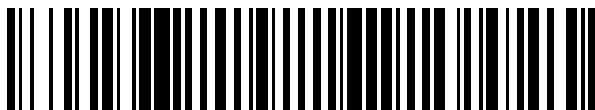


(19)



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS
ESPAÑA



(11) Número de publicación: **2 924 028**

(51) Int. Cl.:

H04N 19/119 (2014.01)
H04N 19/124 (2014.01)
H04N 19/136 (2014.01)
H04N 19/176 (2014.01)
H04N 19/463 (2014.01)
H04N 19/96 (2014.01)

(12)

TRADUCCIÓN DE PATENTE EUROPEA

T3

(96) Fecha de presentación y número de la solicitud europea: **12.01.2012 E 21176055 (8)**

(97) Fecha y número de publicación de la concesión europea: **20.07.2022 EP 3905680**

(54) Título: **Aparato de codificación de imagen, procedimiento de codificación de imagen y programa, y aparato de decodificación de imagen, procedimiento de decodificación de imagen y programa**

(30) Prioridad:

13.01.2011 JP 2011004648

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

04.10.2022

(73) Titular/es:

CANON KABUSHIKI KAISHA (100.0%)
30-2 Shimomaruko 3-chome
Ohta-ku, Tokyo 146-8501, JP

(72) Inventor/es:

KOBAYASHI, MASAAKI

(74) Agente/Representante:

DURAN-CORRETJER, S.L.P

ES 2 924 028 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

5 Aparato de codificación de imagen, procedimiento de codificación de imagen y programa, y aparato de decodificación de imagen, procedimiento de decodificación de imagen y programa

10 **[Sector técnico]**

15 La presente invención hace referencia a un aparato de codificación de imagen, un procedimiento de codificación de imagen y un programa, y a un aparato de decodificación de imagen, un procedimiento de decodificación de imagen y un programa. En concreto, la presente invención hace referencia a un procedimiento de codificación de parámetros de calidad de la imagen y a un procedimiento de decodificación perteneciente al cálculo de un parámetro de calidad de la imagen y de introducción de código en la codificación de imagen.

20 **[Estado de la técnica anterior]**

25 H.264/Codificación de video avanzada (AVC, Advanced Video Coding) del Grupo de expertos en imagen en movimiento (MPEG)-4, de aquí en adelante conocido como H.264, se conoce como un procedimiento de grabación por compresión de una imagen en movimiento. H.264 es ampliamente utilizado en la difusión digital terrestre de un segmento y otras aplicaciones. H.264 está caracterizado por proporcionar una pluralidad de capacidades de intrapredicción, utilizando transformaciones de enteros en unidades de 4 x 4 píxeles, además de esquemas de codificación convencionales (ver ISO/IEC 14496-10: 2004 Tecnología de la información - Codificación de objetos audiovisuales Parte 10: Codificación de video avanzada, codificación de video avanzada ITU-T H.264 para servicios audiovisuales genéricos).

30 35 Los esquemas de codificación que utilizan una transformación ortogonal y cuantificación, tipificados mediante MPEG-2 y H.264, llevan a cabo una transformación ortogonal y una cuantificación en imágenes de bloques predeterminados para generar datos de coeficiente de cuantificación. Para el control de la calidad de la imagen, la cuantificación se lleva a cabo utilizando un parámetro de control de la calidad de la imagen llamado parámetro de cuantificación. Específicamente, la cuantificación utilizando un valor pequeño del parámetro de cuantificación produce una calidad de la imagen mejorada, pero con una gran cantidad de códigos. La cuantificación con un valor grande del parámetro de cuantificación produce una baja calidad de la imagen con una cantidad reducida de códigos. Se seleccionan valores óptimos de los parámetros de cuantificación para la codificación según una cantidad objetivo de códigos. Dicho control se conoce como control de velocidad. Se han explicado diversos procedimientos de control de velocidad, incluido TM5. Después de la cuantificación, los datos del coeficiente de cuantificación son sometidos a codificación de longitud variable para generar datos de coeficientes codificados.

