuarto tipo en S104), la unidad de control de contexto 142 determina un valor de contexto usando un valor de un parámetro de control de un bloque vecino (bloque izquierdo) y una profundidad del bloque actual, como el patrón 4 (S111). En otras palabras, la unidad de control de contexto 142 determina un contexto de acuerdo con un parámetro de control del bloque izquierdo y una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control del bloque actual, sin usar un parámetro de control del bloque superior.

Adicionalmente, cuando el tipo de señal es el quinto tipo correspondiente al patrón 5 (quinto tipo en S104), la unidad de control de contexto 142 determina un valor de contexto usando una profundidad del bloque actual como el patrón 5 (S112). En otras palabras, la unidad de control de contexto 142 determina un contexto de acuerdo con una

profundidad jerárquica de una unidad de datos a la que el parámetro de control del bloque actual pertenece, sin usar los parámetros de control del bloque superior y del bloque izquierdo.

La unidad de decodificación de longitud variable 202 realiza los mismos procedimientos. La FIG. 36 es un diagrama de flujo que indica un procedimiento de decodificación aritmética ejecutado por la unidad de decodificación de longitud variable 202. Los procedimientos de la FIG. 36 se obtienen agregando las Etapas S213 y S214 a los procedimientos en la FIG. 7.

Cuando el tipo de señal es el cuarto tipo correspondiente al patrón 4 (el cuarto tipo en S205), la unidad de control de contexto 242 determina un valor de contexto usando un valor de un parámetro de control de un bloque vecino (bloque izquierdo) y una profundidad del bloque actual, como el patrón 4 (S213). En otras palabras, la unidad de control de contexto 242 determina un contexto de acuerdo con un parámetro de control del bloque izquierdo y una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control del bloque actual, sin usar un parámetro de control del bloque superior.

Adicionalmente, cuando el tipo de señal es el quinto tipo correspondiente al patrón 5 (quinto tipo en S205), la unidad de control de contexto 242 determina un valor de contexto usando una profundidad del bloque actual como el patrón 5 (S214). En otras palabras, la unidad de control de contexto 242 determina un contexto de acuerdo con una profundidad jerárquica de datos de una unidad de datos a la que el parámetro de control del bloque actual pertenece, sin usar los parámetros de control del bloque superior y del bloque izquierdo.

En este punto, en la descripción de cada una de las realizaciones, el patrón 2 puede reemplazarse con el patrón 4, y el patrón 3 puede reemplazarse con el patrón 5.

Aunque se describen el aparato de codificación de imagen y el aparato de decodificación de imagen de acuerdo con las realizaciones 1 a 3 de la presente invención, la presente invención no se limita a estas realizaciones.

Por ejemplo, al menos parte del aparato de codificación de imagen, se pueden combinar el aparato de decodificación de imagen y las funciones de las modificaciones de estos aparatos de acuerdo con las realizaciones 1 a 3.

Adicionalmente, todos los valores y los valores lógicos descritos anteriormente son ejemplos para describir específicamente la presente invención, y la presente invención no está limitada por los valores ejemplificados.

Adicionalmente, las divisiones de los bloques funcionales en los diagramas de bloques son ejemplos. De este modo, los bloques funcionales pueden implementarse como un bloque funcional, un bloque funcional puede dividirse en una pluralidad de bloques funcionales, y una parte de las funciones puede conmutarse a otro bloque funcional. Adicionalmente, una pluralidad de bloques funcionales que tiene funciones similares puede ser procesada por un solo hardware o software en paralelo o con división de tiempo.

Las órdenes de las etapas del procedimiento de codificación de imagen realizado por el aparato de codificación de imagen y el procedimiento de decodificación de imagen realizado por el aparato de decodificación de imagen son para describir específicamente la presente invención, y puede ser un orden distinto de las órdenes anteriores. Adicionalmente, parte de las etapas se pueden realizar simultáneamente (en paralelo) con las otras etapas.

(Verificaciones adicionales)

El resultado de las verificaciones sobre las combinaciones de tipos de señal y los procedimientos de determinación de contexto se describirán a continuación. Con las combinaciones, se verifica que el uso de la memoria se puede reducir sin disminuir la tasa BD en mayor medida.

<Combinación 1>

skip_flag Patrón 1 (que se refiere tanto a los bloques superior como izquierdo);
split_coding_unit_flag Patrón 1 (que se refiere tanto a los bloques superior como izquierdo);
merge_flag Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo);
intra_chroma_pred_mode Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo);
inter_pred_flag Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo);
ref_idx Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo);
mvd Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo);
cbf_luma (intra) Patrón 1 (que se refiere tanto a los bloques superior como izquierdo); y
cbf_chroma (intra) Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo).

<Combinación 2>

skip_flag Patrón 1 (que se refiere tanto a los bloques superior como izquierdo);
split_coding_unit_flag Patrón 1 (que se refiere tanto a los bloques superior como izquierdo);
merge_flag Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo);
intra_chroma_pred_mode Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo);
inter_pred_flag Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo);

mvd Patrón 3 (que no se refiere al bloque superior ni al bloque izquierdo);
 cbf_luma(intra) Patrón 5 (que no se refiere al bloque superior ni al bloque izquierdo sino a una profundidad); y
 cbf_chroma (intra) Patrón 5 (que no se refiere al bloque superior ni al bloque izquierdo sino a una profundidad).

(Realización 4)

5 El procesamiento descrito en cada una de las realizaciones puede implementarse simplemente mediante un sistema informático mediante la grabación, sobre un medio de grabación, un programa para implementar la estructura del procedimiento de codificación de imagen en movimiento o el procedimiento de decodificación de imagen en movimiento descrito en la realización. El medio de grabación puede ser cualquier medio de grabación siempre que el programa pueda grabarse en él, tal como un disco magnético, un disco óptico, un disco óptico magnético, una tarjeta IC y una memoria de semiconductores.

De aquí en adelante, se describirán las aplicaciones al procedimiento de codificación de imagen en movimiento o al procedimiento de decodificación de imagen en movimiento descrito en cada una de las realizaciones y se describirá un sistema que usa el mismo.

15 La FIG. 39 ilustra una configuración global de un sistema de provisión de contenido ex100 para implementar servicios de distribución de contenido. El área para proporcionar servicios de comunicación se divide en celdas del tamaño deseado, y las estaciones base ex106 a ex110 que son estaciones inalámbricas fijas se colocan en cada una de las celdas.

El contenido que proporciona el sistema ex100 está conectado a dispositivos, tal como un ordenador ex111, un asistente digital personal (PDA) ex112, una cámara ex113, un teléfono celular ex114 y una máquina de juego ex115, a través de un Internet ex101, un proveedor de servicios de Internet ex102, una red telefónica ex104, así como las estaciones base ex106 a ex110.

25 Sin embargo, la configuración del sistema de provisión de contenido ex100 no se limita a la configuración mostrada en la FIG. 39, y es aceptable una combinación en la que está conectado cualquiera de los elementos. Además, cada uno de los dispositivos puede estar conectado directamente a la red telefónica ex104, en lugar de a través de las estaciones base ex106 a ex110, que son las estaciones inalámbricas fijas. Adicionalmente, los dispositivos pueden interconectarse entre sí mediante una comunicación inalámbrica de corta distancia y otras.

La cámara ex113, tal como una cámara de vídeo digital, es capaz de capturar imágenes en movimiento. Una cámara ex116, tal como una cámara de vídeo digital, es capaz de capturar tanto imágenes fijas como imágenes en movimiento. Adicionalmente, el teléfono celular ex114 puede ser el que cumpla con cualquiera de los estándares, tal como el Sistema Global para Comunicaciones Móviles (GSM), Acceso Múltiple por División de Código (CDMA), Acceso Múltiple por División de Código de Banda Ancha (W-CDMA), Evolución a Largo Plazo (LTE) y Acceso a Paquetes de Alta Velocidad (HSPA). Como alternativa, el teléfono ex114 celular puede ser un Sistema Móvil Personal (PHS).

35 En el sistema de suministro de contenido ex100, un servidor de difusión ex103 está conectado a la cámara ex113 y otros a través de la red telefónica ex104 y la estación base ex 109, que permite la distribución de un show en vivo y otros. Para tal distribución, un contenido (por ejemplo, vídeo de un show de música en vivo) capturado por el usuario usando la cámara ex113 se codifica como se describe anteriormente en cada una de las realizaciones, y el contenido codificado se transmite al servidor de difusión ex103. Por otro lado, el servidor de flujo continuo ex103 lleva a cabo distribución de flujo en los datos de contenido transmitidos a los clientes tras sus solicitudes. Los clientes incluyen el ordenador ex111, el PDA ex112, la cámara ex113, el teléfono celular ex114 y la máquina de juego ex115 que son capaces de decodificar los datos codificados mencionados anteriormente. Cada uno de los dispositivos que han recibido los datos distribuidos decodifica y reproduce los datos codificados.

45 Los datos capturados pueden codificarse por la cámara ex113 o el servidor de flujo continuo ex103 que transmite los datos, o los procedimientos de codificación pueden compartirse entre la cámara ex113 y el servidor de flujo continuo ex103. De manera similar, los datos distribuidos pueden decodificarse por los clientes o el servidor de flujo continuo ex103, o los procedimientos de decodificaciones pueden compartirse entre los clientes y el servidor de flujo continuo ex103. Adicionalmente, los datos de las imágenes fijas y el vídeo capturado no solo por la cámara ex113 sino también por la cámara ex116 pueden transmitirse al servidor de flujo continuo ex103 a través del ordenador ex111. Los procedimientos de codificación pueden ser realizados por la cámara ex116, el ordenador ex111, o el servidor de difusión ex103, o compartido entre ellos.

50 Adicionalmente, generalmente, el ordenador ex111 y un LSI ex500 incluido en cada uno de los dispositivos realizan dichos procedimientos de codificación y decodificación. El LSI ex500 puede estar configurado de un único chip o una pluralidad de chips. El software para codificar y decodificar imágenes en movimiento puede integrarse en algún tipo de medio de grabación (tal como un CD-ROM, un disco flexible, un disco duro) que puede leer el ordenador ex111 y otros, y los procedimientos de codificación y decodificación pueden realizarse usando el software. Adicionalmente, cuando el teléfono celular ex114 está equipado con una cámara, los datos de vídeo obtenidos por la cámara pueden ser transmitidos. Los datos de vídeo son datos codificados por el LSI ex500 incluido en el teléfono celular ex114.

Adicionalmente, el servidor de flujo ex103 puede estar compuesto por servidores y ordenadores, y puede

descentralizar datos y procesar los datos descentralizados, registrar o distribuir datos.

Como se ha descrito anteriormente, los clientes pueden recibir y reproducir los datos codificados en el sistema de provisión de contenido ex100. En otras palabras, los clientes pueden recibir y decodificar la información transmitida por el usuario y reproducir los datos decodificados en tiempo real en el sistema de suministro de contenido ex100, para que el usuario que no tenga ningún derecho y equipo en particular pueda implementar la difusión personal.

La presente invención no se limita al contenido mencionado anteriormente que proporciona el sistema ex100, y al menos el aparato de codificación de imagen en movimiento o el aparato de decodificación de imagen en movimiento descrito en cada una de las Realizaciones se pueden incorporar en un sistema de difusión digital ex200 como se muestra en la FIG. 40. Más específicamente, una estación de difusión ex201 se comunica o transmite, a través de ondas de radio a un satélite de difusión ex202, datos multiplexados obtenidos multiplexando los datos de audio y los datos de vídeo. Los datos de vídeo se codifican de acuerdo con el procedimiento de codificación de imagen en movimiento descrito en cada una de las realizaciones. Al recibir los datos del vídeo, el satélite de difusión ex202 transmite ondas de radio para su difusión. Seguidamente, una antena de uso doméstico ex204 capaz de recibir una difusión por satélite recibe las ondas de radio. Un dispositivo, tal como una televisión (receptor) ex300 y un decodificador (STB) ex217, decodifica los datos multiplexados recibidos y reproduce los datos.

Adicionalmente, un lector/grabador ex218 que (i) lee y decodifica los datos multiplexados grabados en un medio de grabación ex215, tal como un DVD y un BD, o (ii) codifica señales de vídeo en el medio de grabación ex215, y en algunos casos, escribe los datos obtenidos multiplexando una señal de audio en los datos codificados puede incluir el aparato de decodificación de imagen en movimiento o los aparatos de codificación de imagen en movimiento como se muestra en cada una de las realizaciones. En este caso, las señales de vídeo reproducidas se muestran en el monitor ex219, y otro aparato o sistema puede reproducir las señales de vídeo, usando el medio de grabación ex215 en el que se graban los datos multiplexados. Adicionalmente, también es posible implementar el aparato de decodificación de imagen en movimiento en el decodificador ex217 conectado al cable ex203 para una televisión por cable o la antena ex204 para difusión satelital y/o terrestre, para mostrar las señales de vídeo en el monitor ex219 de la televisión ex300. El aparato de decodificación de imagen en movimiento puede implementarse no en el decodificador de salón sino en la televisión ex300.

La FIG. 41 ilustra la televisión (receptor) ex300 que usa el procedimiento de codificación de imagen en movimiento y el procedimiento de decodificación de imagen en movimiento descritos en cada una de las realizaciones. La televisión ex300 incluye: un sintonizador ex301 que obtiene o proporciona datos multiplexados obtenidos al multiplexar los datos de audio y vídeo a través de la antena ex204 o el cable ex203, etc. que recibe una difusión; una unidad de modulación/demodulación ex302 que demodula los datos multiplexados recibidos o modula datos en datos multiplexados a suministrarse al exterior; y una unidad de multiplexación/demultiplexación ex303 que demultiplica los datos multiplexados modulados en datos de vídeo y datos de audio, o multiplexa datos de vídeo y datos de audio codificados por una unidad de procesamiento de señal en datos ex306.

Adicionalmente, la televisión ex300 incluye, además: una unidad de procesamiento de señal ex306 que incluye una unidad de procesamiento de señal de audio ex304 y una unidad de procesamiento de señal de vídeo ex305 que decodifica datos de audio y vídeo y codifica datos de audio y vídeo, respectivamente; un altavoz ex307 que proporciona la señal de audio decodificada; y una unidad de salida ex309 que incluye una unidad de visualización ex308 que muestra la señal de vídeo decodificada, tal como una pantalla. Adicionalmente, la televisión ex300 incluye una unidad de interfaz ex317 que incluye una unidad de entrada de operación ex312 que recibe una entrada de una operación de usuario. Adicionalmente, la televisión ex300 incluye una unidad de control ex310 que controla de manera global cada elemento constituyente de la televisión ex300, y una unidad de circuito de fuente de alimentación ex311 que suministra potencia a cada uno de los elementos. Aparte de la unidad de entrada de operación ex312, la unidad de interfaz ex317 puede incluir: un puente ex313 que está conectado a un dispositivo externo, tal como el lector/grabador ex218; una unidad de ranura ex314 para permitir la conexión del medio de grabación ex216, tal como una tarjeta SD; un controlador ex315 para conectarse a un medio de grabación externo, tal como un disco duro; y un módem ex316 para conectarse a una red de telefonía. En este punto, el medio de grabación ex216 puede grabar eléctricamente información usando un elemento de memoria de semiconductores no volátil/volátil para almacenamiento. Los elementos constituyentes de la televisión ex300 están conectados entre sí a través de un bus síncrono.

En primer lugar, se describirá la configuración en la que la televisión ex300 descodifica datos multiplexados obtenidos desde el exterior a través de la antena ex204 y otros y reproduce los datos descodificados. En la televisión ex300, al recibir una operación del usuario desde un controlador remoto ex220 y otros, la unidad de multiplexación/demultiplexación ex303 demultiplexa los datos multiplexados demodulados por la unidad de modulación/demodulación ex302, bajo control de la unidad de control ex310 que incluye una CPU. Adicionalmente, la unidad de procesamiento de señal de audio ex304 decodifica los datos de audio demultiplexados, y la unidad de procesamiento de señal de vídeo ex305 decodifica los datos de vídeo demultiplexados, usando el procedimiento de decodificación descrito en cada una de las realizaciones, en la televisión ex300. La unidad de salida ex309 proporciona la señal de vídeo decodificada y la señal de audio al exterior. Cuando la unidad de salida ex309 proporciona la señal de vídeo y la señal de audio, las señales pueden almacenarse temporalmente en los búferes ex318 y ex319, y otras para que las señales se reproduzcan en sincronización entre sí. Adicionalmente, la televisión ex300 puede leer los datos multiplexados no a través de una difusión y otros, sino de los medios de grabación ex215 y ex216, tal como un

disco magnético, un disco óptico y una tarjeta SD. A continuación, se describirá una configuración en la que la televisión ex300 codifica una señal de audio y una señal de vídeo, y transmite los datos al exterior o escribe los datos en un medio de grabación. En la televisión ex300, al recibir una operación del usuario desde el controlador remoto ex220 y otros, la unidad de procesamiento de señal de audio ex304 codifica una señal de audio, y la unidad de procesamiento de señal de vídeo ex305 codifica una señal de vídeo, bajo control de la unidad de control ex310 usando el procedimiento de codificación de imagen como se describe en cada una de las realizaciones. La unidad de multiplexación/demultiplexación ex303 multiplexa la señal de vídeo y la señal de audio codificadas, y proporciona la señal resultante al exterior. Cuando la unidad de multiplexación/demultiplexación ex303 multiplexa la señal de vídeo y la señal de audio, las señales pueden almacenarse temporalmente en los búferes ex320 y ex321, y otras para que las señales se reproduzcan en sincronización entre sí. En este punto, los búferes ex318 a ex321 pueden ser varios como se ilustra, o al menos un búfer puede compartirse en la televisión ex300. Adicionalmente, los datos pueden almacenarse en un búfer distinto de los búferes ex318 a ex321 para evitar el desbordamiento y el subflujo del sistema entre la unidad de modulación/demodulación ex302 y la unidad de multiplexación/demultiplexación ex303, por ejemplo.

Adicionalmente, la televisión ex300 puede incluir una configuración para recibir una entrada de AV desde un micrófono o una cámara distinta de la configuración para obtener datos de audio y de vídeo desde una difusión o de un medio de grabación, y puede codificar los datos obtenidos. Aunque la televisión ex300 puede codificar, multiplexar y proporcionar datos externos en la descripción, puede que no sea capaz de realizar todos los procedimientos, sino que solo pueda recibir uno, decodificar y proporcionar datos externos.

Adicionalmente, cuando el lector/grabador ex218 lee o escribe los datos multiplexados desde o en un medio de grabación, una de las televisiones ex300 y el lector/grabador ex218 pueden decodificar o codificar los datos multiplexados, y la televisión ex300 y el lector/grabador ex218 pueden compartir la decodificación o codificación.

A modo de ejemplo, La FIG. 42 ilustra una configuración de una unidad de reproducción/grabación de información ex400 cuando los datos se leen o escriben desde o en un disco óptico. La unidad de reproducción/grabación de información ex400 incluye elementos constituyentes ex401 a ex407 que se describirán más adelante. El cabezal óptico ex401 irradia un punto láser en una superficie de grabación del medio de grabación ex215 que es un disco óptico para escribir información, y detecta luz reflejada desde la superficie de grabación del medio de grabación ex215 para leer la información. La unidad de grabación de modulación ex402 acciona eléctricamente un láser de semiconductores incluido en el cabezal óptico ex401, y modula la luz de láser de acuerdo con datos grabados. La unidad de desmodulación de reproducción ex403 amplifica una señal de reproducción obtenida detectando eléctricamente la luz reflejada desde la superficie de grabación usando un fotodetector incluido en el cabezal óptico ex401, y demodula la señal de reproducción separando un componente de señal grabado en el medio de grabación ex215 para reproducir la información necesaria. El búfer ex404 mantiene temporalmente la información a grabarse en el medio de grabación ex215 y la información reproducida desde el medio de grabación ex215. Un motor de disco ex405 rota el medio de grabación ex215. Una unidad de servocontrol ex406 mueve el cabezal óptico ex401 a una pista de información predeterminada mientras controla el mecanismo de rotación del motor de disco ex405 para seguir el punto láser. La unidad de control de sistema ex407 controla la totalidad de la unidad de reproducción/grabación de información ex400. La unidad de control del sistema ex407 puede implementar los procedimientos de lectura y escritura usando información diversa almacenada en el búfer ex404 y generando y agregando nueva información según sea necesario, y mediante la unidad de grabación de modulación ex402, la unidad de demodulación de reproducción ex403 y la unidad de servocontrol ex406 que registran y reproducen información a través del cabezal óptico ex401 mientras se opera de manera coordinada. La unidad de control del sistema ex407 incluye, por ejemplo, un microprocesador, y ejecuta el procesamiento al hacer que un ordenador ejecute un programa de lectura y escritura.

Aunque el cabezal óptico ex401 irradia un punto láser en la descripción, puede realizar grabaciones de alta densidad usando luz de campo cercano.

La FIG. 43 ilustra esquemáticamente el medio de grabación ex215 que es el disco óptico. Sobre la superficie de grabación del medio de grabación ex215, las ranuras de guía se forman en espiral, y una información rastrea registros ex230, por adelantado, información de dirección que indica una posición absoluta en el disco de acuerdo con el cambio en la forma de las ranuras de guía. La información de dirección incluye información para determinar posiciones de bloques de grabación ex231 que son una unidad para grabar datos. Un aparato que graba y reproduce datos reproduce la pista de información ex230 y lee la información de dirección para determinar las posiciones de los bloques de grabación. Adicionalmente, el medio de grabación ex215 incluye un área de grabación de datos ex233, un área de circunferencia interna ex232 y un área de circunferencia externa ex234. El área de grabación de datos ex233 es un área para su uso al grabar los datos de usuario. El área de circunferencia interna ex232 y el área de circunferencia externa ex234 que están dentro y fuera del área de grabación de datos ex233, respectivamente, son para uso específico, excepto para grabar los datos del usuario. La unidad de reproducción/grabación de información 400 lee y escribe datos de audio codificados, datos de vídeo codificados, o datos multiplexados obtenidos al multiplexar los datos de audio codificados y los datos de vídeo codificados, desde y sobre el área de grabación de datos ex233 del medio de grabación ex215.

Aunque un disco óptico que tiene una capa, tal como un DVD y un BD se describe como un ejemplo en la descripción, el disco óptico no está limitado a tal, y puede ser un disco óptico que tenga una estructura multicapa y que pueda grabarse en una parte que no sea la superficie. Adicionalmente, el disco óptico puede tener una estructura para

grabación/reproducción multidimensional, tal como la grabación de información usando luz de colores con diferentes longitudes de onda en la misma porción del disco óptico y la grabación de información que tiene diferentes capas desde varios ángulos.

Adicionalmente, el coche ex210 que tiene la antena ex205 puede recibir datos del satélite ex202 y otros, y reproducir vídeo en el dispositivo de visualización, tal como el sistema de navegación del coche ex211 configurado en el coche ex210, en un sistema de difusión digital ex200. En este punto, una configuración del sistema de navegación del coche ex211 será la que, por ejemplo, incluya una unidad receptora de GPS en la configuración ilustrada en la FIG. 41. Lo mismo será cierto para la configuración del ordenador ex111, el teléfono celular ex114 y otros.

La FIG. 44A ilustra el teléfono celular ex114 que usa el procedimiento de codificación de imagen en movimiento y el procedimiento de decodificación de imagen en movimiento descritos en cada una de las realizaciones. El teléfono celular ex114 incluye: una antena ex350 para transmitir y recibir ondas de radio a través de la estación base ex110; una unidad de cámara ex365 capaz de capturar imágenes en movimiento y fijas; y una unidad de visualización ex358 tal como un visualizador de cristal líquido para visualizar datos tales como vídeo descodificado capturado por la unidad de cámara ex365 o recibido por la antena ex350. El teléfono celular ex114 incluye, además: una unidad de cuerpo principal que incluye un conjunto de teclas de operación ex366; una unidad de salida de audio ex357 tal como un altavoz para la salida de audio; una unidad de entrada de audio ex356 tal como un micrófono para la entrada de audio; una unidad de memoria ex367 para almacenar vídeo capturado o imágenes fijas, audio grabado, datos codificados o decodificados del vídeo recibido, las imágenes fijas, correos electrónicos u otros; y una unidad de ranura ex364 que es una unidad de interfaz para un medio de grabación que almacena datos de la misma forma que la unidad de memoria ex367.

A continuación, un ejemplo de una configuración del teléfono celular ex114 se describirá con referencia a la FIG. 44B. En el teléfono celular ex114, una unidad de control principal ex360 diseñada para controlar en general cada unidad del cuerpo principal, incluida la unidad de visualización ex358, así como las teclas de operación ex366 están conectadas entre sí, a través de un bus síncrono ex370, a una unidad de circuito de fuente de potencia ex361, una unidad de control de entrada de operación ex362, una unidad de procesamiento de señal de vídeo ex355, una unidad de interfaz de cámara ex363, una unidad de control de pantalla de cristal líquido (LCD) ex359, una unidad de modulación/demodulación ex352, una unidad de multiplexación/demultiplexación ex353, una unidad de procesamiento de señal de audio ex354, la unidad de ranura ex364 y la unidad de memoria ex367.

Cuando la operación de un usuario enciende una tecla de potencia de llamada o una tecla de encendido, la unidad de circuito de suministro de potencia ex361 suministra a las unidades respectivas potencia de un paquete de baterías para activar el teléfono celular ex114.

En el teléfono celular ex114, la unidad de procesamiento de señales de audio ex354 convierte las señales de audio recogidas por la unidad de entrada de audio ex356 en modo de conversación por voz en señales de audio digital bajo el control de la unidad de control principal ex360 que incluye una CPU, ROM y RAM. Seguidamente, la unidad de modulación/demodulación ex352 realiza un procesamiento de espectro extendido en las señales de audio digital, y la unidad de transmisión y recepción ex351 realiza la conversión de digital a analógico y la conversión de frecuencia en los datos, para transmitir los datos resultantes a través de la antena ex350. También, en el teléfono celular ex114, la unidad de transmisión y de recepción ex351 amplifica los datos recibidos por la antena ex350 en modo de conversación por voz y realiza la conversión en frecuencia y la conversión de digital a analógico sobre los datos.

Seguidamente, la unidad de modulación/demodulación ex352 realiza un procesamiento inverso de espectro ensanchado en los datos, y la unidad de procesamiento de señal de audio ex354 lo convierte en señales de audio analógicas, para generarlos a través de la unidad de salida de audio ex357. Adicionalmente, cuando se transmite un correo electrónico en modo de comunicación de datos, los datos de texto del correo electrónico ingresado al operar las teclas de operación ex366 y otros del cuerpo principal se envían a la unidad de control principal ex360 a través de la unidad de control de entrada de operación ex362. La unidad de control principal ex360 da lugar a que la unidad de modulación/demodulación ex352 realice un procesamiento de espectro ensanchado sobre los datos de texto, y la unidad de transmisión y de recepción ex351 realiza la conversión de analógico a digital y la conversión en frecuencia sobre los datos resultantes para transmitir los datos a la estación base ex110 por medio de la antena ex350. Cuando se recibe un correo electrónico, un procesamiento que es aproximadamente inverso al procesamiento para transmitir un correo electrónico se realiza sobre los datos recibidos, y los datos resultantes se proporcionan a la unidad de visualización ex358.

Cuando el vídeo, las imágenes fijas, o vídeo y audio en modo de comunicación de datos se transmite o se transmiten, la unidad de procesamiento de señal de vídeo ex355 comprime y codifica las señales de vídeo suministradas desde la unidad de cámara ex365 usando el procedimiento de codificación de imagen en movimiento que se muestra en cada una de las realizaciones, y transmite los datos de vídeo codificados a la unidad de multiplexación/demultiplexación ex353. Por el contrario, durante cuando la unidad de cámara ex365 captura vídeo, las imágenes fijas, y otras, la unidad de procesamiento de señal de audio ex354 codifica las señales de audio recogidas por la unidad de entrada de audio ex356, y transmite los datos de audio codificados a la unidad de multiplexación/demultiplexación ex353.

La unidad de multiplexación/demultiplexación ex353 multiplexa los datos de vídeo codificados suministrados desde la unidad de procesamiento de señal de vídeo ex355 y los datos de audio codificados suministrados desde la unidad de procesamiento de señal de audio ex354, usando un procedimiento predeterminado. Seguidamente, la unidad de modulación/desmodulación ex352 realiza un procesamiento de espectro ensanchado sobre las señales de audio digital, y la unidad de transmisión y de recepción ex351 realiza una conversión de analógico a digital y una conversión en frecuencia sobre los datos, con el fin de transmitir los datos resultantes por medio de la antena ex350.

Al recibir datos de un archivo de vídeo que está vinculado a una página web y a otros en modo de comunicación de datos o al recibir un correo electrónico con vídeo y/o audio adjunto, para decodificar los datos multiplexados recibidos a través de la antena ex350, la unidad de multiplexación/demultiplexación ex353 demultiplexa los datos multiplexados en un flujo de bits de datos de vídeo y un flujo de bits de datos de audio, y suministra a la unidad de procesamiento de señal de vídeo ex355 los datos de vídeo codificados y la unidad de procesamiento de señal de audio ex354 con los datos de audio codificados, a través del bus síncrono ex370. La unidad de procesamiento de señal de vídeo ex355 decodifica la señal de vídeo usando un procedimiento de decodificación de imagen en movimiento correspondiente al procedimiento de codificación de imagen en movimiento que se muestra en cada una de las realizaciones, y luego la unidad de visualización ex358 muestra, por ejemplo, el vídeo y las imágenes fijas incluidas en el archivo de vídeo vinculado a la página web a través de la unidad de control LCD ex359. Adicionalmente, la unidad de procesamiento de señales de audio ex354 descodifica la señal de audio, y la unidad de salida de audio ex357 proporciona el audio.

Adicionalmente, de manera similar a la televisión ex300, un terminal tal como el teléfono celular ex114 probablemente tenga 3 tipos de configuraciones de implementación que incluyen no solo (i) un terminal de transmisión y recepción que incluye un aparato de codificación y un aparato de decodificación, pero también (ii) un terminal transmisor que incluye solo un aparato de codificación y (iii) un terminal receptor que incluye solo un aparato de decodificación. Aunque el sistema de difusión digital ex200 recibe y transmite los datos multiplexados obtenidos al multiplexar datos de audio en datos de vídeo en la descripción, los datos multiplexados pueden ser datos obtenidos al multiplexar no datos de audio sino datos de caracteres relacionados con vídeo en datos de vídeo, y pueden no ser datos multiplexados sino datos de vídeo en sí.

En consecuencia, el procedimiento de codificación de imagen en movimiento y el procedimiento de decodificación de instantáneas en movimiento en cada una de las realizaciones se pueden usar en cualquiera de los dispositivos y sistemas descritos. De este modo, se pueden obtener las ventajas descritas en cada una de las realizaciones.

Adicionalmente, la presente invención no se limita a las realizaciones, y son posibles diversas modificaciones y revisiones sin apartarse del ámbito de la presente invención.

(Realización 5)

Los datos de vídeo se pueden generar conmutando, según sea necesario, entre (i) el procedimiento de codificación de imagen en movimiento o el aparato de codificación de imagen en movimiento que se muestra en cada una de las realizaciones y (ii) un procedimiento de codificación de imagen en movimiento o un aparato de codificación de imagen en movimiento de conformidad con un estándar diferente, tal como MPEG-2, MPEG4-AVC y VC-1.

En este punto, cuando se genera una pluralidad de datos de vídeo que se ajusta a los diferentes estándares y luego se decodifica, los procedimientos de decodificación deben seleccionarse para cumplir con los diferentes estándares. Sin embargo, ya que a qué estándar no se puede detectar cada una de la pluralidad de datos de vídeo a decodificar, existe el problema de que no se puede seleccionar un procedimiento de decodificación apropiado.

Para resolver el problema, los datos multiplexados obtenidos multiplexando datos de audio y otros en datos de vídeo tienen una estructura que incluye información de identificación que indica con qué estándar cumplen los datos de vídeo. Se describirá en lo sucesivo la estructura específica de los datos multiplexados que incluyen los datos de vídeo generados en el procedimiento de codificación de imagen en movimiento y por el aparato de codificación de imagen en movimiento mostrados en cada una de las realizaciones. Los datos multiplexados son un flujo digital en el formato de Flujo de Transporte de MPEG-2.

La FIG. 45 ilustra una estructura de datos multiplexados. Como se ilustra en la FIG. 45, los datos multiplexados se pueden obtener multiplexando al menos uno de un flujo de vídeo, un flujo de audio, un flujo de gráficos de presentación (PG) y un flujo de gráficos interactivos. El flujo de vídeo representa el vídeo primario y el vídeo secundario de una película, el flujo de audio (IG) representa una parte de audio primaria y una parte de audio secundaria que se mezclará con la parte de audio primaria, y el flujo de gráficos de presentación representa los subtítulos de una película. En este punto, el vídeo primario es vídeo normal a visualizarse en una pantalla, y el vídeo secundario es vídeo a visualizarse en una ventana más pequeña en el vídeo principal. Adicionalmente, el flujo de gráficos interactivo representa una pantalla interactiva a generarse disponiendo los componentes de la GUI en una pantalla. El flujo de vídeo se codifica en el procedimiento de codificación de imagen en movimiento o por el aparato de codificación de imagen en movimiento que se muestra en cada una de las realizaciones, o en un procedimiento de codificación de imagen en movimiento o por un aparato de codificación de imagen en movimiento de conformidad con un estándar convencional, tal como MPEG-2, MPEG4-AVC y VC-1. El flujo de audio está codificado de acuerdo con un estándar, tal como Dolby-AC-3, Dolby Digital Plus, MLP, DTS, DTS-HD y PCM lineal.

Cada flujo incluido en los datos multiplexados se identifica por PID. Por ejemplo, 0x1011 se asigna al flujo de vídeo a usar para el vídeo de una película, 0x1100 a 0x111F se asignan a los flujos de audio, 0x1200 a 0x121F se asignan a los flujos de gráficos de presentación, 0x1400 a 0x141F se asignan a los flujos de gráficos interactivos, 0x1B00 a 0x1B1F se asignan a los flujos de vídeo a usar para el vídeo secundario de la película, y 0x1A00 a 0x1A1F se asignan a los flujos de audio para que el vídeo secundario se mezcle con el audio primario.

La FIG. 46 ilustra esquemáticamente cómo se multiplexan datos. En primer lugar, un flujo ex235 de vídeo compuesto por tramas de vídeo y un flujo ex238 de audio compuesto por tramas de audio se transforman en un flujo ex236 de paquetes de PES y un flujo ex239 de paquetes de PES, y adicionalmente en paquetes ex237 de TS y paquetes ex240 de TS, respectivamente. De manera similar, los datos de un flujo ex241 de gráficos de presentación y los datos de un flujo ex244 de gráficos interactivo se transforman en un flujo ex242 de paquetes de PES y un flujo ex245 de paquetes de PES, y adicionalmente en paquetes ex243 de TS y paquetes ex246 de TS, respectivamente. Estos paquetes de TS se multiplexan en un flujo para obtener datos multiplexados ex247.

La FIG. 47 ilustra cómo se almacena un flujo de vídeo en un flujo de paquetes de PES en más detalle. La primera barra en la FIG. 47 muestra un flujo de cuadro de vídeo en un flujo de vídeo. La segunda barra muestra el flujo de paquetes de PES. Como lo indican las flechas indicadas como yy1, yy2, yy3 e yy4 en la FIG. 47, el flujo de vídeo se divide en imágenes como imágenes I, imágenes B e imágenes P, cada una de las cuales es una unidad de presentación de vídeo, y las imágenes se almacenan en una carga útil de cada uno de los paquetes PES. Cada uno de los paquetes de PES tiene un encabezamiento de PES, y el encabezamiento de PES almacena una Indicación de Tiempo de Presentación (PTS) que indica un tiempo de visualización de la imagen, y una Indicación de Tiempo de decodificación (DTS) que indica un tiempo de decodificación de la imagen.

La FIG. 48 ilustra un formato de paquetes de TS a escribir finalmente en los datos multiplexados. Cada uno de los paquetes TS es un paquete de longitud fija de 188 bytes que incluye un encabezado TS de 4 bytes con información, tal como un PID para identificar un flujo y una carga útil TS de 184 bytes para almacenar datos. Los paquetes de PES se dividen y se almacenan en las cabidas útiles de TS, respectivamente. Cuando se usa un BD ROM, cada uno de los paquetes TS recibe un TP_Extra_Header de 4 bytes, resultando así en paquetes fuente de 192 bytes. Los paquetes fuente se escriben en los datos multiplexados. El TP_Encabezamiento_Adicional almacena información tal como una Indicación_Tiempo_Llegada (ATS). La ATS muestra un tiempo de inicio de transferencia en el que se ha de transferir cada uno de los paquetes de TS a un filtro PID. Los números que se incrementan desde el cabezal de los datos multiplexados se denominan números de paquete fuente (SPN) como se muestra en la parte inferior de la FIG. 48.

Cada uno de los paquetes TS incluidos en los datos multiplexados incluye no solo flujos de audio, vídeo, subtítulos y otros, sino también una Tabla de Asociación de Programa (PAT), una Tabla de Mapa de Programa (PMT), y una Referencia de Reloj de Programa (PCR). La PAT muestra qué indica un PID en una PMT usada en los datos multiplexados, y una PID de la misma PAT se registra como cero. El PMT almacena los PID de los flujos de vídeo, audio, subtítulos y otros incluidos en los datos multiplexados, y la información de atributos de los flujos correspondientes a los PID. La PMT también tiene diversos descriptores relacionados con los datos multiplexados. Los descriptores tienen información tal como información de control de copia que muestra si se permite o no el copiado de los datos multiplexados. El PCR almacena la información de tiempo STC correspondiente a un ATS que muestra cuándo se transfiere el paquete de PCR a un decodificador, para lograr la sincronización entre un reloj de tiempo de llegada (ATC) que es un eje de tiempo de ATS y un reloj de tiempo del sistema (STC) que es un eje de tiempo de PTS y DTS.

La FIG. 49 ilustra la estructura de datos de la PMT en detalle. Un encabezamiento de PMT está dispuesto en la parte superior de la PMT. El encabezamiento de la PMT describe la longitud de datos incluidos en la PMT y otros. Una pluralidad de descriptores relacionados con los datos multiplexados está dispuesta después del encabezamiento de PMT. La información tal como la información de control de copia se describe en los descriptores. Después de los descriptores, está dispuesta una pluralidad de piezas de información de flujo relacionados con los flujos incluidos en los datos multiplexados. Cada pieza de información de flujo incluye descriptores de flujo, cada uno de los cuales describe información, tal como un tipo de flujo para identificar un códec de compresión de un flujo, un PID de flujo y la información de atributo de flujo (tal como una tasa de cuadro o una relación de aspecto). Los descriptores de flujo son iguales en número al número de flujos en los datos multiplexados.

Cuando los datos multiplexados se graban en un medio de grabación y otros, se graba junto con archivos de información de datos multiplexados.

Cada uno de los ficheros de información de datos multiplexados es información de gestión de los datos multiplexados como se muestra en la FIG. 50. Los archivos de información de datos multiplexados están en correspondencia uno a uno con los datos multiplexados, y cada uno de los archivos incluye información de datos multiplexados, información de atributo de flujo, y un mapa de entrada.

Como se ilustra en la FIG. 50, la información de datos multiplexados incluye una tasa de sistema, un tiempo de inicio de reproducción y un tiempo de finalización de reproducción. La tasa de sistema indica la tasa de transferencia máxima a la que un decodificador objetivo de sistema que se va a describir más adelante transfiere los datos multiplexados a un filtro de PID. Los intervalos de las ATS incluidas en los datos multiplexados se establecen para que no sean

superiores a una tasa de sistema. El tiempo de inicio de reproducción indica una PTS en un cuadro de vídeo en el cabezal de los datos multiplexados. Un intervalo de un cuadro se añade a una PTS en un cuadro de vídeo al final de los datos multiplexados, y la PTS se establece al tiempo de fin de reproducción.

Como se muestra en la FIG. 51, una pieza de la información del atributo se registra en la información del atributo del flujo, para cada PID de cada flujo incluido en los datos multiplexados. Cada pieza de información de atributo tiene información diferente dependiendo de si la transmisión correspondiente es un flujo de vídeo, un flujo de audio, un flujo de gráficos de presentación o un flujo de gráficos interactivos. Cada pieza de la información del atributo de flujo de vídeo contiene información que incluye qué tipo de códec de compresión se utiliza para comprimir la transmisión de vídeo y la resolución, relación de aspecto y tasa de cuadro de las piezas de datos de imagen que se incluyen en el flujo de vídeo. Cada pieza de la información del atributo de flujo de audio contiene información que incluye qué tipo de códec de compresión se utiliza para comprimir la transmisión de audio, cuántos canales están incluidos en el flujo de audio, qué idioma admite el flujo de audio y qué tan alta es la frecuencia de muestreo. La información de atributo de flujo de vídeo y la información de atributo de flujo de audio se usan para inicialización de un decodificador antes de que el reproductor reproduzca la información.

En la Realización 5, los datos multiplexados a usar son de un tipo de flujo incluido en la PMT. Adicionalmente, cuando los datos multiplexados se graban en un medio de grabación, se usa la información de atributo de flujo de vídeo incluida en la información de datos multiplexados. Más específicamente, el procedimiento de codificación de imagen en movimiento o el aparato de codificación de imagen en movimiento descritos en cada una de las realizaciones incluye una etapa de o una unidad para asignar información única que indica datos de vídeo generados por el procedimiento de codificación de imagen en movimiento o el aparato de codificación de imagen en movimiento en cada una de las realizaciones, al tipo de flujo incluido en el PMT o la información del atributo de flujo de vídeo. Con la estructura, los datos de vídeo generados por el procedimiento de codificación de imagen en movimiento o el aparato de codificación de imagen en movimiento descritos en cada una de las realizaciones pueden distinguirse de los datos de vídeo que se ajustan a otro estándar.

Adicionalmente, la FIG. 52 ilustra las etapas del procedimiento de decodificación de imagen en movimiento de acuerdo con la realización 5. En la Etapa exS100, el tipo de flujo incluido en el PMT o la información del atributo de transmisión de vídeo se obtiene desde los datos multiplexados. A continuación, en la Etapa exS101, se determina si el tipo de flujo o la información de atributo de flujo de vídeo indica o no que los datos multiplexados se generan por el procedimiento de codificación de imagen en movimiento o el aparato de codificación de imagen en movimiento en cada una de las realizaciones. Cuando se determina si el tipo de flujo o la información de atributo de flujo de vídeo indica o no que los datos multiplexados se generan por el procedimiento de codificación de imagen en movimiento o el aparato de codificación de imagen en movimiento en cada una de las realizaciones, en la Etapa exS102, el tipo de flujo o la información del atributo del flujo de vídeo se decodifica mediante el procedimiento de decodificación de imagen en movimiento en cada una de las realizaciones. Adicionalmente, cuando el tipo de flujo o la información del atributo de flujo de vídeo indica conformidad con los estándares convencionales, tal como MPEG-2, MPEG4-AVC y VC-1, en la Etapa exS103, el tipo de flujo o la información del atributo del flujo de vídeo se decodifica mediante un procedimiento de decodificación de imagen en movimiento de conformidad con los estándares convencionales.

En consecuencia, asignar un nuevo valor único al tipo de flujo o la información de atributo de flujo de vídeo posibilita la determinación de si el procedimiento de decodificación de imagen en movimiento o el aparato de decodificación de imagen en movimiento que se describen en cada una de las realizaciones puede realizar o no la decodificación. Incluso con una entrada de datos multiplexados que se ajusta a un estándar diferente, se puede seleccionar un procedimiento o aparato de decodificación apropiado. De este modo, se hace posible decodificar información sin error alguno. Adicionalmente, el procedimiento o aparato de codificación de imagen en movimiento, o el procedimiento o aparato de decodificación de imagen en movimiento en la presente realización 5 se puede usar en los dispositivos y sistemas anteriormente descritos.

(Realización 6)

Cada uno de los procedimientos de codificación de imagen en movimiento, el aparato de codificación de imagen en movimiento, el procedimiento de decodificación de imagen en movimiento, y el aparato de decodificación de imagen en movimiento en cada una de las realizaciones se consigue típicamente en forma de un circuito integrado o un circuito integrado a gran escala (LSI). Como ejemplo del LSI, la FIG. 53 ilustra una configuración del LSI ex500 que se hace en un chip. El LSI ex500 incluye elementos ex501, ex502, ex503, ex504, ex505, ex506, ex507, ex508 y ex509 se describirán a continuación, y los elementos están conectados entre sí a través de un bus ex510. La unidad de circuito de fuente de alimentación ex505 se activa suministrando a cada uno de los elementos con potencia cuando se activa la unidad de circuito de fuente de alimentación ex505.

Por ejemplo, cuando se realiza la codificación, el LSI ex500 recibe una señal AV de un micrófono ex117, una cámara ex113 y otras a través de un AV IO ex509 bajo el control de una unidad de control ex501 que incluye una CPU ex502, un controlador de memoria ex503, un controlador de flujo ex504 y una unidad de control de frecuencia de accionamiento ex512. La señal de AV recibida se almacena temporalmente en una memoria externa ex511, tal como una SDRAM, tal como una SDRAM. Bajo control de la unidad de control ex501, los datos almacenados se segmentan en porciones de datos de acuerdo con la cantidad de procesamiento y la velocidad a transmitir a una unidad de

procesamiento de señal ex507. Seguidamente, a continuación, la unidad de procesamiento de señal ex507 codifica una señal de audio y/o una señal de vídeo. En este punto, la codificación de la señal de vídeo es la codificación descrita en cada una de las realizaciones. Adicionalmente, la unidad de procesamiento de señal ex507 multiplexa en ocasiones los datos de audio codificados y los datos de vídeo codificados, y una ES de flujo ex506 proporciona los datos multiplexados al exterior. Los datos multiplexados proporcionados se transmiten a la estación base ex107, o se escriben en el medio de grabación ex215. Cuando los conjuntos de datos se multiplexan, los datos deberían almacenarse temporalmente en el búfer ex508 de modo que los conjuntos de datos se sincronizan entre sí.

Aunque la memoria ex511 es un elemento externo al LSI ex500, se puede incluir en el LSI ex500. El búfer ex508 no está limitado a un búfer, pero puede estar compuesto de búferes. Adicionalmente, el LSI ex500 puede estar fabricado en un chip o una pluralidad de chips.

Adicionalmente, aunque la unidad de control ex501 incluye la CPU ex502, el controlador de memoria ex503, el controlador de flujo ex504, la unidad de control de frecuencia de accionamiento ex512, la configuración de la unidad de control ex501 no se limita a tal. Por ejemplo, la unidad de procesamiento de señal ex507 puede incluir adicionalmente una CPU. La inclusión de otra CPU en la unidad de procesamiento de señal ex507 puede mejorar la velocidad de procesamiento. Adicionalmente, como otro ejemplo, la CPU ex502 puede servir o ser parte de la unidad de procesamiento de señal ex507, y, por ejemplo, puede incluir una unidad de procesamiento de señal de audio. En tal caso, la unidad de control ex501 incluye la unidad de procesamiento de señal ex507 o la CPU ex502 que incluye una parte de la unidad de procesamiento de señal ex507.

El nombre usado aquí es LSI, pero también puede llamarse IC, sistema LSI, súper LSI o ultra LSI dependiendo del grado de integración.

Más aún, las maneras para conseguir la integración no se limitan al LSI, y un circuito especial o un procesador de propósito general y así sucesivamente pueden conseguir también la integración. El Campo de Matriz de Puertas Programables (FPGA) que puede programarse después de la fabricación de LSI o un procesador reconfigurable que permite la reconfiguración de la conexión o configuración de un LSI puede usarse para el mismo fin.

En el futuro, con avance en la tecnología de semiconductores, una tecnología completamente nueva puede reemplazar a LSI. Los bloques funcionales pueden integrarse usando tal tecnología. La posibilidad es que la presente invención se aplique a biotecnología.

(Realización 7)

Cuando los datos de vídeo se decodifican por el procedimiento de codificación de imagen en movimiento o por el aparato de codificación de imagen en movimiento descrito en cada una de las realizaciones, en comparación con cuando los datos de vídeo se ajustan a un estándar convencional, tal como MPEG-2, MPEG4-AVC y VC-1, la cantidad de computación probablemente aumenta. De este modo, el LSI ex500 necesita establecer una frecuencia de accionamiento más alta que la de la CPU ex502 a usarse cuando se decodifican datos de vídeo de conformidad con el estándar convencional. Sin embargo, cuando la frecuencia de accionamiento es más alta, hay un problema de que aumenta el consumo de potencia.

Para resolver el problema, el aparato de decodificación de imagen en movimiento, tal como la televisión ex300 y el LSI ex500 está configurada para determinar a qué estándar se ajustan los datos de vídeo y conmutar entre las frecuencias de accionamiento de acuerdo con el estándar determinado. La FIG. 54 ilustra una configuración ex800 en la realización 7. Una unidad de conmutación de frecuencia de accionamiento ex803 establece una frecuencia de accionamiento a una frecuencia de accionamiento superior cuando los datos de vídeo se generan por el procedimiento de codificación de imagen o el aparato de codificación de imagen descrito en cada realización. Seguidamente, la unidad de conmutación de frecuencia de accionamiento ex803 ordena a la unidad de procesamiento de decodificación ex801 que ejecute el procedimiento de decodificación de imagen en movimiento descrito en cada una de las realizaciones para decodificar los datos de vídeo. Cuando los datos de vídeo se ajustan al estándar convencional, la unidad de conmutación de frecuencia de accionamiento ex803 establece una frecuencia de accionamiento a una frecuencia de accionamiento inferior que la de los datos de vídeo generados por el procedimiento de codificación de imagen o el aparato de codificación de imagen descrito en cada realización. Seguidamente, la unidad de conmutación de frecuencia de accionamiento ex803 ordena a la unidad de procesamiento de decodificación ex802 que se ajusta al estándar convencional que decodifique los datos de vídeo.

Más específicamente, la unidad de conmutación de frecuencia de accionamiento ex803 incluye la CPU ex502 y la unidad de control de frecuencia de accionamiento ex512 en la FIG. 53. En este punto, cada una de la unidad de procesamiento de decodificación ex801 que ejecuta el procedimiento de decodificación de imagen en movimiento descrito en cada una de las realizaciones y la unidad de procesamiento de decodificación ex802 que se ajusta al estándar convencional se corresponden con la unidad de procesamiento de señal ex507 en la FIG. 53. La CPU ex502 determina a qué estándar se ajusta los datos de vídeo. Seguidamente, la unidad de control de frecuencia de accionamiento ex512 determina una frecuencia de accionamiento basándose en una señal desde la CPU ex502. Adicionalmente, la unidad de procesamiento de señal ex507 decodifica los datos de vídeo basándose en la señal desde la CPU ex502. Por ejemplo, la información de identificación descrita en la realización 5 se usa probablemente

para identificar los datos de vídeo. La información de identificación no se limita a la descrita en la realización 5 sino que puede ser cualquier información siempre que la información indique a qué estándar se ajusta los datos de vídeo. Por ejemplo, cuando los datos de vídeo estándar se pueden determinar basándose en una señal externa para determinar que los datos de vídeo se utilizan para una televisión o un disco, etc., la determinación puede hacerse basándose en tal señal externa. Adicionalmente, la CPU ex502 selecciona una frecuencia de accionamiento basada en, por ejemplo, una tabla de búsqueda en la que los estándares de los datos de vídeo están asociados con las frecuencias de accionamiento como se muestra en la FIG. 56. La frecuencia de accionamiento puede seleccionarse almacenando la tabla de búsqueda en el búfer ex508 y en una memoria interna de un LSI, y con referencia a la tabla de búsqueda por la CPU ex502.

La FIG. 55 ilustra las etapas para ejecutar un procedimiento en la realización 7. En primer lugar, en la Etapa exS200, la unidad de procesamiento de señal ex507 obtiene información de identificación de los datos multiplexados. A continuación, en la Etapa exS201, la CPU ex502 determina si los datos de vídeo se generan o no basándose en la información de identificación mediante el procedimiento de codificación y el aparato de codificación descrito en cada una de las realizaciones. Cuando los datos de vídeo se generan mediante el procedimiento de codificación y el aparato de codificación descrito en cada una de las realizaciones, en la Etapa exS202, la CPU ex502 transmite una señal para establecer la frecuencia de accionamiento a una frecuencia de accionamiento más alta a la unidad de control de frecuencia de accionamiento ex512. Seguidamente, la unidad de control de frecuencia de accionamiento ex512 establece la frecuencia de accionamiento a la frecuencia de accionamiento más alta. Por otro lado, cuando la información de identificación indica que los datos de vídeo se ajustan al estándar convencional, tal como MPEG-2, MPEG4-AVC y VC-1, en la Etapa exS203, la CPU ex502 transmite una señal para establecer la frecuencia de accionamiento a una frecuencia de accionamiento menor que la unidad de control de frecuencia de accionamiento ex512. Seguidamente, la unidad de control de frecuencia de accionamiento ex512 establece la frecuencia de accionamiento a la frecuencia de accionamiento inferior que la de en el caso en el que los datos de vídeo se generan por el procedimiento de codificación de instantáneas en movimiento y el aparato de codificación de instantáneas en movimiento descritos en cada una de las realizaciones.

Adicionalmente, junto con la conmutación de las frecuencias de accionamiento, el efecto de conservación de potencia puede mejorarse cambiando la tensión que se aplicará al LSI ex500 o un aparato que incluya el LSI ex500. Por ejemplo, cuando la frecuencia de accionamiento es más baja, la tensión a aplicar en el LSI ex500 o el aparato que incluye el LSI ex500 se ajusta probablemente a una tensión inferior a la de en el caso donde la frecuencia de accionamiento se establece superior.

Adicionalmente, cuando la cantidad de computación para decodificar es mayor, la frecuencia de accionamiento puede establecerse más alta, y cuando la cantidad de computación para la decodificación es menor, la frecuencia de accionamiento puede establecerse más baja como procedimiento para establecer la frecuencia de accionamiento. De este modo, el procedimiento de ajuste no se limita a los anteriormente descritos. Por ejemplo, cuando la cantidad de cálculo para decodificar datos de vídeo de conformidad con MPEG4-AVC es mayor que la cantidad de cálculo para decodificar datos de vídeo generados por el procedimiento de codificación de imagen en movimiento y el aparato de codificación de imagen en movimiento descrito en cada una de las realizaciones, la frecuencia de accionamiento probablemente esté configurada en orden inverso al ajuste descrito anteriormente.

Adicionalmente, el procedimiento para establecer la frecuencia de accionamiento no se limita al procedimiento para establecer la frecuencia de accionamiento más baja. Por ejemplo, cuando la información de identificación indica que los datos de vídeo se generan mediante el procedimiento de codificación de imagen en movimiento y el aparato de codificación de imagen en movimiento descrito en cada una de las realizaciones, la tensión a aplicar en el LSI ex500 o el aparato que incluye el LSI ex500 se ajusta probablemente superior. Cuando la información de identificación indica que los datos de vídeo se ajustan al estándar convencional, tal como MPEG-2, MPEG4-AVC y VC-1, la tensión a aplicar en el LSI ex500 o el aparato que incluye el LSI ex500 se ajusta probablemente inferior. Como otro ejemplo, cuando la información de identificación indica que los datos de vídeo se generan mediante el procedimiento de codificación de imagen en movimiento y el aparato de codificación de imagen en movimiento descrito en cada una de las realizaciones, el accionamiento de la CPU ex502 probablemente no deba suspenderse. Cuando la información de identificación indica que los datos de vídeo se ajustan al estándar convencional, tal como MPEG-2, MPEG4-AVC y VC-1, el accionamiento de la CPU ex502 probablemente se suspende en un momento dado porque la CPU ex502 tiene capacidad de procesamiento adicional. Incluso la información de identificación indica que los datos de vídeo se generan mediante el procedimiento de codificación de imagen en movimiento y el aparato de codificación de imagen en movimiento descrito en cada una de las realizaciones, en el caso de que la CPU ex502 pueda tener un retraso de tiempo, el accionamiento de la CPU ex502 probablemente se suspende en un momento dado. En tal caso, el tiempo de suspensión probablemente se establezca más corto que en el caso en que la información de identificación indica que los datos de vídeo se ajustan al estándar convencional, tal como MPEG-2, MPEG4-AVC y VC-1.

Por consiguiente, el efecto de conservación de potencia puede mejorarse conmutando entre las frecuencias de accionamiento de acuerdo con la estándar al que se ajustan los datos de vídeo. Adicionalmente, cuando el LSI ex500 o el aparato que incluye el LSI ex500 se acciona con una batería, la duración de la batería se puede extender con el efecto de ahorro de potencia.

(Realización 8)

Hay casos en que una pluralidad de datos de vídeo que se ajusta a diferentes estándares, se proporciona a los dispositivos y sistemas, tal como una televisión y un teléfono celular. Para permitir la decodificación de la pluralidad de datos de vídeo que se ajusta a los diferentes estándares, la unidad de procesamiento de señal ex507 del LSI ex500 debe cumplir con los diferentes estándares. Sin embargo, los problemas de aumento en la escala del circuito del LSI ex500 y el aumento en el coste surgen con el uso individual de las unidades de procesamiento de señal ex507 que se ajustan a los estándares respectivos.

Para resolver el problema, lo que se concibe es una configuración en la cual la unidad de procesamiento de decodificación para implementar el procedimiento de decodificación de imagen en movimiento descrito en cada una de las realizaciones y la unidad de procesamiento de decodificación que se ajusta al estándar convencional, tal como MPEG-2, MPEG4-AVC y VC-1 se comparten parcialmente. Ex900 en la FIG. 57A muestra un ejemplo de la configuración. Por ejemplo, el procedimiento de decodificación de imagen en movimiento descrito en cada una de las realizaciones y el procedimiento de decodificación de imagen en movimiento que se ajusta a MPEG4-AVC tienen, en parte en común, los detalles del procesamiento, tal como la codificación de entropía, cuantización inversa, filtrado de desbloqueo y predicción de movimiento compensado. Los detalles de procesamiento a compartirse probablemente incluyen el uso de una unidad ex902 de procesamiento de decodificación que se ajusta a MPEG4-AVC. Por el contrario, una unidad de procesamiento de decodificación dedicada ex901 se usa probablemente para otro procesamiento exclusivo de la presente invención. Dado que la presente invención se caracteriza por la decodificación aritmética en particular, por ejemplo, la unidad de procesamiento de decodificación dedicada ex901 se usa para la decodificación aritmética. En caso contrario, la unidad de procesamiento de decodificación probablemente se comparte para una de las cuantizaciones inversas, filtrado de desbloqueo y compensación de movimiento, o todo el procesamiento. La unidad de procesamiento de decodificación para implementar el procedimiento de decodificación de imagen en movimiento descrita en cada una de las realizaciones puede compartirse para el procesamiento a compartir, y una unidad de procesamiento de decodificación dedicada puede usarse para procesamiento único al del de MPEG4-AVC.

Adicionalmente, ex1000 en la FIG. 57B muestra otro ejemplo en el que el procesamiento se comparte parcialmente. Este ejemplo usa una configuración que incluye una unidad de procesamiento de decodificación dedicada ex1001 que admite el procesamiento exclusivo de la presente invención, una unidad de procesamiento de decodificación dedicada ex1002 que soporta procesamiento único a otro estándar convencional y una unidad de procesamiento de decodificación ex1003 que soporta procesamiento a compartir entre el procedimiento de decodificación de imagen en movimiento en la presente invención y el procedimiento de decodificación de imagen en movimiento. En este punto, las unidades de procesamiento de decodificación especializadas ex1001 y ex1002 no están necesariamente especializadas para el procesamiento de acuerdo con el aspecto de la presente invención y el procesamiento del estándar convencional, respectivamente, y pueden ser las que pueden implementar procesamiento general. Adicionalmente, la configuración de la realización 8 puede implementarse por el LSI ex500.

En consecuencia, reducir la escala del circuito de un LSI y reducir el coste son posibles compartiendo la unidad de procesamiento de decodificación para el procesamiento a compartirse entre el procedimiento de decodificación de imagen en movimiento de acuerdo con el aspecto de la presente invención y el procedimiento de decodificación de imagen de instantáneas en movimiento de conformidad con el estándar convencional.

[Aplicabilidad industrial]

La presente invención es aplicable a un procedimiento de codificación de imagen, un procedimiento de decodificación de imagen, un aparato de codificación de imagen, y un aparato de decodificación de imagen, y en particular, es aplicable a un procedimiento de codificación de imagen, un procedimiento de decodificación de imagen, un aparato de codificación de imagen y un aparato de decodificación de imagen que usa codificación aritmética y decodificación aritmética.

[Lista de signos de referencia]

100	Aparato de codificación de imagen
101	Unidad de Control
102	Unidad de sustracción
103	Unidad de transformación y cuantización
104	Unidad de codificación de longitud variable
105	Unidad de cuantización inversa y transformación inversa
106, 206	Unidad de agregación
107, 207	Unidad de intrapredicción
108, 208	Unidad de interpredicción
109, 209	Conmutador
121	Señal de entrada de imagen
122, 125, 225	Señal residual
123, 223	Coeficientes de transformación cuantizados

124	Flujo de bits
126	Señal de imagen decodificada
127, 128, 129, 227, 228	Señal de predicción de imagen
130,	230 Parámetro de control
141	Unidad de binarización
142, 242	Unidad de control de contexto
143	Unidad de codificación aritmética binaria
151, 251	Secuencia binaria
152, 252	Índice de contexto
200	Aparato de decodificación de imagen
201	Unidad de Control
202	Unidad de decodificación de longitud variable
204	Unidad de cuantización inversa
205	Unidad de transformación inversa
224	Coefficientes de transformación ortogonal
226	Señal de imagen decodificada
229	Señal de imagen
241	Unidad de binarización inversa
243	Unidad de decodificación aritmética binaria

REIVINDICACIONES

1. Un procedimiento de decodificación para decodificar un parámetro de control para controlar la decodificación de una imagen, comprendiendo el procedimiento de decodificación:

5 determinar (S204) un contexto para un bloque actual en la imagen, de entre una pluralidad de contextos; y
realizar (S210) una decodificación aritmética en un flujo de bits correspondiente al bloque actual, usando el contexto determinado para obtener un parámetro de control para el bloque actual, usándose el parámetro de control para decodificar el bloque actual; en el que
la determinación (S204) incluye además:

10 determinar (S205) un elemento de sintaxis correspondiente al parámetro de control para el bloque actual;
determinar (S206) el contexto usando ambos parámetros de control decodificados para un bloque izquierdo y un bloque superior, cuando un tipo de contexto del elemento de sintaxis se categoriza en un primer tipo;
determinar (S208) el contexto mediante el uso de un valor fijo predeterminado, cuando el tipo de contexto se categoriza en un segundo tipo; y
15 determinar (S214) el contexto usando una profundidad jerárquica de una unidad de datos a la que pertenece el parámetro de control para el bloque actual, cuando el tipo de contexto se categoriza en un tercer tipo; en el que

el bloque izquierdo es un bloque vecino a la izquierda del bloque actual, y el bloque superior es un bloque vecino sobre la parte superior del bloque actual;
20 el segundo tipo es diferente del primer tipo, y el tercer tipo es diferente del primer tipo y del segundo tipo;
uno de una bandera de división y una bandera de salto se categoriza en el primer tipo;
una bandera de fusión y el modo de intrapredicción de crominancia se categorizan en el segundo tipo; y
una primera bandera de coeficiente (cbf_luma) y una segunda bandera de coeficiente (cbf_cb, cbf_cr) se categorizan en el tercer tipo;
25 en el que
la bandera de división indica si el bloque actual se divide o no en una pluralidad de bloques, y la bandera de salto indica si se debe saltar o no el bloque actual;
la bandera de fusión indica si se usa o no un modo de fusión para predecir el bloque actual, en el que el bloque actual se decodifica basándose en una predicción usando un vector de movimiento y un índice de imagen de referencia que se copian de un bloque vecino en el modo de fusión, y el modo de intrapredicción de crominancia se usa para la intrapredicción de crominancia del bloque actual;
30 la primera bandera de coeficiente (cbf_luma) y la segunda bandera de coeficiente (cbf_cb, cbf_cr) indican la presencia o ausencia de coeficientes de luminancia o crominancia; y
el número total de contextos de la segunda bandera de coeficiente (cbf_cb, cbf_cr) es menor que el número total de contextos de la primera bandera de coeficiente (cbf_luma).

35 2. El procedimiento de decodificación de acuerdo con la reivindicación 1, en el que el parámetro de control categorizado en el primer tipo pertenece a una unidad de datos de un tamaño mayor o igual a un tamaño de una unidad de datos a la que pertenece el parámetro de control categorizado en el segundo tipo.

40 3. El procedimiento de decodificación de acuerdo con la reivindicación 1 o la reivindicación 2, en el que en la determinación de un contexto (S204), se determina además que el parámetro de control decodificado para el bloque superior no está disponible en la decodificación, cuando el bloque actual está en un límite de corte.

4. Un aparato de decodificación para decodificar un parámetro de control para controlar la decodificación de una imagen, comprendiendo el aparato de decodificación:

45 una unidad de determinación de contexto (242) configurada para determinar un contexto para un bloque actual en la imagen, de entre una pluralidad de contextos; y
una unidad de decodificación aritmética (243) configurada para realizar la decodificación aritmética en un flujo de bits correspondiente al bloque actual, usando el contexto determinado para obtener un parámetro de control para el bloque actual, usándose el parámetro de control para decodificar el bloque actual; en el que
la unidad de determinación de contexto (242) está configurada para ejecutar el procedimiento de decodificación de acuerdo con cualquiera de las reivindicaciones 1 a 3.

FIG. 1

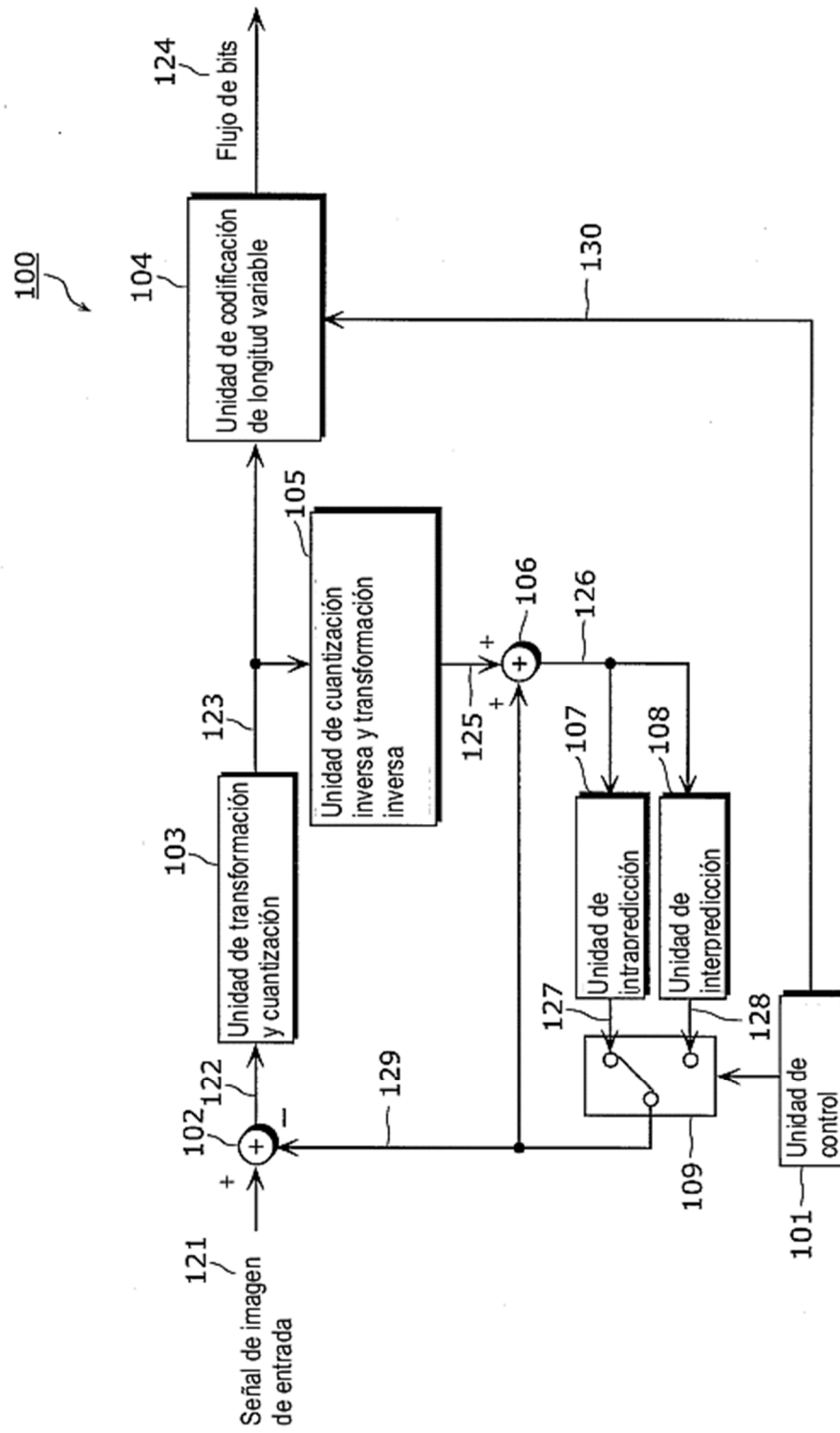


FIG. 2

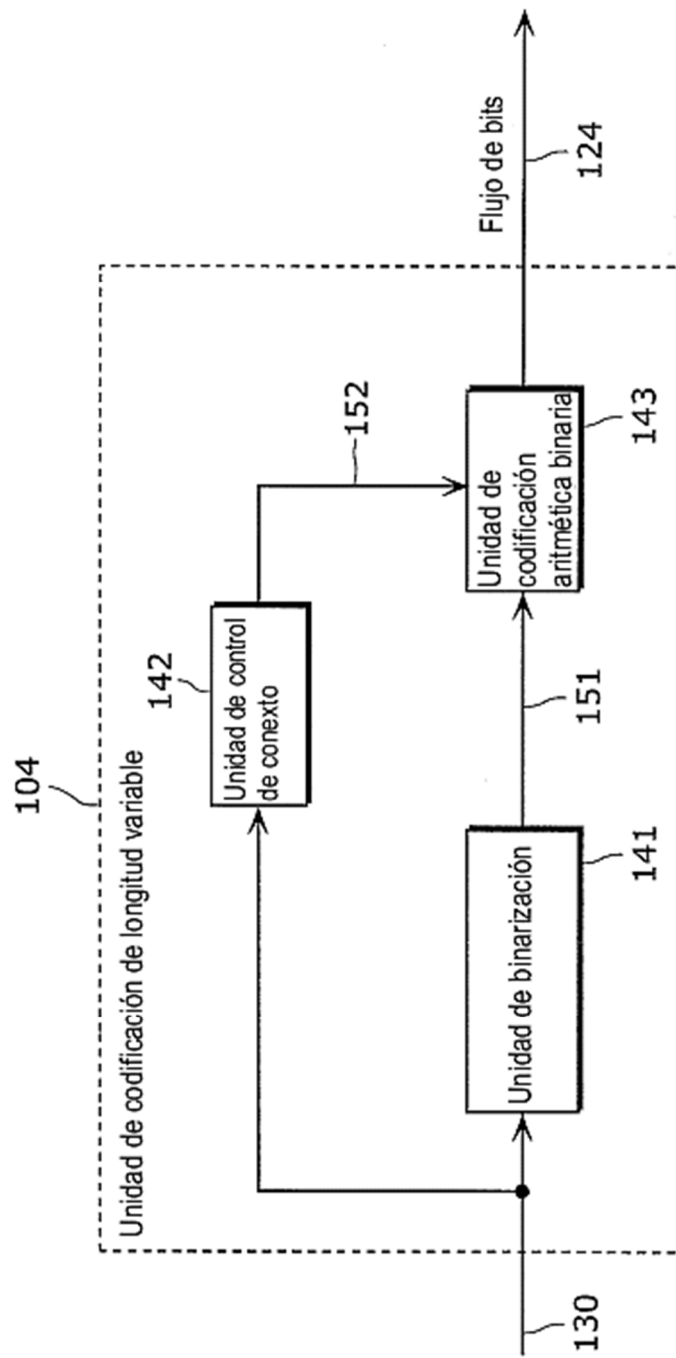


FIG. 3

	c2 Tipo de señal	c3 Esquema de binarización	c4 Modelo de contexto para primer elemento	c5 Condición de bloque izquierdo L condL	c6 Condición de bloque superior A condA
s1	split_coding_unit_flag	Longitud fija	0, 1, 0, 2 (condL && availableL) + (condA && availableA)	cuDepth[xL][yL] > cuDepth[xP][yP]	cuDepth[xA][yA] > cuDepth[xP][yP]
s2	skip_flag	Longitud fija	0, 1, 2	skip_flag[xL][yL]	skip_flag[xA][yA]
s3	merge_flag	Longitud fija	0, 1, 2	merge_flag[xL][yL]	merge_flag[xA][yA]
s4	intra_chroma_pred_mode	Longitud variable	0, 1, 2	IntraPredMode[xL][yL] == 4	IntraPredMode[xA][yA] == 4
s5	inter_pred_flag	Longitud fija	0, 1, 2	inter_pred_flag[xL][yL]	inter_pred_flag[xA][yA]
s6	mvd_lc mvd_l0 mvd_l1	Longitud variable	0, 1, 2	mvd_lc[xL][yL] > 16 mvd_l0[xL][yL] > 16 mvd_l1[xL][yL] > 16	mvd_lc[xA][yA] > 16 mvd_l0[xA][yA] > 16 mvd_l1[xA][yA] > 16
s7	ref_idx_lc ref_idx_l0 ref_idx_l1	Longitud variable	0, 1, 2, 0, 3 (condL && availableL) + (condA && availableA) < 1	ref_idx_lc[xL][yL] > 0 ref_idx_l0[xL][yL] > 0 ref_idx_l1[xL][yL] > 0	ref_idx_lc[xA][yA] > 0 ref_idx_l0[xA][yA] > 0 ref_idx_l1[xA][yA] > 0
s8	Cbf_cb(MODE_INTRA)	Longitud fija	0, 1, 2, 3	cbf_cb[xL][yL]	cbf_cb[xA][yA]
s9	Cbf_cr(MODE_INTRA)	Longitud fija	0, 1, 2, 3	cbf_cr[xL][yL]	cbf_cr[xA][yA]
s10	Cbf_luma	Longitud fija	0, 1, 2, 3	cbf_luma[xL][yL]	cbf_luma[xA][yA]
s10	no_residual_data_flag	Longitud fija	0, 1, 2, 3	no_residual_data_flag[xL][yL]	no_residual_data_flag[xA][yA]
NA	alf_cu_flag	1 binario	0, 1, 2	alf_cu_flag[xL][yL]	alf_cu_flag[xA][yA]

FIG. 4

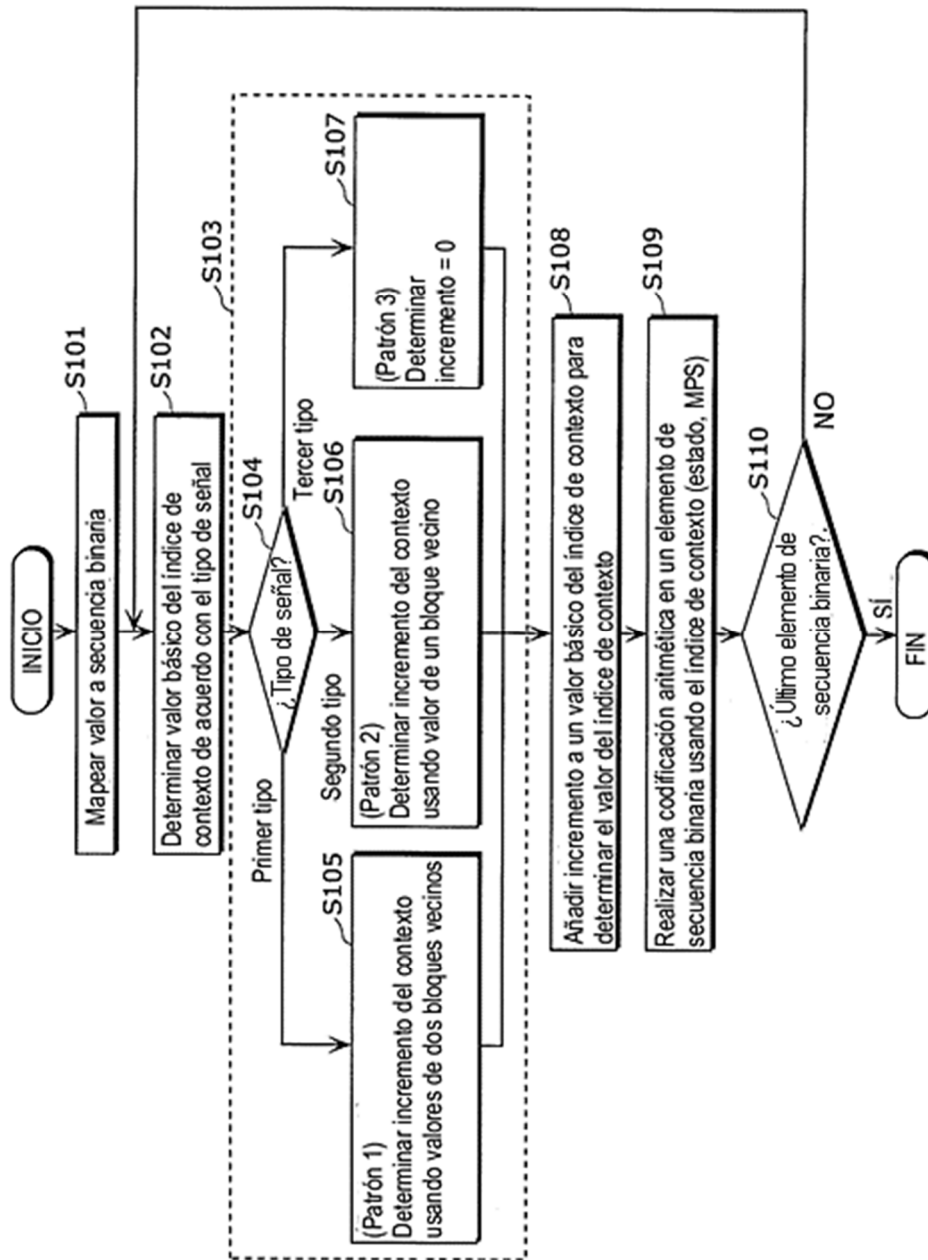


FIG. 5

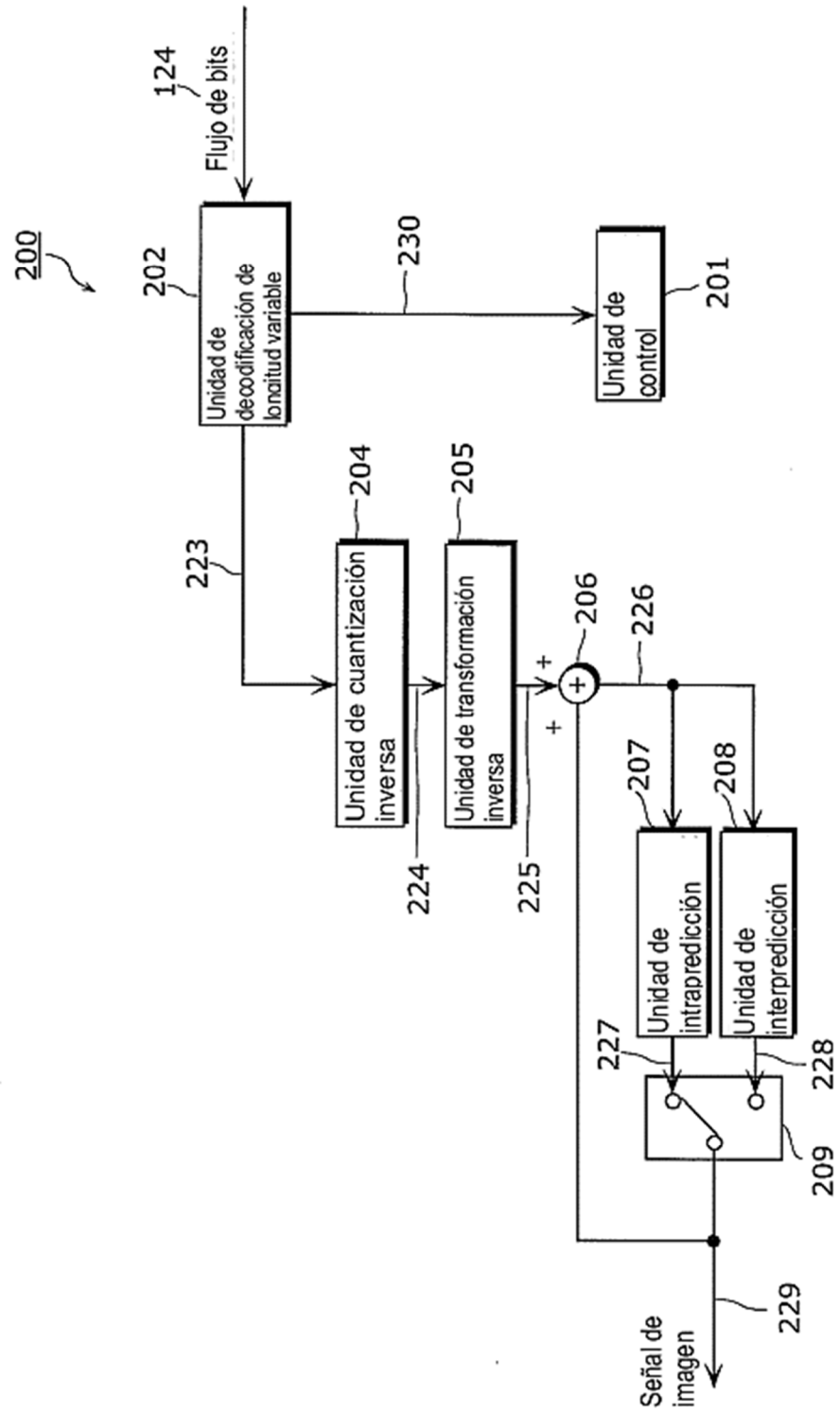


FIG. 6

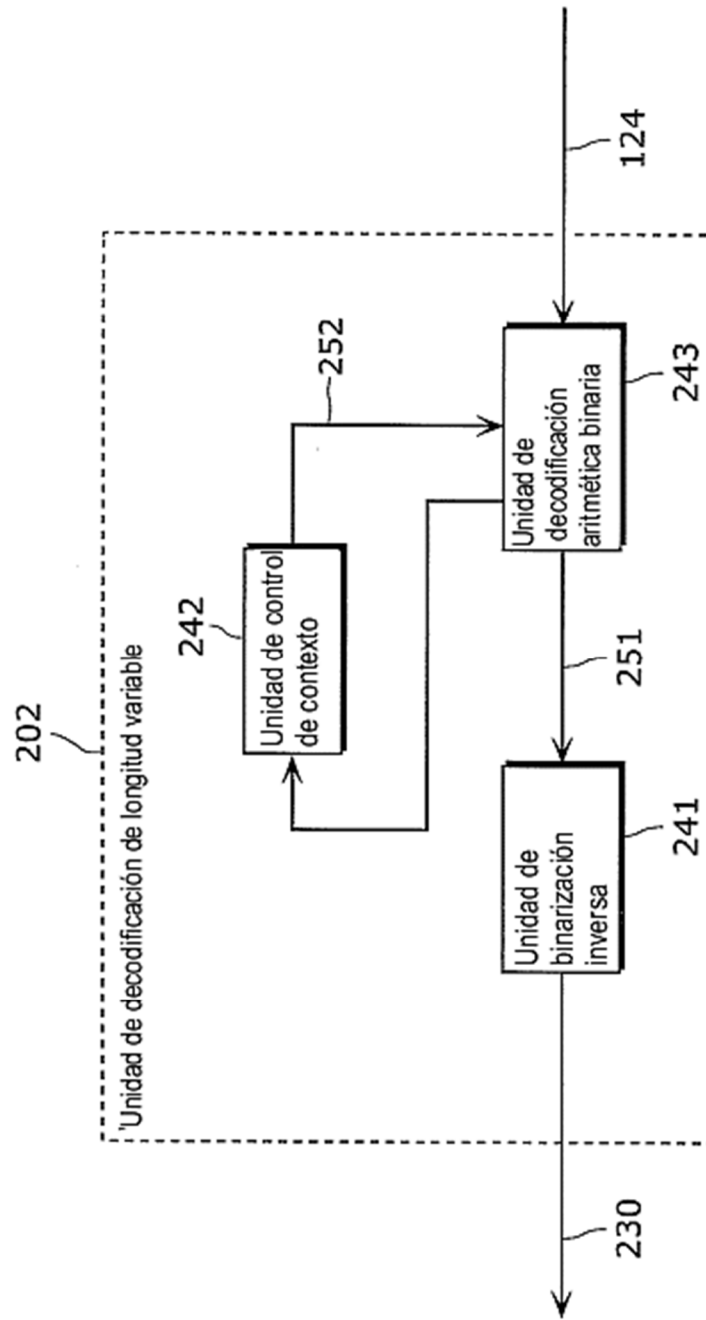


FIG. 7

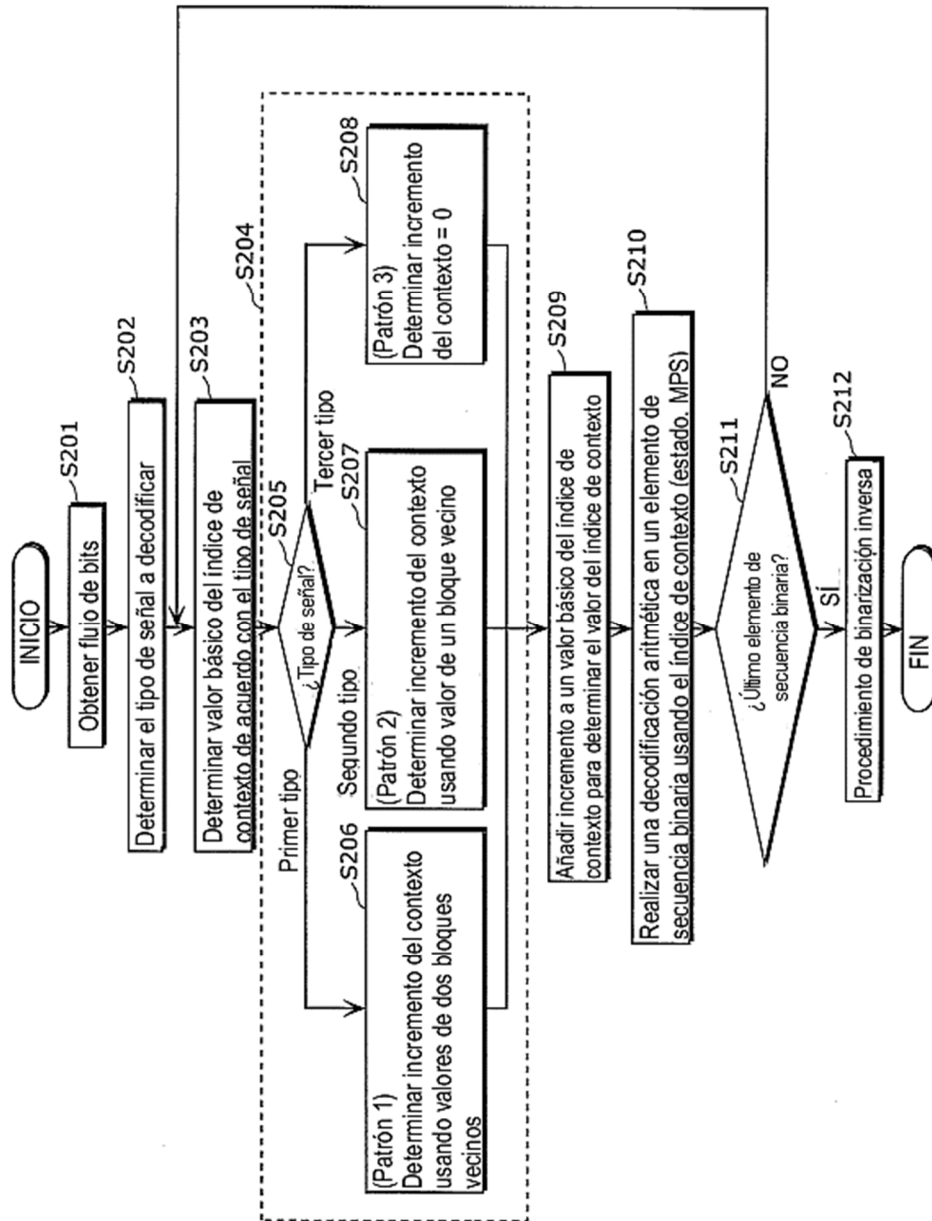


FIG. 8

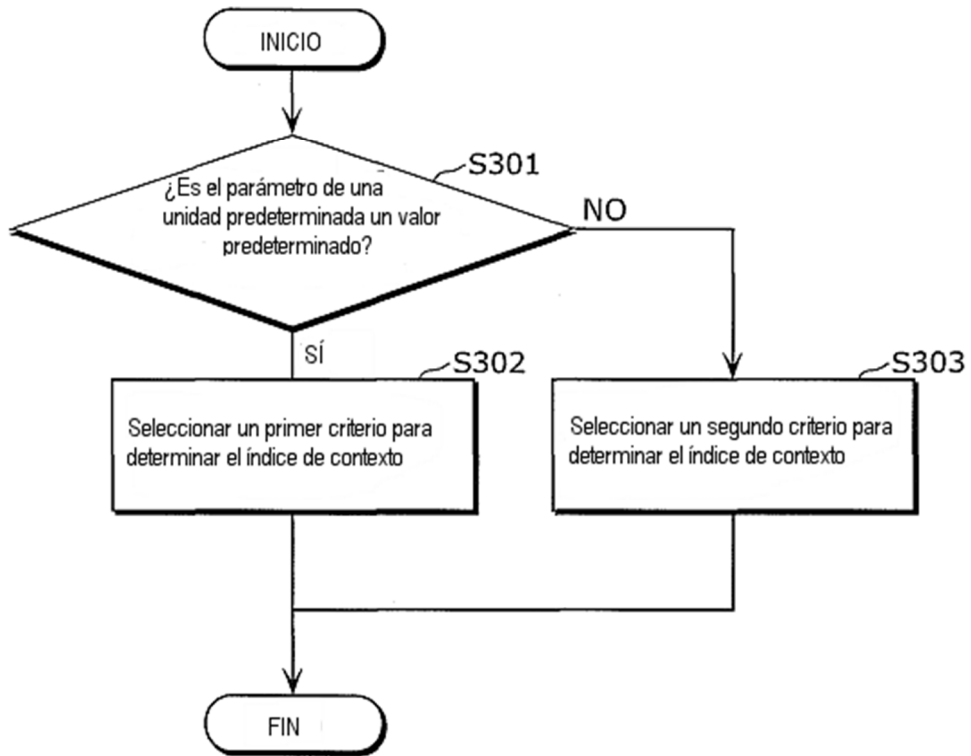


FIG. 9B

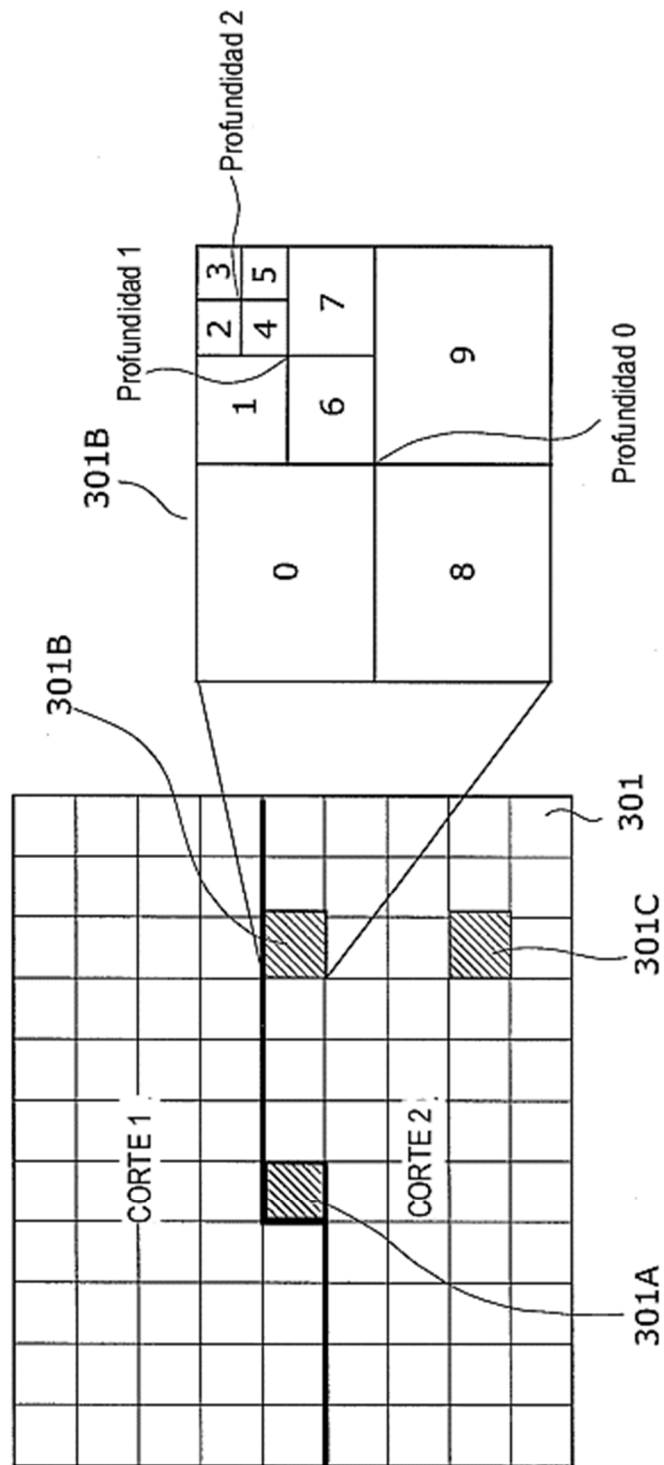


FIG. 10

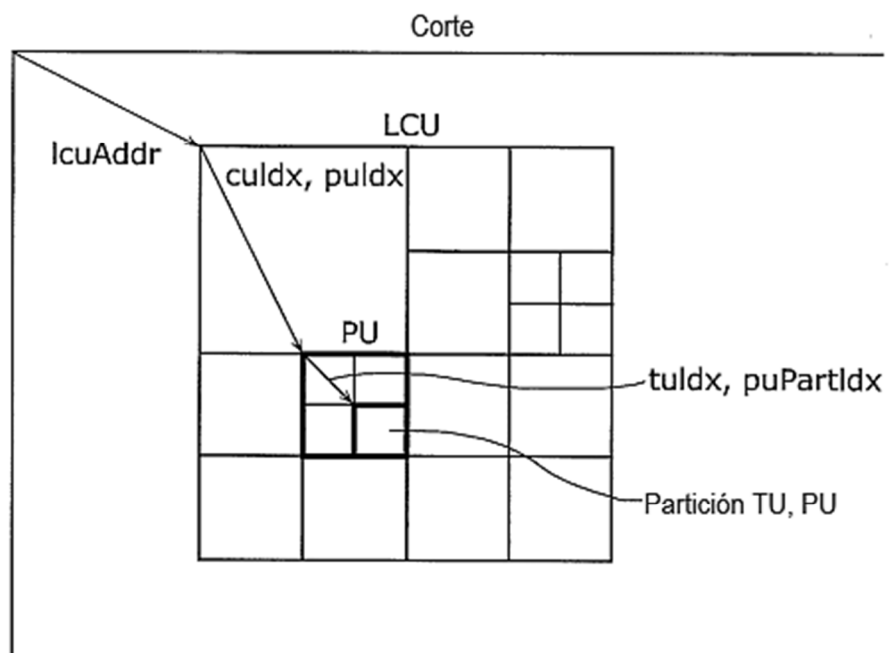


FIG. 11

Patrón de verificación	Tipo de señal (elemento de sintaxis)	Esquema de binarización	Incremento del índice de contexto del elemento de secuencia binaria (binIdx = 0)	c5 Condición de bloque izquierdo L condL	c6 Condición de bloque superior A condA
Patrón 1	split_coding_unit_flag	Longitud fija	0, 1, o 2 (condL && availableL) + (condA && availableA)	$\text{cuDepth}[xL][yL] > \text{cuDepth}[xP][yP]$	$\text{cuDepth}[xA][yA] > \text{cuDepth}[xP][yP]$
Patrón 2			0 o 1 (condL && availableL)	$\text{cuDepth}[xL][yL] > \text{cuDepth}[xP][yP]$	Fijado (no guardado en memoria)
Patrón 3			Fijado a 0	Fijado	Fijado (no guardado en memoria)

FIG. 12A

Patrón 2: usando solo bloque izquierdo (split_coding_unit_flag)

	Intra				Acceso aleatorio				Retardo bajo				Retardo bajo (P)			
	Y tasa BD	U tasa BD	V tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	V tasa BD
Clase A	0,02	-0,07	0,04	0,04	0,09	0,56	-0,12	0,18	0,15	0,33	0,28	0,16	0,16	0,17	-0,13	0,13
Clase B	0,02	0,04	0,03	0,06	0,06	0,08	0,18	0,20	0,16	0,22	0,08	0,25	0,25	0,23	0,69	0,69
Clase C	0,03	0,09	0,06	0,06	0,18	0,16	0,20	0,18	0,06	0,35	0,87	0,12	0,12	-0,21	0,85	0,85
Clase D	0,05	-0,05	0,06	0,06	0,32	0,39	0,18	0,18	0,36	0,79	-0,89	0,38	0,38	0,90	-0,72	-0,72
Clase E	0,13	-0,02	0,01	0,01	0,17	0,18	0,18	0,18	0,17	0,40	0,16	0,21	0,21	0,22	0,21	0,21
Todo	0,05	0,01	0,03	0,03	0,17	0,18	0,18	0,18	0,17	0,40	0,16	0,21	0,21	0,22	0,21	0,21
Tiempo de cod [%]	100 %				99 %				99 %				99 %			
Tiempo de dec [%]	101 %				103 %				102 %				103 %			

FIG. 12B

Patrón 3: no usando ni bloque izquierdo ni bloque superior (split_coding_unit_flag)

	Intra				Acceso aleatorio				Retardo bajo				Retardo bajo (P)			
	Y tasa BD	U tasa BD	V tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	V tasa BD
Clase A	0,05	-0,05	0,01	0,01	0,15	0,56	-0,27	0,49	0,32	0,20	0,53	0,36	0,36	0,27	0,54	0,54
Clase B	0,11	-0,01	0,04	0,04	0,23	0,16	0,49	0,23	0,32	0,20	0,35	0,22	0,22	0,05	0,52	0,52
Clase C	0,04	0,09	0,13	0,13	0,18	0,46	0,23	0,23	0,25	0,57	0,65	0,26	0,26	0,41	0,46	0,46
Clase D	0,07	-0,05	0,02	0,02	0,29	-0,13	-0,05	-0,05	0,65	0,44	-0,65	0,61	0,61	1,03	-0,06	-0,06
Clase E	0,20	-0,13	-0,33	-0,33	0,24	0,17	0,22	0,22	0,36	0,34	0,29	0,35	0,35	0,39	0,41	0,41
Todo	0,10	-0,01	-0,01	-0,01	0,24	0,17	0,22	0,22	0,36	0,34	0,29	0,35	0,35	0,39	0,41	0,41
Tiempo de cod [%]	100 %				99 %				99 %				99 %			
Tiempo de dec [%]	101 %				103 %				103 %				103 %			

FIG. 13

Número de verificación	Tipo de señal (elemento de sintaxis)	Esquema de binarización	Incremento del índice de contexto del elemento de secuencia binaria (binIdx = 0)	C5 Condición de bloque izquierdo L condL	C6 Condición de bloque superior A condA
Patrón 1	skip_flag	Longitud fija	0, 1, 2	skip_flag[xL][yL]	skip_flag[xA][yA]
Patrón 2			0 o. 1	skip_flag[xL][yL]	Fijado
Patrón 3			Fijado a 0	Fijado	Fijado

FIG. 14A

Patrón 2: usando solo bloque izquierdo (skip_flag)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	0,05	-0,23	-0,01	0,12	0,21	0,23	0,07	-0,03	-0,04
Clase B	0,00	0,00	0,00	0,11	0,03	0,02	0,08	0,24	0,11	0,11	0,19	0,44
Clase C	0,00	0,00	0,00	0,06	0,16	0,10	0,09	0,39	0,07	0,26	-0,09	0,37
Clase D	0,00	0,00	0,00	0,11	-0,11	-0,10	0,03	0,13	-0,33	0,28	0,32	0,16
Clase E	0,00	0,00	0,00									
Todo	0,00	0,00	0,00	0,09	-0,02	-0,02	0,08	0,25	0,05	0,17	0,07	0,22
Tiempo de cod [%]	1,00			0,99			0,99			0,99		
Tiempo de dec [%]												
	101 %			103 %			102 %			103 %		

FIG. 14B

Patrón 3: no usando ni bloque izquierdo ni bloque superior (skip_flag)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	0,11	0,11	-0,04	0,20	0,45	0,15	0,17	0,32	0,36
Clase B	0,00	0,00	0,00	0,21	0,06	0,14	-0,01	0,06	0,12	0,15	0,03	0,18
Clase C	0,00	0,00	0,00	0,04	0,20	-0,17	0,26	0,20	0,26	0,16	-0,32	-0,18
Clase D	0,00	0,00	0,00	0,15	-0,31	-0,08	0,22	-0,05	-0,20	0,31	0,31	-0,07
Clase E	0,00	0,00	0,00									
Todo	0,00	0,00	0,00	0,14	-0,00	-0,02	0,17	0,20	0,11	0,19	0,09	0,10
Tiempo de cod [%]	1,00			0,99			0,99			0,99		
Tiempo de dec [%]												
	101 %			103 %			102 %			103 %		

FIG. 15

Número de verificación	Tipo de señal (elemento de sintaxis)	Esquema de binarización	Incremento del índice de contexto del elemento de secuencia binaria (binIdx = 0)	c5 Condición de bloque izquierdo L condL	c6 Condición de bloque superior A condA
Patrón 1	merge_flag	Longitud fija	0, 1, 2	merge_flag[xL][yL]	merge_flag[xA][yA]
Patrón 2			0 o 1	merge_flag[xL][yL]	Fijado
Patrón 3			Fijado a 0	Fijado	Fijado

FIG. 16A

Patrón 2: usando solo bloque izquierdo (merge_flag)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	0,00	-0,00	0,09	-0,06	0,22	0,16	-0,03	0,07	0,12
Clase B	0,00	0,00	0,00	-0,03	-0,08	0,11	-0,09	0,06	0,15	0,05	0,09	0,23
Clase C	0,00	0,00	0,00	-0,04	0,19	0,06	-0,01	-0,02	0,04	-0,00	-0,20	0,33
Clase D	0,00	0,00	0,00	-0,01	-0,47	-0,26	0,07	0,04	-0,40	0,06	0,05	-0,51
Clase E	0,00	0,00	0,00									
Todo	0,00	0,00	0,00	-0,02	-0,11	-0,03	-0,03	0,09	0,02	0,01	0,00	0,08
Tiempo de cod [%]	100 %			99 %			99 %			99 %		
Tiempo de dec [%]	101 %			103 %			102 %			103 %		

FIG. 16B

Patrón 3: no usando ni bloque izquierdo ni bloque superior (merge_flag)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	0,03	0,16	0,05	-0,03	0,19	0,05	-0,04	0,37	0,34
Clase B	0,00	0,00	0,00	0,02	0,06	0,27	-0,03	0,21	0,10	0,06	-0,03	0,36
Clase C	0,00	0,00	0,00	-0,01	0,19	0,11	-0,02	-0,09	0,01	-0,02	-0,57	-0,09
Clase D	0,00	0,00	0,00	0,02	-0,04	-0,33	0,02	0,18	-0,41	0,05	0,29	-0,34
Clase E	0,00	0,00	0,00				-0,02	0,12	-0,03	0,01	0,02	0,11
Todo	0,00	0,00	0,00	0,01	0,05	0,04	-0,02	0,12	-0,03			
Tiempo de cod [%]	100 %			99 %			99 %			99 %		
Tiempo de dec [%]	101 %			103 %			103 %			103 %		

FIG. 17

Número de verificación	Tipo de señal (elemento de sintaxis)	Esquema de binarización	Incremento del índice de contexto del elemento de secuencia binaria (binIdx = 0)	c5 Condición de bloque izquierdo L condL	c6 Condición de bloque superior A condA
Patrón 1	ref_idx_lc	Longitud variable	0, 1, 2, o 3	ref_idx_lc[xL][yL] > 0	ref_idx_lc[xA][yA] > 0
	ref_idx_l0		00: condAfalse condLtrue	ref_idx_l0[xL][yL] > 0	ref_idx_l0[xA][yA] > 0
	ref_idx_l1		10: condAtrue condLfalse 11: condAtrue condLtrue	ref_idx_l1[xL][yL] > 0	ref_idx_l1[xA][yA] > 0
Patrón 2	ref_idx_lc	Longitud variable	fixed_0: condLfalse fixed_1: condLtrue	ref_idx_lc[xL][yL] > 0	Fijado
	ref_idx_l0			ref_idx_l0[xL][yL] > 0	
	ref_idx_l1			ref_idx_l1[xL][yL] > 0	
Patrón 3	ref_idx_lc	Longitud variable	Fijado	Fijado	Fijado
	ref_idx_l0				
	ref_idx_l1				

FIG. 18A

Patrón 2: usando solo bloque izquierdo (ref_idx)

ref_idx	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	-0,01	-0,09	-0,14	-0,07	0,16	0,14	-0,05	0,11	0,20
Clase B	0,00	0,00	0,00	-0,01	-0,06	-0,07	0,01	0,03	0,08	-0,05	-0,21	0,06
Clase C	0,00	0,00	0,00	0,04	0,11	-0,10	-0,04	0,41	0,18	-0,07	-0,62	-0,41
Clase D	0,00	0,00	0,00	0,13	-0,25	-0,09	0,09	0,09	-0,47	-0,04	-0,08	-0,11
Clase E	0,00	0,00	0,00				-0,01	0,18	0,02	-0,05	-0,19	-0,05
Todo	0,00	0,00	0,00	0,04	-0,07	-0,07						
Tiempo de cod [%]		100 %			99 %			99 %			99 %	
Tiempo de dec [%]		101 %			103 %			103 %			103 %	

FIG. 18B

Patrón 3: no usando ni bloque izquierdo ni bloque superior (ref_idx)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	0,02	0,23	-0,02	-0,03	0,11	-0,00	-0,03	0,05	0,14
Clase B	0,00	0,00	0,00	-0,02	0,06	0,01	-0,03	0,02	0,12	0,05	0,01	0,07
Clase C	0,00	0,00	0,00	-0,04	0,04	0,13	-0,09	0,39	-0,06	0,01	-0,22	0,10
Clase D	0,00	0,00	0,00	0,04	-0,60	-0,24	0,06	-0,15	-0,55	0,02	0,19	-0,53
Clase E	0,00	0,00	0,00				-0,02	0,11	-0,09	0,01	-0,00	-0,01
Todo	0,00	0,00	0,00	-0,01	-0,14	-0,02						
Tiempo de cod [%]		100 %			99 %			99 %			99 %	
Tiempo de dec [%]		101 %			103 %			103 %			103 %	

FIG. 18C

Tipo de señal (elemento de sintaxis)	Modelo de contexto	Condición de bloque izquierdo L condL	Condición de bloque superior A condA
ref_idx_lc	(condL && availableL) + (condA && availableA) < 1	ref_idx_lc[xL][yL] > 0	ref_idx_lc[xA][yA] > 0
ref_idx_l0	(condL && availableL) + (condA && availableA) < 1	ref_idx_l0[xL][yL] > 0	ref_idx_l0[xA][yA] > 0
ref_idx_l1	(condL && availableL) + (condA && availableA) < 1	ref_idx_l1[xL][yL] > 0	ref_idx_l1[xA][yA] > 0

FIG. 18D

Tipo de señal (elemento de sintaxis)	Modelo de contexto	Condición de bloque izquierdo L condL	Condición de bloque superior A condA
ref_idx_lc	Derivado de l0 e l1	Derivado de l0 e l1	Derivado de l0 e l1
ref_idx_l0	(condL && availableL) + (condA && availableA) < 1	ref_idx_l0[xL][yL] > 0	ref_idx_l0[xA][yA] > 0
ref_idx_l1	(condL && availableL) + (condA && availableA) < 1	ref_idx_l1[xL][yL] > 0	ref_idx_l1[xA][yA] > 0

FIG. 19

Patrón de verificación	Tipo de señal (elemento de sintaxis)	Esquema de binarización	Incremento del índice de contexto del elemento de secuencia binaria element (binIdx = 0) 0, 1, or 2 (condL && availableL) + (condA && availableA) (condL && availableL)	c5 Condición de bloque izquierdo L condL	c6 Condición de bloque superior A condA
Patrón 1	inter_pred_flag	Longitud fija	(condL && availableL) + (condA && availableA)	inter_pred_flag[xL][yL]	inter_pred_flag[xA][yA]
Patrón 2			(condL && availableL)	inter_pred_flag[xL][yL]	Fijado
Patrón 3			Fijado	Fijado	Fijado

FIG. 20A

Patrón 2: usando solo bloque izquierdo (inter_pred_flag)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	0,01	-0,07	-0,06	0,00	0,06	-0,06	0,00	0,00	0,00
Clase B	0,00	0,00	0,00	0,02	0,03	-0,02	-0,02	0,03	0,22	0,00	0,00	0,00
Clase C	0,00	0,00	0,00	-0,02	-0,02	0,02	-0,02	0,18	-0,01	0,00	0,00	0,00
Clase D	0,00	0,00	0,00	0,08	-0,06	-0,12	-0,01	-0,28	-0,75	0,00	0,00	0,00
Clase E	0,00	0,00	0,00				0,02	0,02	-0,11	0,00	0,00	0,00
Todo	0,00	0,00	0,00	0,03	-0,04	-0,04	-0,00	99 %		0,00	99 %	
Tiempo de cod [%]								103 %			103 %	
Tiempo de dec [%]												

FIG. 20B

Patrón 3: no usando ni bloque izquierdo ni bloque superior (inter_pred_flag)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	0,01	-0,14	-0,14	-0,02	0,18	-0,16	0,00	0,00	0,00
Clase B	0,00	0,00	0,00	-0,02	-0,12	0,01	-0,04	-0,10	-0,09	0,00	0,00	0,00
Clase C	0,00	0,00	0,00	-0,00	0,08	0,34	-0,07	0,30	-0,09	0,00	0,00	0,00
Clase D	0,00	0,00	0,00	0,00	-0,22	-0,12	0,05	0,57	-0,51	0,00	0,00	0,00
Clase E	0,00	0,00	0,00				-0,02	0,21	-0,19	0,00	0,00	0,00
Todo	0,00	0,00	0,00	-0,01	-0,10	0,06		99 %		0,00	99 %	
Tiempo de cod [%]								103 %			103 %	
Tiempo de dec [%]												

FIG. 21

Número de verificación	Tipo de señal (elemento de sintaxis)	Esquema de binarización	Incremento del índice de contexto del elemento de secuencia binaria (binIdx = 0)	c5 Condición de bloque izquierdo L condL	c6 Condición de bloque superior A conda
Patrón 1	mvd_lc	Longitud variable	0, 1, o 2 (condL && availableL) + (conda && availableA)	mvd_lc[xL][yL] > 16	mvd_lc[xA][yA] > 16
	mvd_l0			mvd_l0[xL][yL] > 16	mvd_l0[xA][yA] > 16
	mvd_l1			mvd_l1[xL][yL] > 16	mvd_l1[xA][yA] > 16
Patrón 2	mvd_lc	Longitud variable	(condL && availableL)	mvd_lc[xL][yL] > 16	Fijado
	mvd_l0			mvd_l0[xL][yL] > 16	
	mvd_l1			mvd_l1[xL][yL] > 16	
Patrón 3	mvd_lc	Longitud variable	Fijado	Fijado	Fijado
	mvd_l0				
	mvd_l1				

FIG. 22A

Patrón 2: usando solo bloque izquierdo (mvd)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	0,02	0,20	-0,06	0,01	0,19	0,11	-0,04	0,21	0,19
Clase B	0,00	0,00	0,00	-0,01	-0,09	-0,13	-0,06	-0,03	0,05	-0,01	0,04	0,20
Clase C	0,00	0,00	0,00	0,01	0,00	-0,14	-0,06	-0,02	-0,16	0,02	-0,35	-0,41
Clase D	0,00	0,00	0,00	0,10	-0,22	0,08	-0,06	0,24	-0,39	0,05	0,56	-0,04
Clase E	0,00	0,00	0,00				0,13					
Todo	0,00	0,00	0,00	0,02	-0,11	-0,07	-0,00	0,09	-0,07	0,00	0,09	0,00
Tiempo de cod [%]	100 %			99 %			99 %			99 %		
Tiempo de dec [%]	101 %			103 %			103 %			103 %		

FIG. 22B

Patrón 3: no usando ni bloque izquierdo ni bloque superior (mvd)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,00	0,00	0,00	0,00	0,13	0,08	0,05	0,21	0,14	-0,07	-0,02	0,05
Clase B	0,00	0,00	0,00	-0,03	-0,02	-0,11	-0,04	-0,20	0,21	0,06	0,00	0,29
Clase C	0,00	0,00	0,00	-0,04	0,25	-0,12	0,05	-0,01	-0,08	-0,02	-0,22	-0,29
Clase D	0,00	0,00	0,00	0,01	-0,13	0,15	0,11	-0,00	-0,44	-0,04	-0,17	-0,39
Clase E	0,00	0,00	0,00									
Todo	0,00	0,00	0,00	-0,02	0,03	-0,04	0,04	0,01	-0,01	-0,02	-0,09	-0,06
Tiempo de cod [%]	100 %			99 %			99 %			99 %		
Tiempo de dec [%]	101 %			103 %			103 %			103 %		

FIG. 22C

Tipo de señal (elemento de sintaxis)	Modelo de contexto	Condición de bloque izquierdo L condL	Condición de bloque superior A conda
mvd_lc	(condL && availableL) + (conda && availableA)	mvd_lc[xL][yL] > 16	mvd_lc[xA][yA] > 16
mvd_l0	(condL && availableL) + (conda && availableA)	mvd_l0[xL][yL] > 16	mvd_l0[xA][yA] > 16
mvd_l1	(condL && availableL) + (conda && availableA)	mvd_l1[xL][yL] > 16	mvd_l1[xA][yA] > 16

FIG. 22D

Tipo de señal (elemento de sintaxis)	Modelo de contexto	Condición de bloque izquierdo L condL	Condición de bloque superior A conda
mvd_lc	Derivado de l0 e l1	Derivado de l0 e l1	Derivado de l0 e l1
mvd_l0	(condL && availableL) + (conda && availableA)	mvd_l0[xL][yL] > 16	mvd_l0[xA][yA] > 16
mvd_l1	(condL && availableL) + (conda && availableA)	mvd_l1[xL][yL] > 16	mvd_l1[xA][yA] > 16

FIG. 23A

Número de verificación	Tipo de señal (elemento de sintaxis)	Esquema de binarización	Incremento del índice de contexto del elemento de secuencia binaria element (binIdx = 0) 0, 1, 2, or 3	c5 Condición de bloque izquierdo L condL	c6 Condición de bloque superior A condA
Patrón 1	no_residual_data_flag	Longitud fija	00: condAfalse condLfalse 01: condAfalse condLtrue 10: condAtrue condLfalse 11: condAtrue condLtrue	no_residual_data_flag [xL][yL]	no_residual_data_flag [xA][yA]
Patrón 2			fixed_0: condLfalse fixed_1: condLtrue	no_residual_data_flag [xL][yL]	Fijado
Patrón 3			Fijado	Fijado	Fijado

FIG. 23B

Sintaxis del árbol de transformación

Transform_tree(x0, y0, log2TrafoSize,trafoDepth, blkIdx){	Descriptor
si (entropy_coding_mode_flag && trafoDepth == 0 && IntraSplitFlag == 0)	
{	
si (Predmode != MODE_INTRA)	
no_residual_data_flag	u(1) ae(v)
residualDataPresentFlag = !no_residual_data_flag	

FIG. 25A

Número de verificación	Tipo de señal (elemento de sintaxis)	Esquema de binarización	Incremento del índice de contexto del elemento de secuencia binaria (binIdx = 0)	c5 Condición de bloque izquierdo L condL	c6 Condición de bloque superior A condA
Patrón 1	intra_chroma_pred_mode	Longitud variable	0, 1, o 2 (condL && availableL) + (condA && availableA)	intraPredMode[xL][yL] == 4	intraPredMode[xA][yA] == 4
Patrón 2			(condL && availableL)	intraPredMode[xL][yL] == 4	Fijado
Patrón 3			Fijado	Fijado	Fijado

FIG. 25B

intra_chroma_pred_mode	intraPredMode[xB][yB]				
	0	1	2	3	X(4 ≤ X < 34)
0	0	1	2	3	X
1	1	0	0	0	0
2	2	2	1	1	1
3	3	3	2	2	2
4	n/a	n/a	n/a	n/a	3

FIG. 26A

Patrón 2: usando sólo bloque izquierdo (intra_chroma_pred_mode)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,02	-0,17	0,00	0,01	0,51	-0,09	0,01	0,15	0,03	-0,03	-0,11	-0,02
Clase B	-0,00	0,06	-0,05	-0,01	-0,01	0,20	-0,02	-0,07	0,09	0,02	-0,04	0,11
Clase C	0,00	-0,02	0,01	-0,01	0,57	-0,19	-0,12	-0,59	-0,63	-0,13	-0,32	-0,81
Clase D	0,02	-0,12	-0,19	0,08	-0,04	0,01	-0,12	0,69	-1,13	-0,12	0,72	-0,20
Clase E	0,03	-0,12	0,12				-0,12	0,69	-1,13	-0,12	0,72	-0,20
Todo	0,01	-0,04	-0,04	0,02	0,16	0,04	-0,06	0,01	-0,34	-0,08	0,01	-0,22
Tiempo de cod [%]	100 %			99 %			99 %			99 %		
Tiempo de dec [%]	101 %			103 %			103 %			103 %		

FIG. 26B

Patrón 3: no usando ni bloque izquierdo ni bloque superior (intra_chroma_pred_mode)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,03	0,05	-0,05	-0,03	0,85	-0,99	0,00	-0,22	0,49	-0,02	-0,19	0,69
Clase B	-0,00	0,00	-0,03	-0,06	0,14	0,08	0,01	0,25	0,42	0,00	0,11	0,67
Clase C	0,01	0,09	0,06	-0,11	0,38	0,12	0,08	0,46	-0,70	-0,04	-0,33	-0,39
Clase D	-0,02	0,01	-0,15	0,04	-0,18	0,44	-0,11	-0,17	-0,81	-0,09	-0,04	-1,49
Clase E	-0,00	-0,10	0,26				0,00	0,08	-0,26	-0,03	-0,12	0,01
Todo	-0,00	0,02	0,02	-0,04	0,10	0,15						
Tiempo de cod [%]	100 %			99 %			99 %			99 %		
Tiempo de dec [%]	101 %			103 %			103 %			104 %		

FIG. 27

Número de verificación	Tipo de señal (elemento de sintaxis)	Esquema de binarización	Incremento del índice de contexto del elemento de secuencia binaria (binIdx = 0)	c5 Condición de bloque izquierdo L condL	c6 Condición de bloque superior A condA
Patrón 1	Cbf_cb(MODE_INTRA)	Longitud fija	0, 1, 2, o 3 00: condAfalse condLfalse 01: condAfalse condLtrue 10: condAtrue condLfalse 11: condAtrue condLtrue	Cbf_cb[xL][yL]	Cbf_cb[xA][yA]
	Cbf_cr(MODE_INTRA)			Cbf_cr[xL][yL]	Cbf_cr[xA][yA]
	Cbf_luma			Cbf_luma[xL][yL]	Cbf_luma[xA][yA]
Patrón 2	Cbf_cb(MODE_INTRA)		fixed_0: condLfalse fixed_1: condLtrue	Cbf_cb[xL][yL]	Fijado
	Cbf_cr(MODE_INTRA)			Cbf_cr[xL][yL]	
	Cbf_luma			Cbf_luma[xL][yL]	
Patrón 3	Cbf_cb(MODE_INTRA)		Fijado	Fijado	Fijado
	Cbf_cr(MODE_INTRA)				
	Cbf_luma				

FIG. 28A

Patrón 2: usando solo bloque izquierdo (cbf_luma, cbf_cr, cbf_cb)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,01	0,05	0,02	-0,01	0,45	-0,26	-0,06	0,16	0,18	-0,08	0,26	0,47
Clase B	0,02	0,05	0,10	-0,03	-0,05	0,08	-0,07	-0,02	-0,15	0,00	0,06	0,31
Clase C	0,02	0,12	0,06	-0,11	0,39	-0,03	-0,01	0,58	-0,13	0,03	0,19	-0,43
Clase D	0,02	-0,03	0,06	0,14	-0,52	0,04	-0,04	-0,52	-1,42	0,00	-0,12	-1,21
Clase E	0,03	0,13	0,18									
Todo	0,02	0,06	0,08	-0,00	-0,05	0,01	-0,04	0,09	-0,28	-0,02	0,12	-0,11
Tiempo de cod [%]	100 %			99 %			99 %			99 %		
Tiempo de dec [%]	101 %			103 %			103 %			103 %		

FIG. 28B

Patrón 3: no usando ni bloque izquierdo ni bloque superior (cbf_luma, cbf_cr, cbf_cb)

	Intra			Acceso aleatorio			Retardo bajo			Retardo bajo (P)		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,04	0,06	0,15	-0,04	0,37	-0,13	0,06	0,46	0,20	0,04	0,38	0,32
Clase B	0,07	0,10	0,15	-0,00	0,23	-0,06	-0,06	0,23	0,35	0,06	0,24	0,53
Clase C	0,03	0,13	0,20	-0,09	0,59	0,53	-0,09	0,90	-0,79	0,03	0,17	-0,77
Clase D	-0,00	0,07	0,19	0,12	0,16	0,19	-0,02	-0,13	-1,29	0,05	-0,45	-0,42
Clase E	0,12	0,14	0,26									
Todo	0,05	0,10	0,19	0,01	0,30	0,15	-0,02	0,40	-0,29	0,04	0,14	-0,04
Tiempo de cod [%]	100 %			99 %			99 %			99 %		
Tiempo de dec [%]	101 %			103 %			103 %			103 %		

FIG. 29A

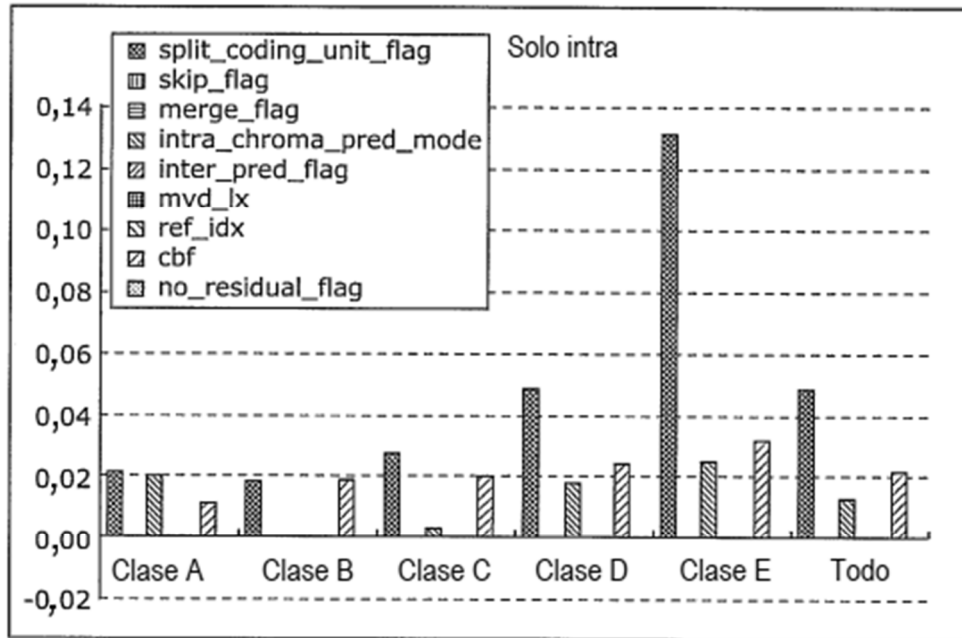


FIG. 29B

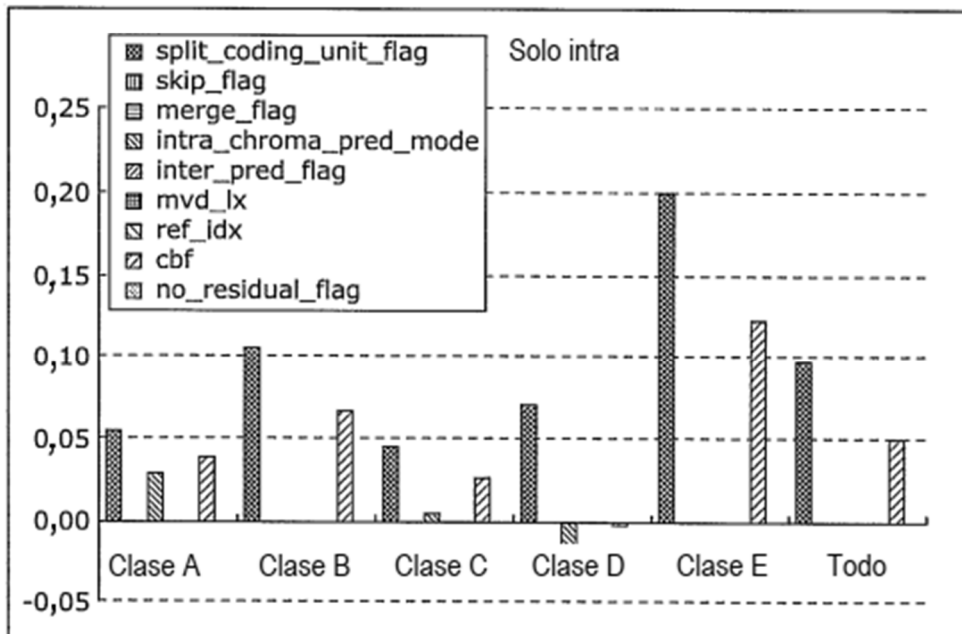


FIG. 30A

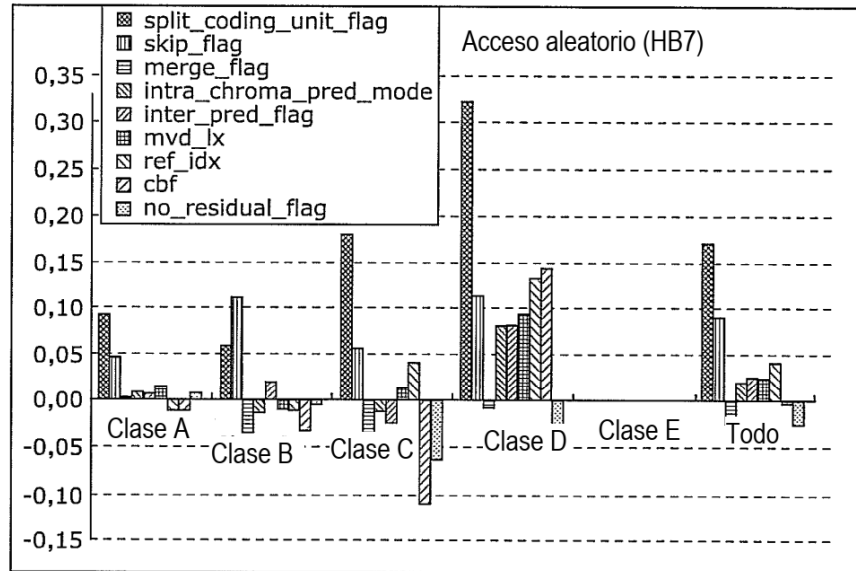


FIG. 30B

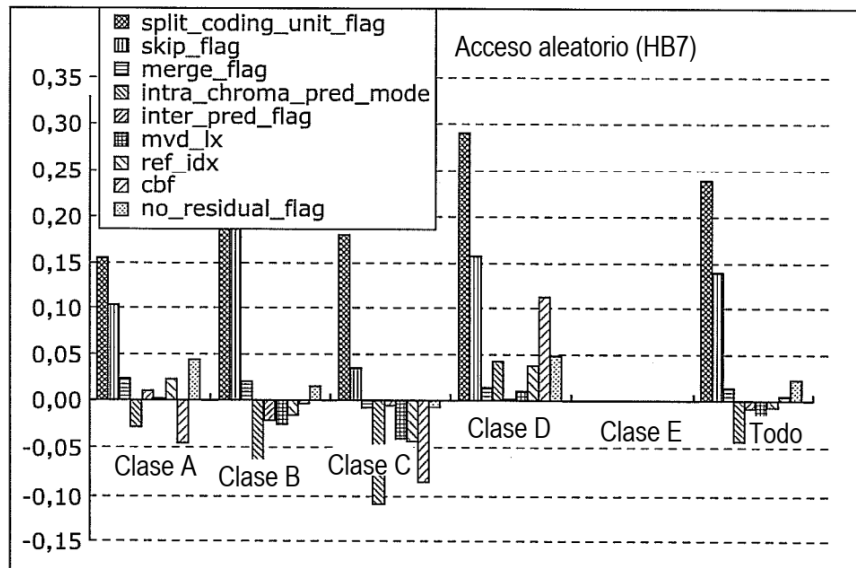


FIG. 31A

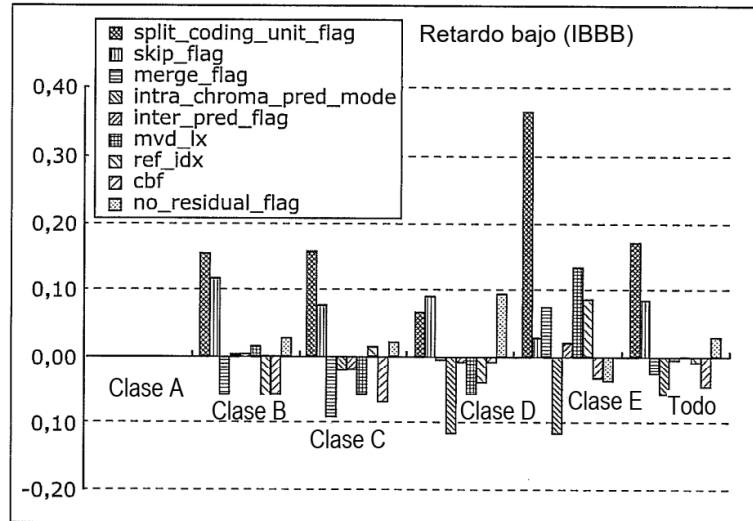


FIG. 31B

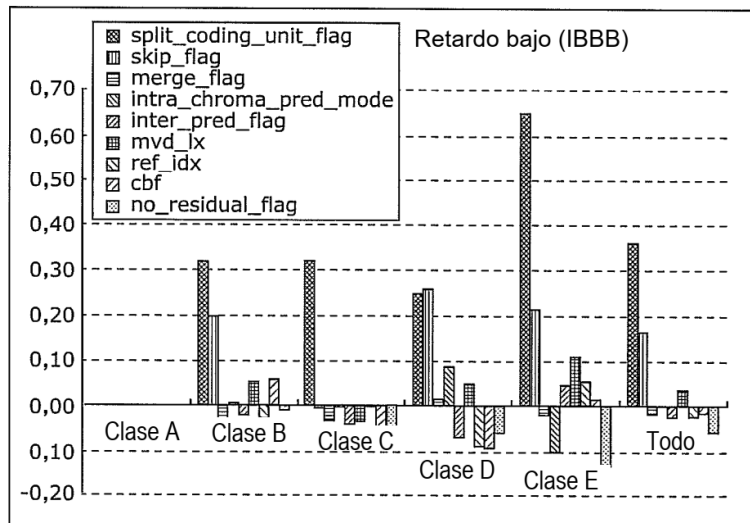


FIG. 32A

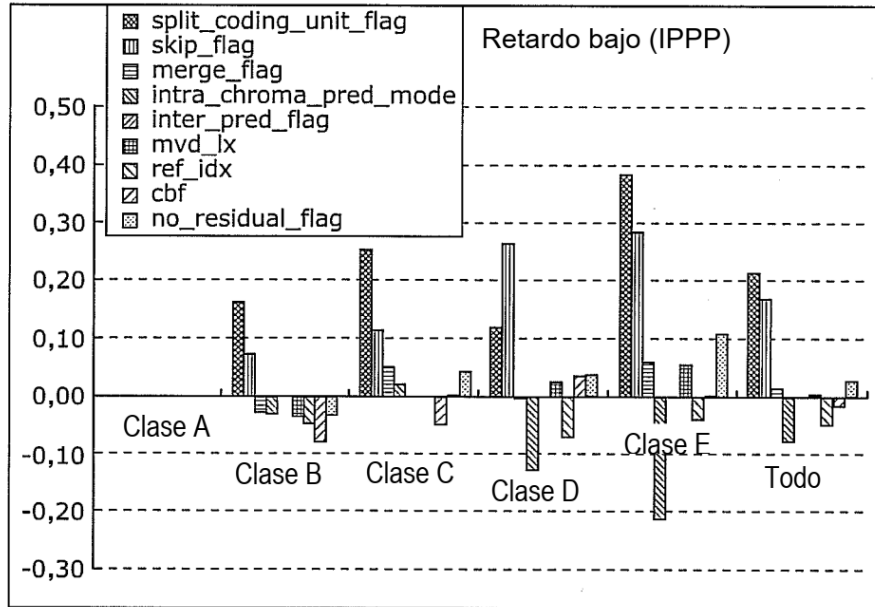


FIG. 32B

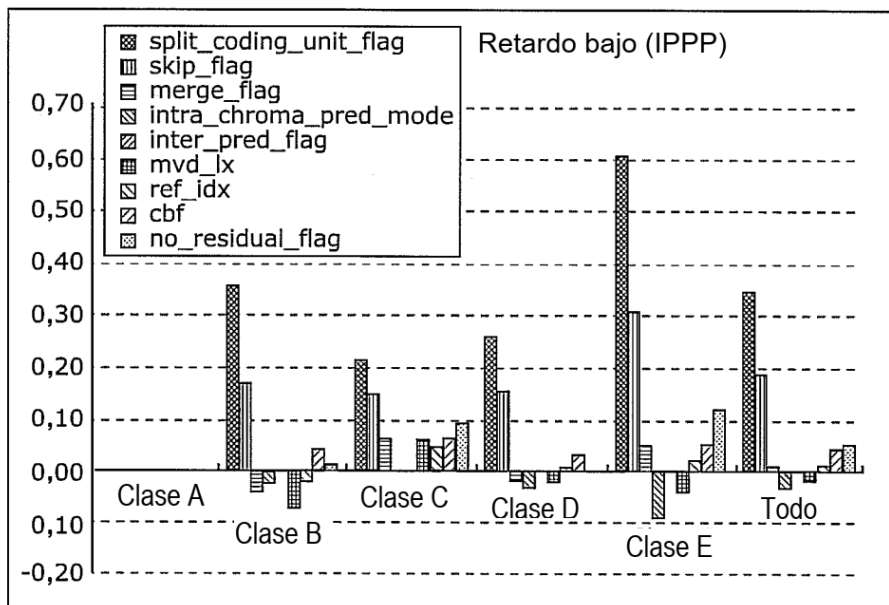


FIG. 33A

Parámetro	Superior Izquierdo
skip	Si
split	Si
merge	No
intra-chroma	No
inter_pred	No
ref_idx	No
mvd	No
noRes	No
cbf	No

FIG. 33B

	Intra			Acceso aleatorio			Retardo bajo IBBB			Retardo bajo IPPP		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,06	0,18	0,22	-0,01	0,36	0,59						
Clase B	0,08	0,14	0,21	0,05	0,29	0,33	0,01	1,05	0,36	0,07	0,89	0,24
Clase C	0,04	0,18	0,25	-0,02	0,34	0,17	-0,11	0,49	0,37	0,00	0,56	0,37
Clase D	0,02	0,35	0,34	0,00	-0,03	0,02	-0,05	0,56	0,32	0,04	0,34	0,09
Clase E	0,06	0,40	0,39				0,06	2,11	-0,59	0,13	1,61	-0,70
Todo	0,05	0,24	0,27	0,01	0,24	0,30	-0,03	0,99	0,18	0,05	0,80	0,06
Tiempo de cod[%]	100%			100%			100%			100%		
Tiempo de dec[%]	99%			99%			99%			98%		

FIG. 34A

Parámetro	Superior	Izquierdo
skip	Si	Si
split	Si	Si
merge	No	No
intra-chroma	No	Si
inter_pred	No	No
ref_idx	No	No
mvd	No	No
noRes	No	No
cbf	No	Si

FIG. 34B

	Intra			Acceso aleatorio			Retardo bajo IBPB			Retardo bajo IPPP		
	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD	Y tasa BD	U tasa BD	V tasa BD
Clase A	0,02	-0,02	0,14	-0,04	0,22	-0,01						
Clase B	0,03	0,15	0,1	-0,03	0,06	-0,18	0,01	0,3	0,4	0,08	0,39	0,64
Clase C	0,03	0,13	0,08	-0,02	0,32	-0,05	-0,07	0,64	0,36	0,04	0,62	0,5
Clase D	0,04	0,03	0,15	0,06	-0,27	-0,49	-0,11	0,89	-0,57	0,05	0,71	-0,68
Clase E	0,07	0,16	0,22				0	1,62	-1,45	0,28	1,96	-1,06
Todo	0,04	0,09	0,13	-0,01	0,15	-0,18	-0,04	0,78	-0,2	0,1	0,82	-0,05
Tiempo de cod [%]			103%			103%			103%			100%
Tiempo de dec [%]			100%			101%			102%			

FIG. 35

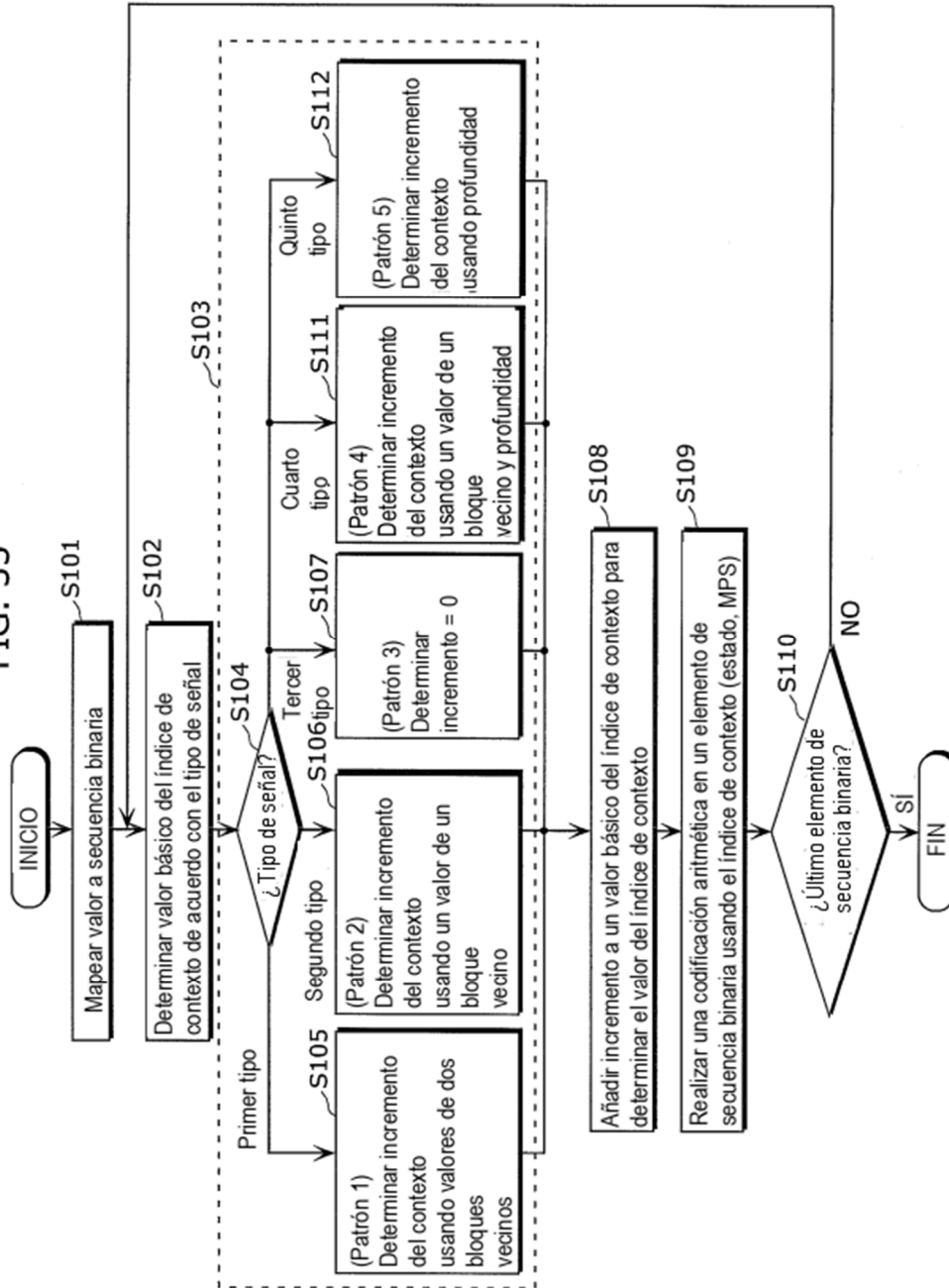


FIG. 36

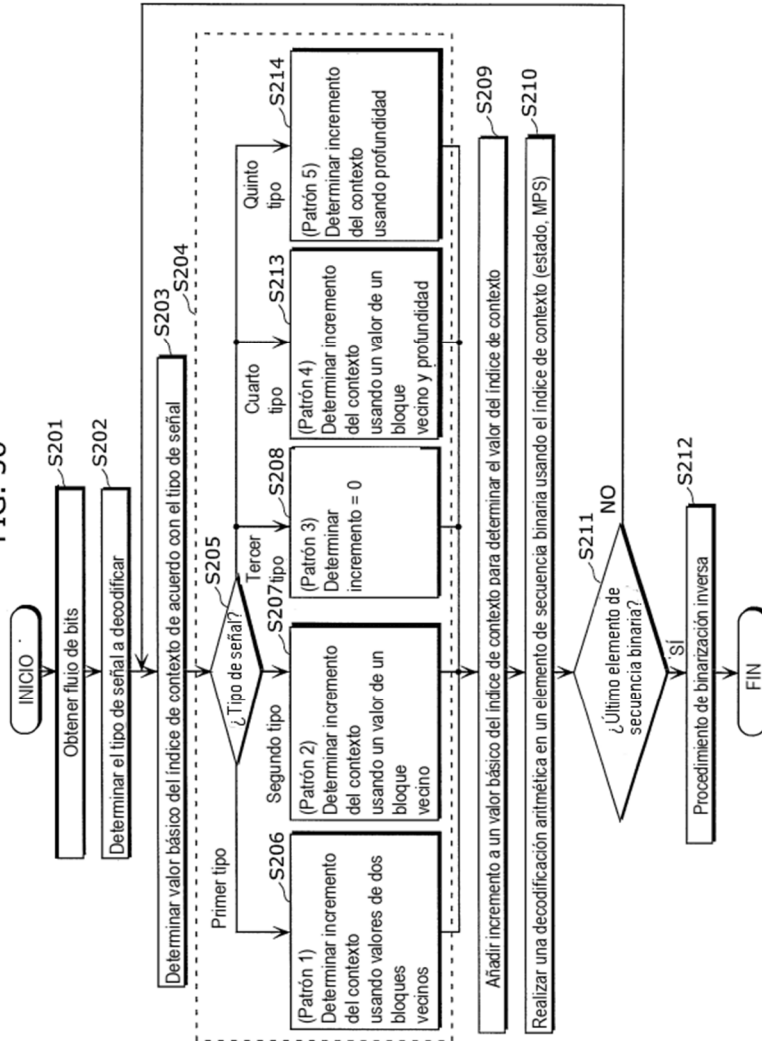


FIG. 37

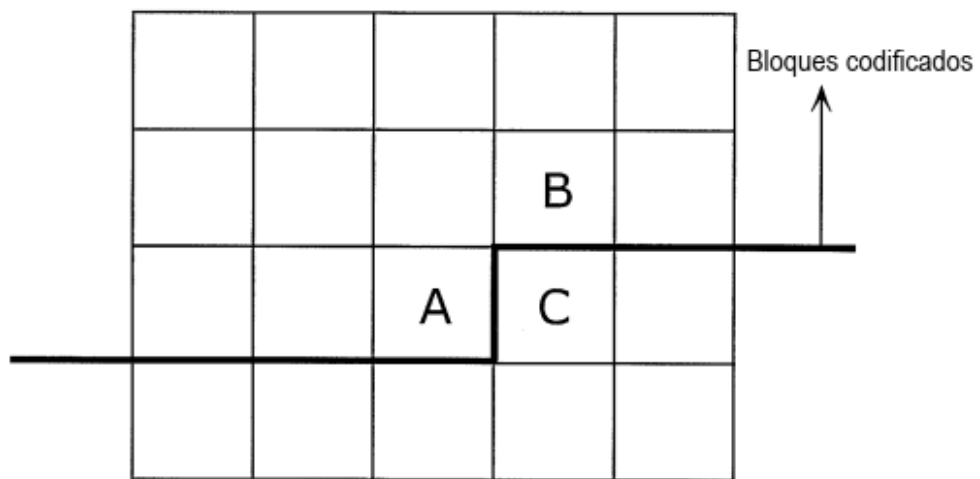
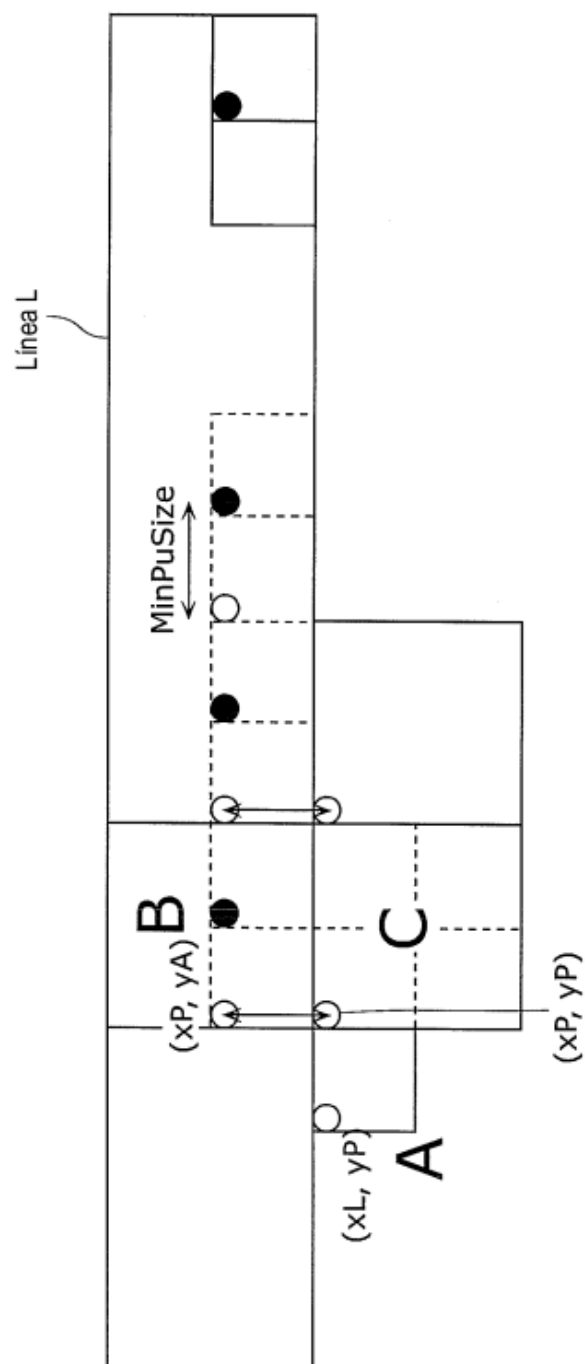


FIG. 38



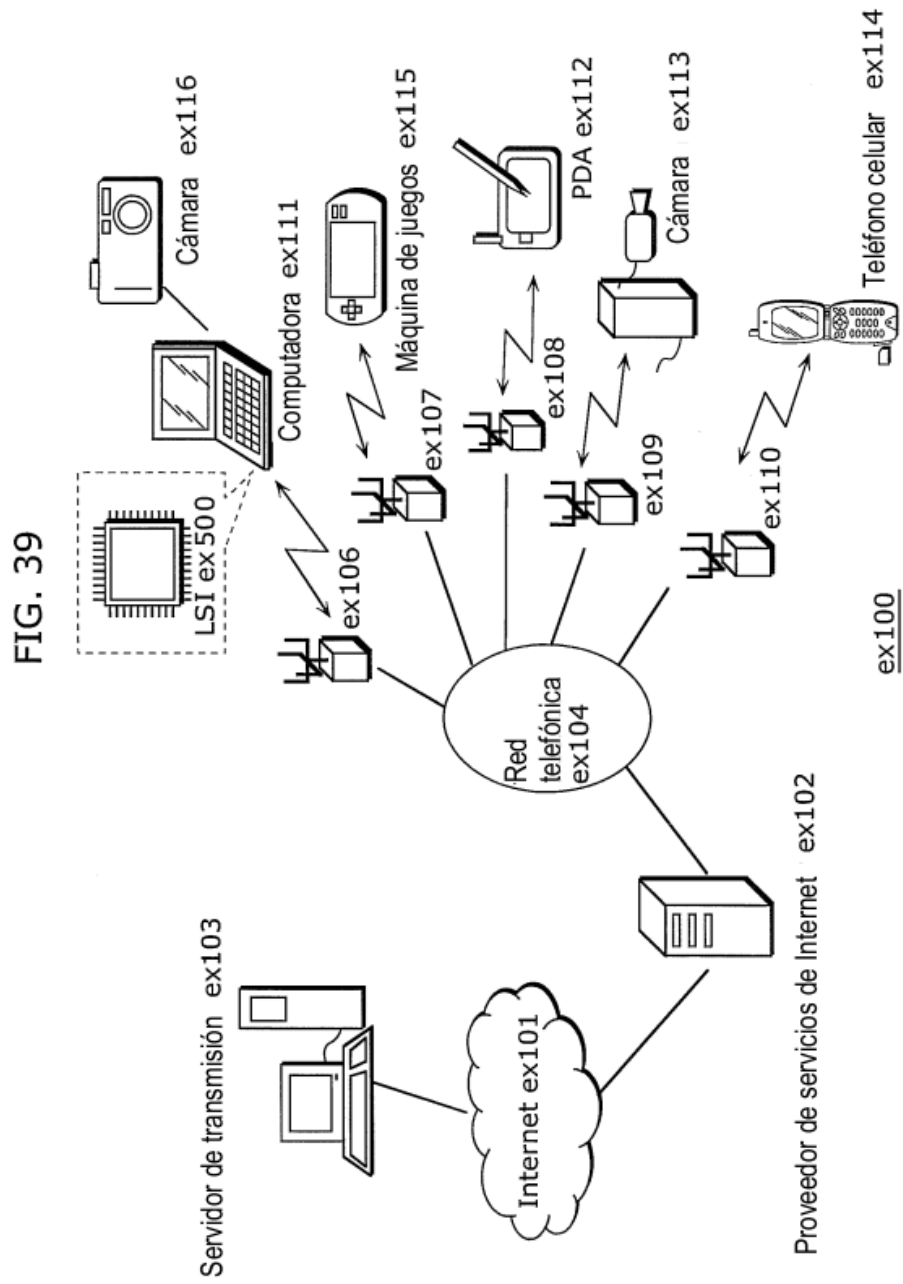


FIG. 40

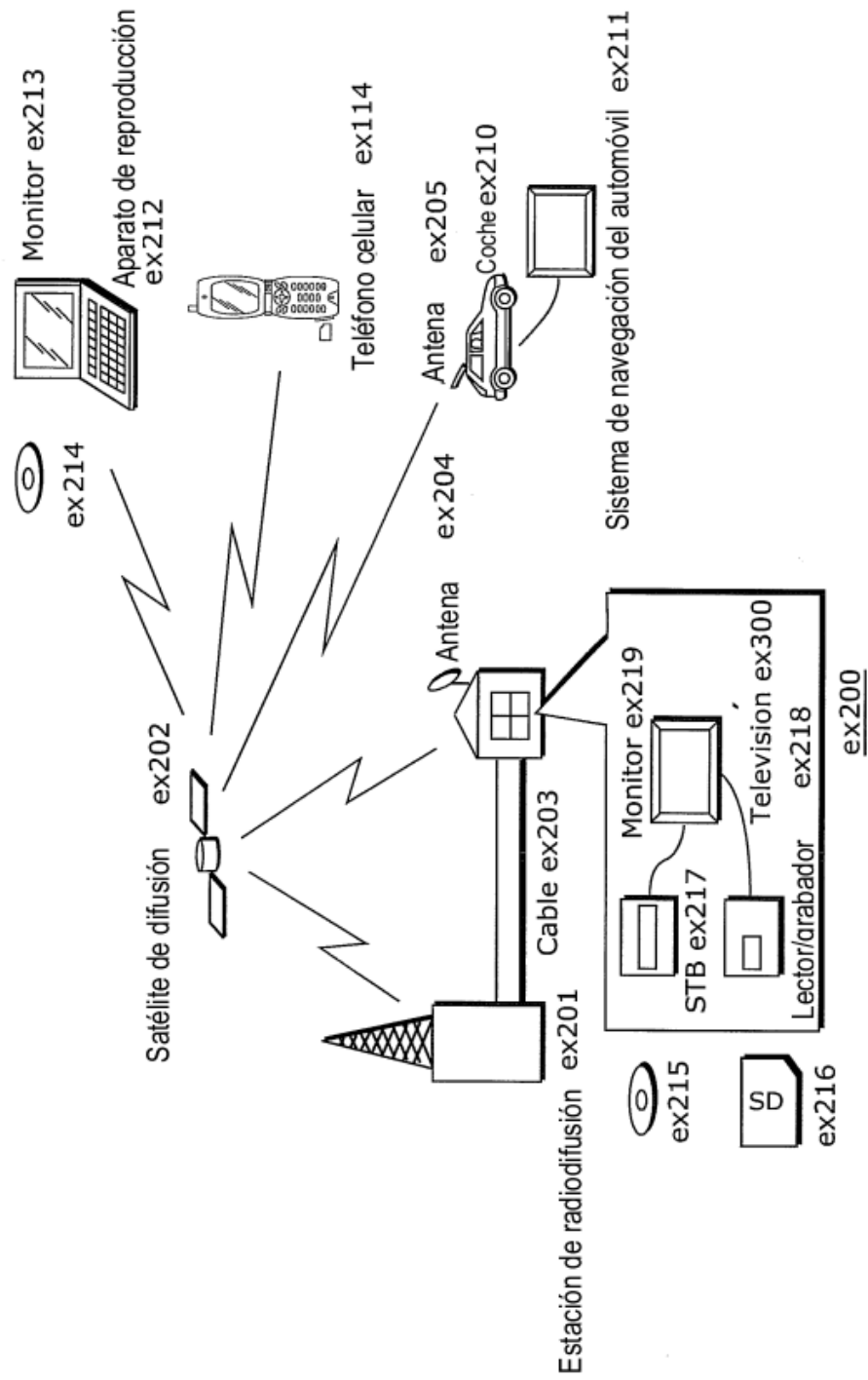


FIG. 41

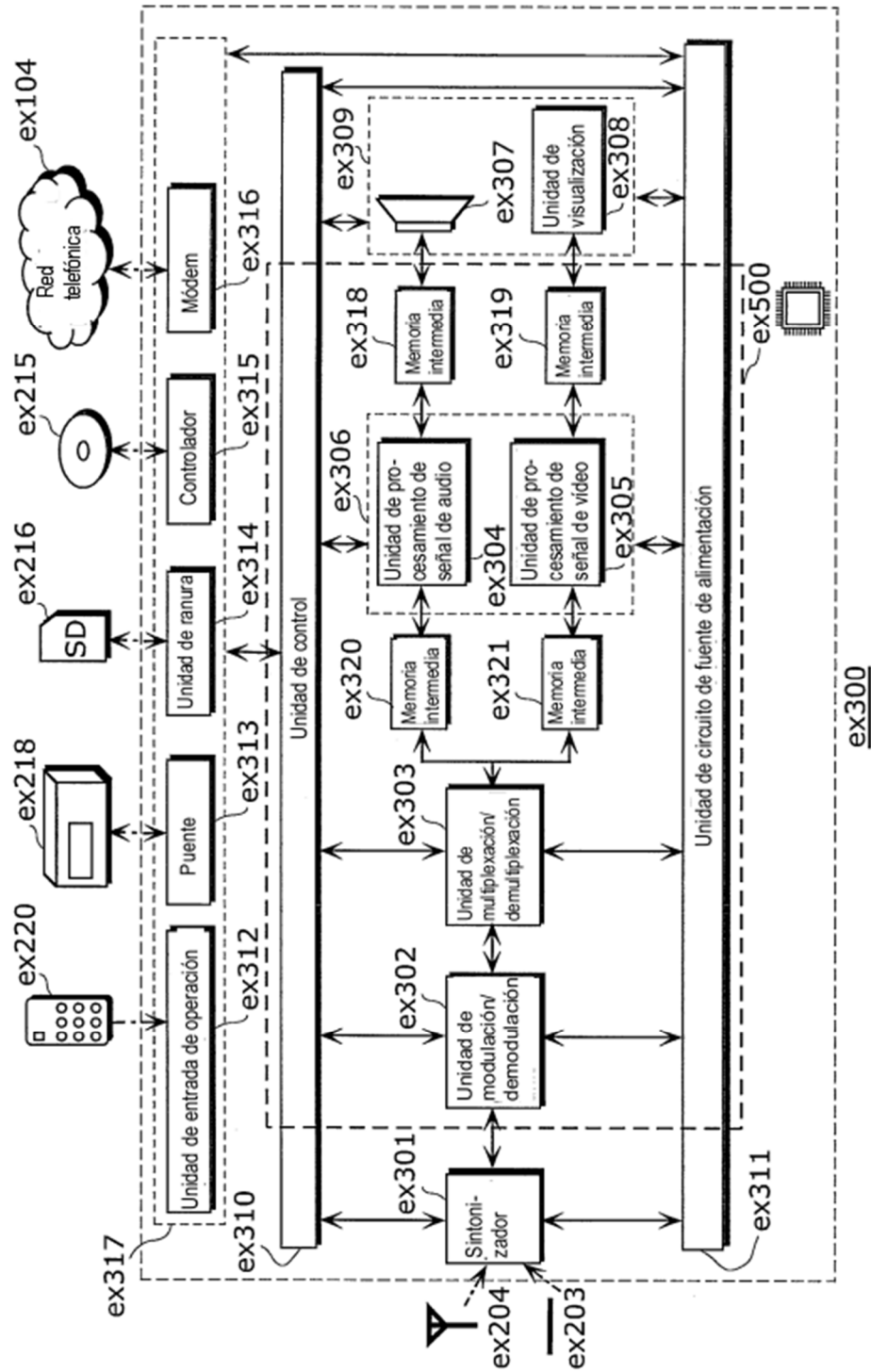


FIG. 42

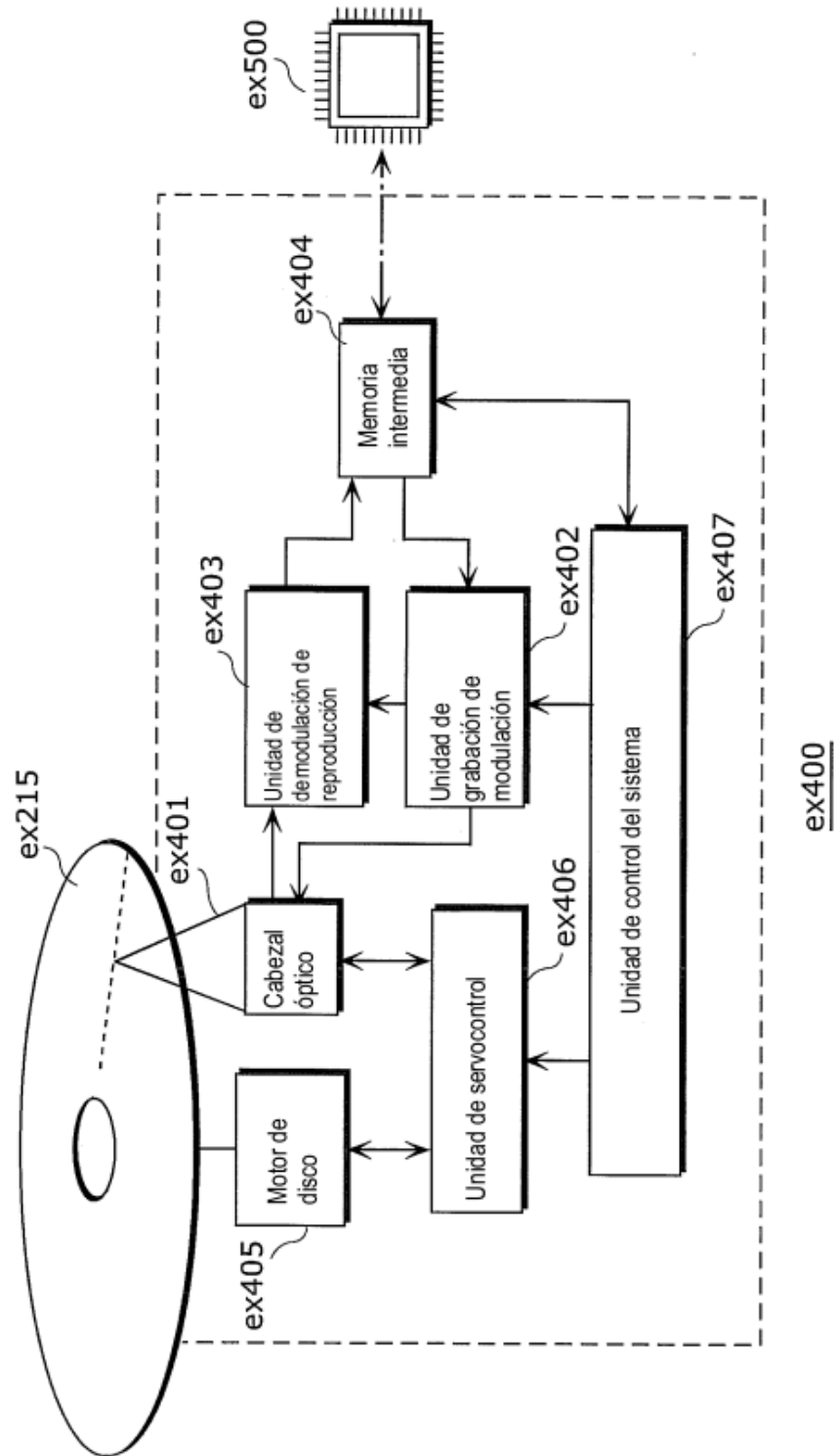


FIG. 43

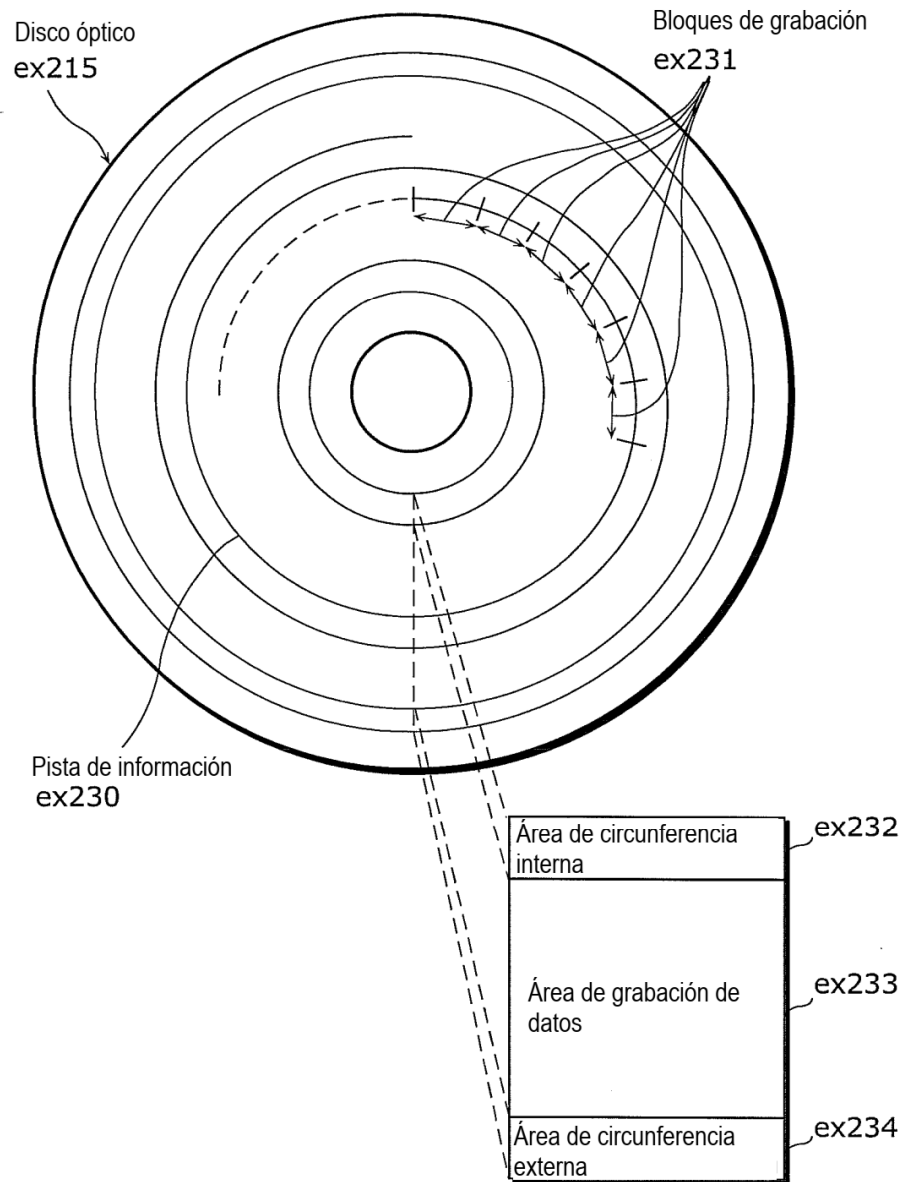


FIG. 44A



FIG. 44B

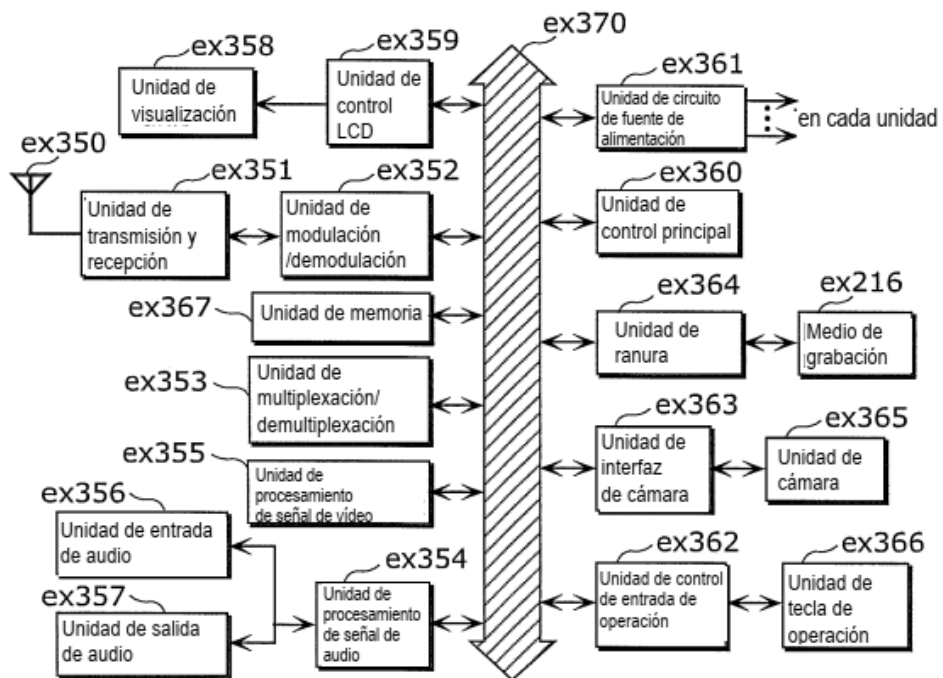


FIG. 45

Flujo de vídeo (PID=0x1011, vídeo primario)
Flujo de audio (PID=0x1100)
Flujo de audio (PID=0x1101)
Flujo de gráficos de presentación (PID=0x1200)
Flujo de gráficos de presentación (PID=0x1201)
Flujo de gráficos interactivo (PID=0x1400)
Flujo de vídeo (PID=0x1B00, vídeo secundario)
Flujo de vídeo (PID=0x1B01, vídeo secundario)

FIG. 46

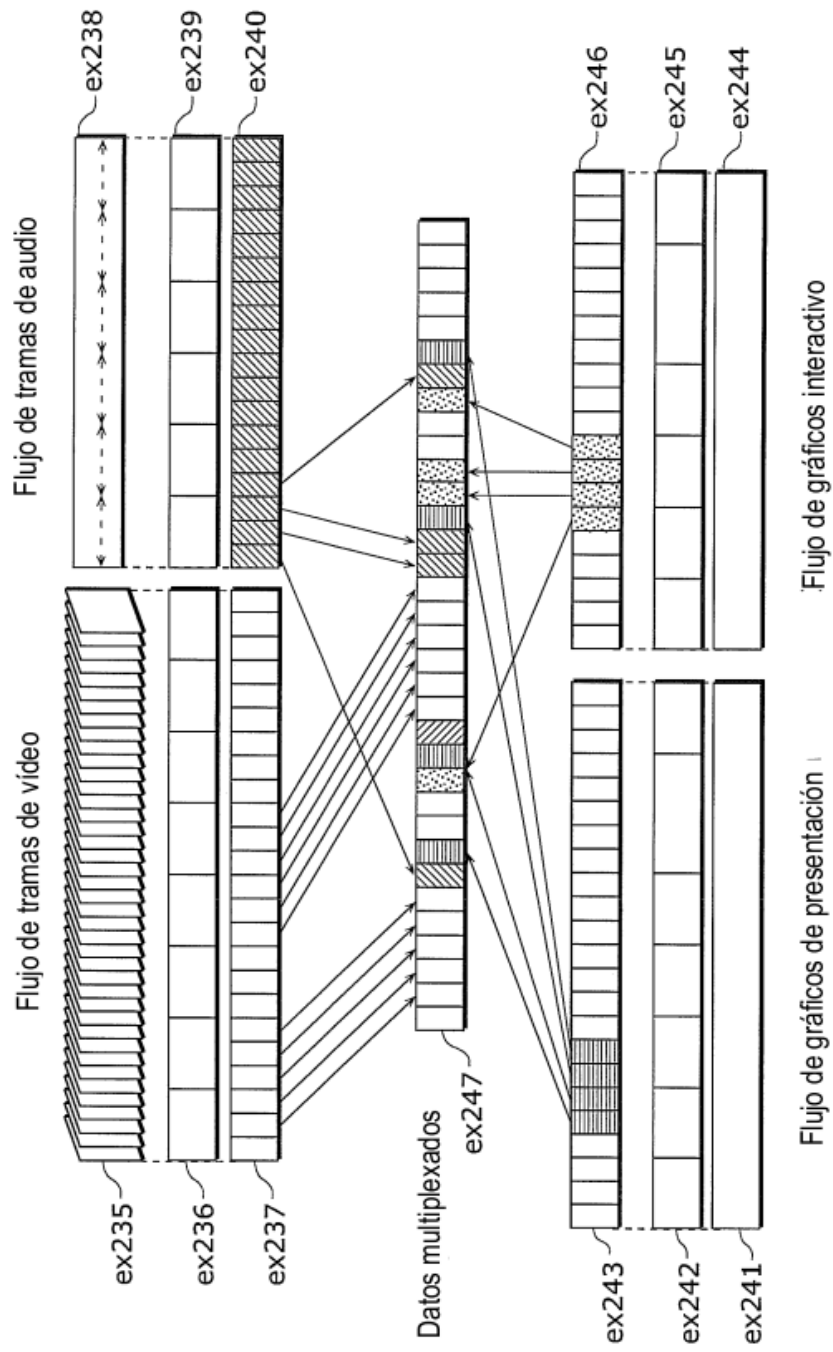


FIG. 47

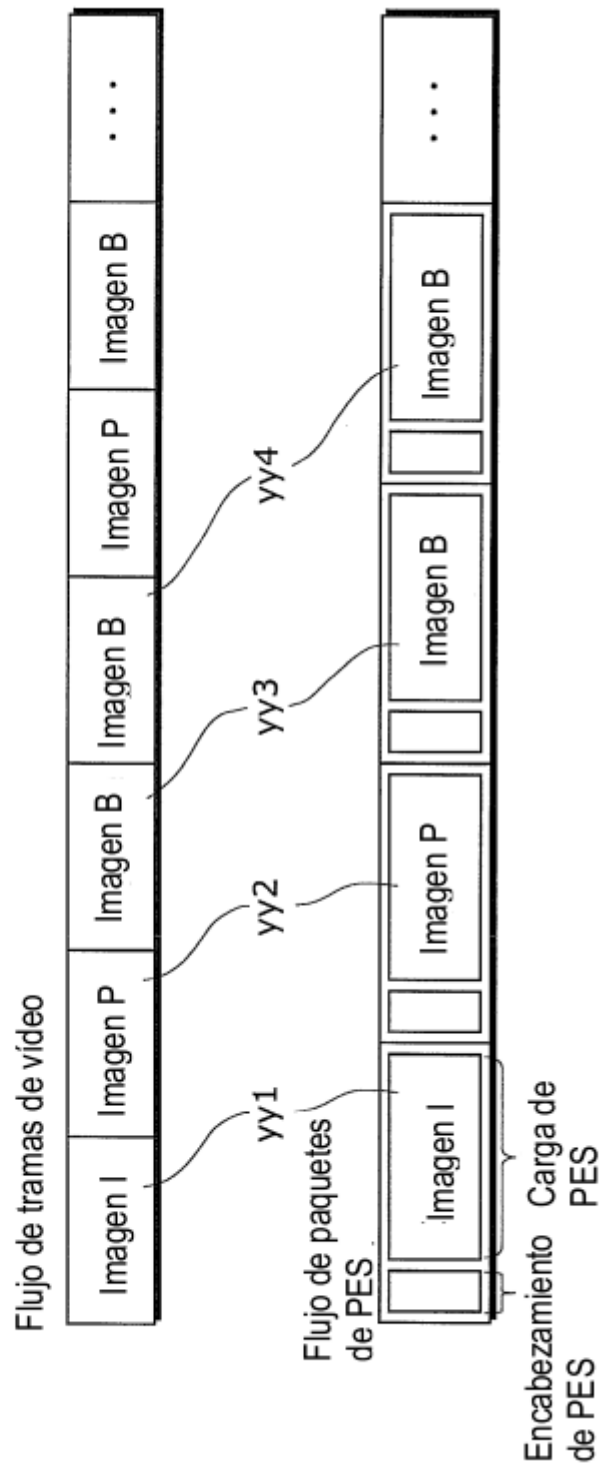
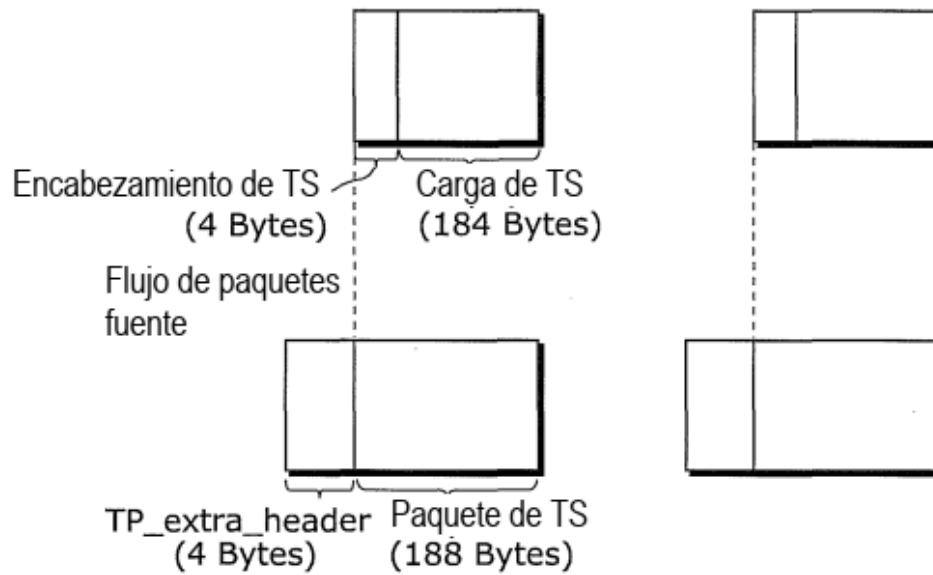


FIG. 48

Flujo de paquetes de TS



Datos multiplexados

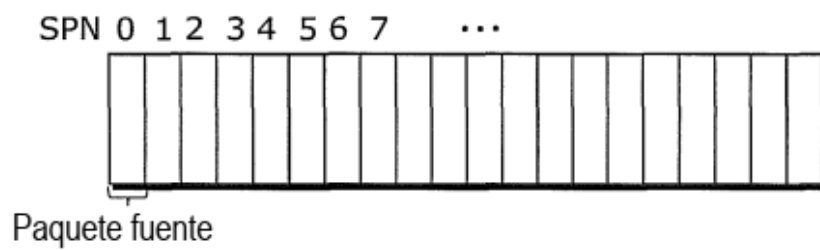


FIG. 49

Estructura de datos de PMT

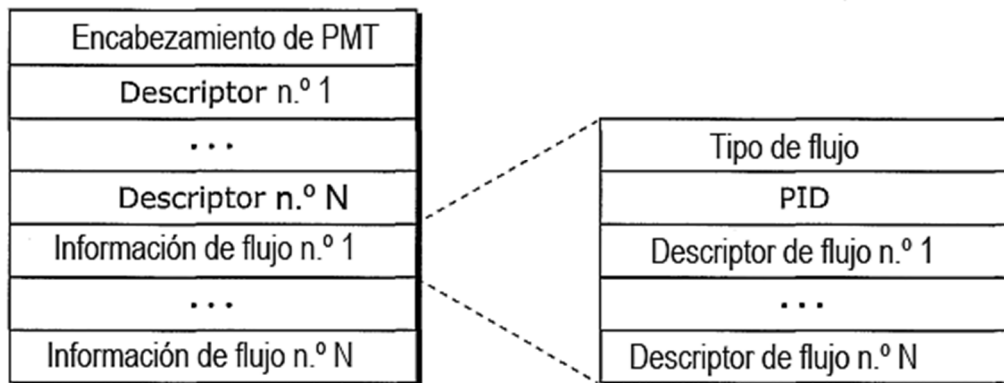


FIG. 50

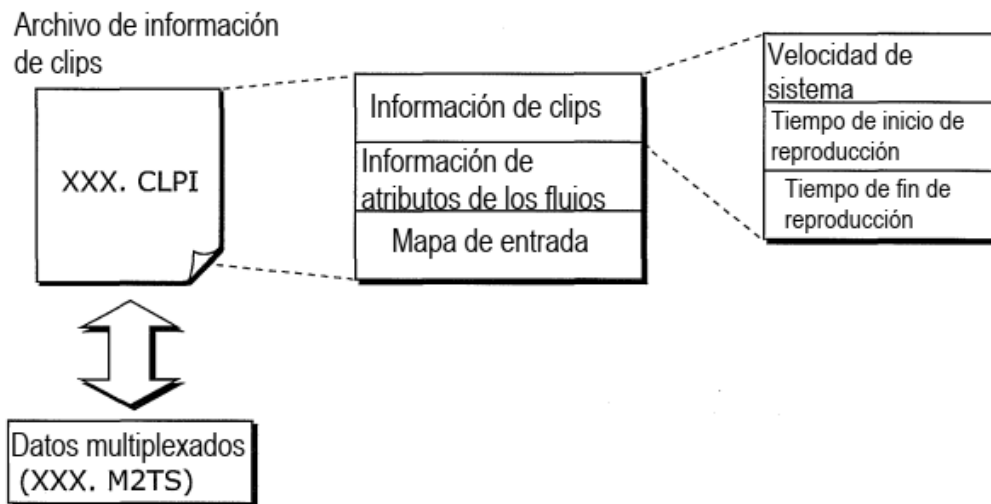


FIG. 51

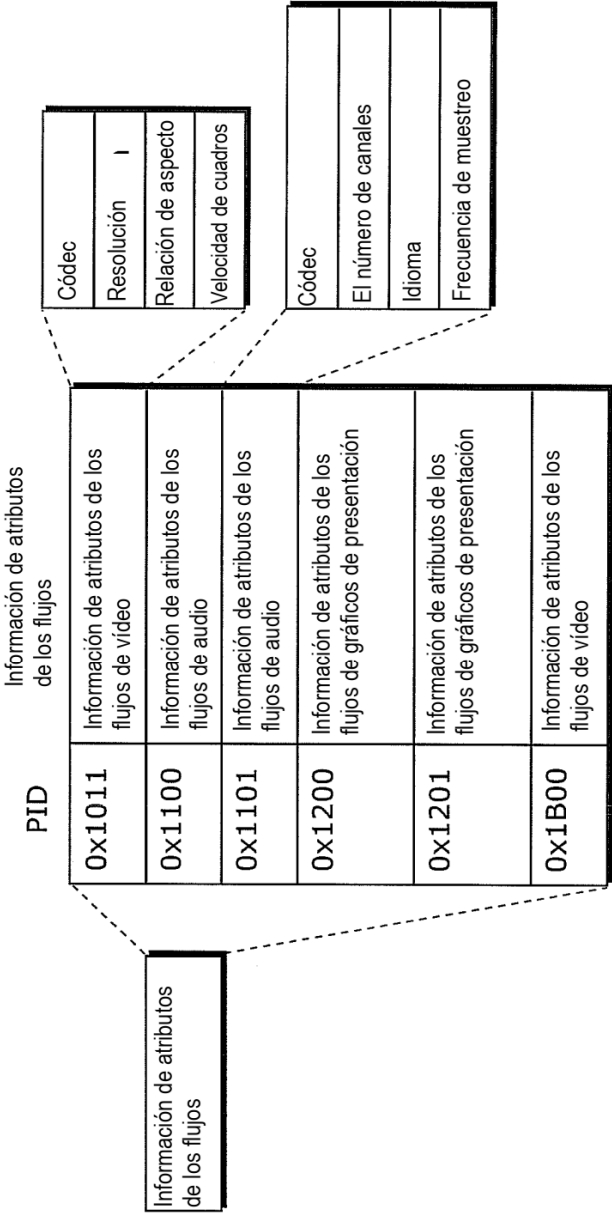


FIG. 52

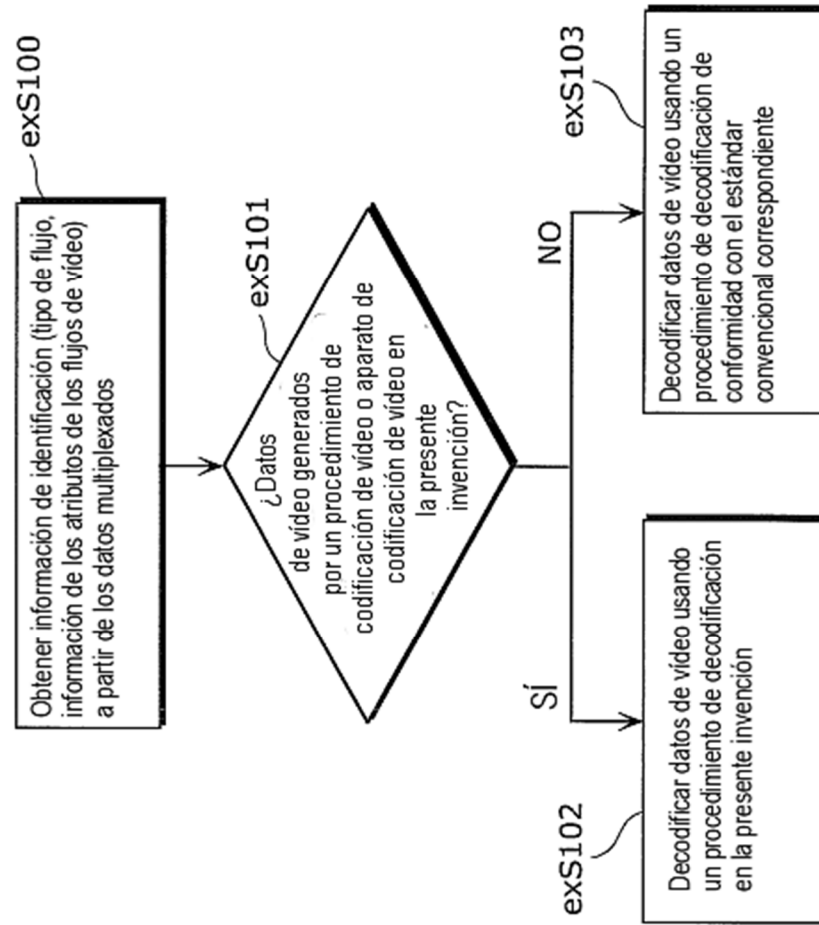


FIG. 53

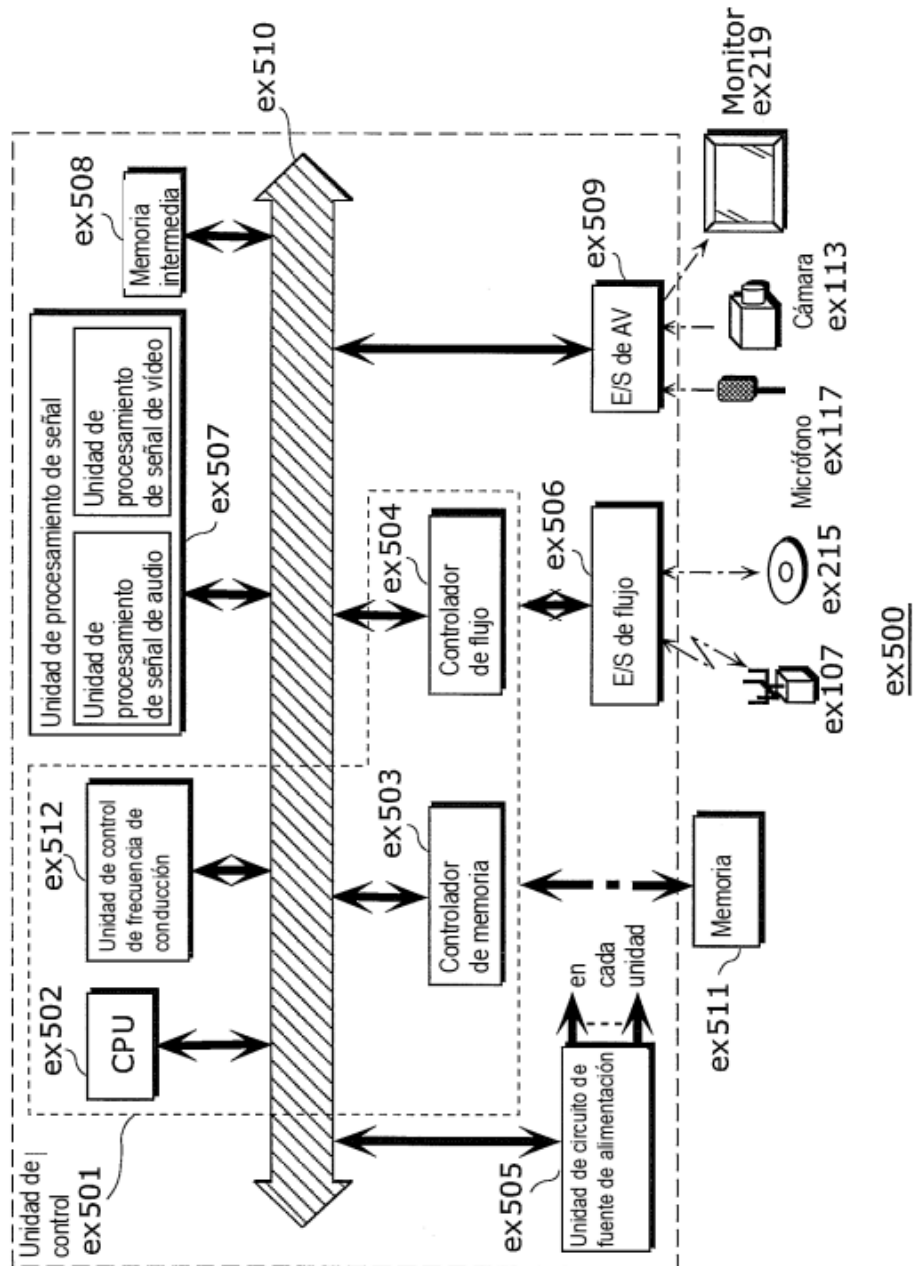


FIG. 54

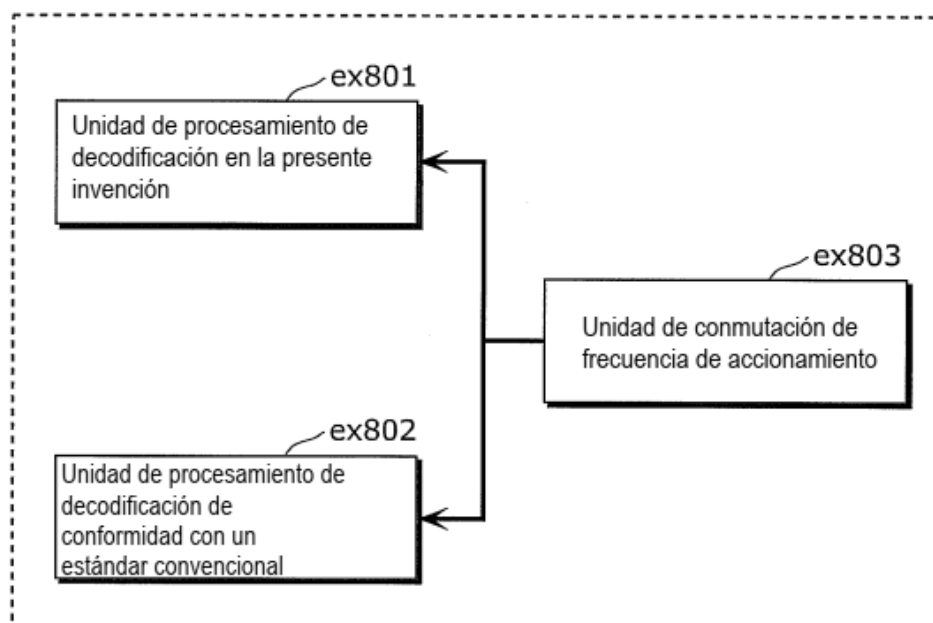


FIG. 55

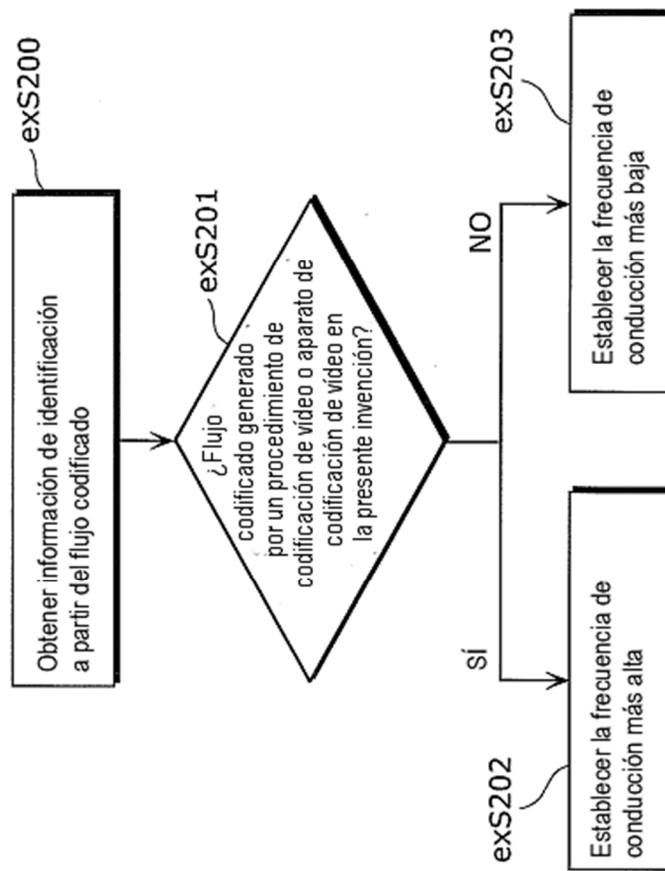


FIG. 56

Estándar correspondiente	Frecuencia de conducción
MPEG-4 AVC	500 MHz
MPEG-2	350 MHz
⋮	⋮

FIG. 57A

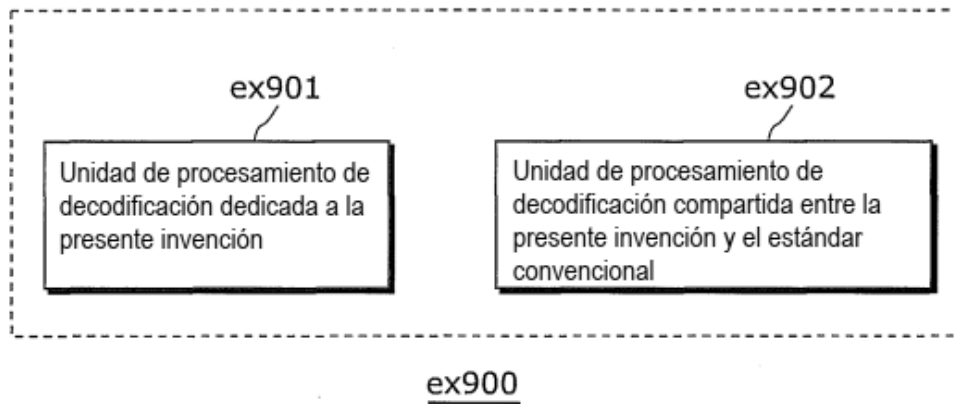


FIG. 57B