40 45 Los parámetros de cuantificación también están codificados para generar código de parámetro de cuantificación. Por ejemplo, H.264 utiliza un valor de diferencia entre un parámetro de cuantificación que se utiliza para cuantificar un bloque que precede a un bloque objetivo y un parámetro de cuantificación que se utiliza para cuantificar el bloque objetivo. El valor de diferencia se conoce como delta del parámetro de cuantificación (QP_DELTA). El QP_DELTA calculado se codifica en el código de QP_DELTA mediante codificación de Golomb o codificación aritmética. Por ejemplo, el códec de código de QP_DELTA mediante codificación de Golomb produce un bit de código incluso si el valor de diferencia es 0. Los datos codificados del coeficiente y el código del parámetro de cuantificación generado de este modo son transmitidos a un decodificador. El decodificador decodifica los datos codificados del coeficiente y el código del parámetro de cuantificación para generar datos de coeficiente de cuantificación y parámetros de cuantificación. A continuación, el decodificador lleva a cabo una cuantificación inversa y una transformación ortogonal inversa en los datos del coeficiente de cuantificación utilizando los parámetros de cuantificación, generando de este modo una imagen decodificada.

50 55 Una sola unidad de procesamiento en MPEG-2 y H.264 se denomina macrobloque. Los macrobloques son bloques de retícula de 16 x 16 píxeles en los que una imagen está dividida en un patrón de retícula. En términos de píxeles, los bloques que se transformarán ortogonalmente en MPEG-2 tienen un tamaño de 8 x 8 píxeles, y en H.264, un tamaño de 8 x 8 píxeles o de 4 x 4 píxeles. Es decir, un solo macrobloque incluye una pluralidad de bloques de transformación ortogonal. MPEG-2 y H.264 pueden controlar los parámetros de cuantificación (control de velocidad) en unidades de macrobloques. Los bloques de transformación ortogonal incluidos en el mismo macro bloque se cuantifican con el mismo parámetro de cuantificación. Según una tecnología mejorada de H.264, una imagen se divide en un patrón de retícula en bloques de retícula llamados bloques de árbol de codificación más grandes (LCTB, Largest Coding Tree Blocks). Un LCTB tiene un tamaño de 64 x 64 píxeles. Los LCTB se dividen en bloques de tamaños más pequeños llamados bloques de árbol de codificación (CTB, Coding Tree Block) mediante la utilización de una estructura de árbol cuaternario de región. Para buscar o introducir/exportar datos estructurados en árbol, se debe especificar el orden de escaneo. Un CTB incluye bloques de transformación ortogonal llamados unidades de transformación (TU, Transform Unit). Las TU también se pueden dividir en tamaños más pequeños mediante la utilización de una estructura de árbol cuaternario de región. Cada unidad tiene un indicador de división. Un bloque que tiene un indicador de división de verdadero está estructurado para incluir cuatro bloques de división que tienen un tamaño de 1/2 en altura y anchura. Un bloque que tiene un indicador de división de falso no incluye ningún bloque de división y en su lugar contiene datos reales en el bloque. En otras palabras, solo las TU no divididas

5 contienen datos de coeficientes en bloques de imagen. Un LCTB (bloque de retícula) incluye una pluralidad de bloques estructurados jerárquicamente de diferentes tamaños de transformaciones ortogonales. El dividir o no un bloque se puede determinar por diversos procedimientos. Uno de los procedimientos de determinación se explica en la Solicitud de Patente japonesa abierta a inspección pública Nº 2005-191706, que incluye la utilización de multiplicadores de Lagrange para calcular costes de bloques y seleccionar un procedimiento de división de bloques de menor coste.

10 Supóngase que un procedimiento de codificación de imagen incluye codificar una imagen en unidades de bloques de retícula en los que la imagen se divide en un patrón de retícula. Si los bloques de retícula son de gran tamaño y el control de la calidad de la imagen se lleva a cabo en unidades de bloques de retícula, los bordes entre bloques de diferentes calidades de imagen pueden resultar apreciables.

15 Los parámetros de calidad de la imagen se codifican y transmiten independientemente de si el parámetro de calidad de la imagen cambia. Si los bloques de retícula se subdividen en unidades más pequeñas para el control de la calidad de la imagen, existe un problema de sobrecarga causado por el código del parámetro de calidad de la imagen.

[Características de la invención]

20 La presente invención está dirigida a un aparato de codificación de imagen que permite el control de la calidad de la imagen en unidades de bloques apropiados y puede hacer que los bordes de bloque sean más pequeños para generar una imagen en la que los bordes son menos apreciables. La invención se expone en el conjunto de reivindicaciones adjunto. Otros aspectos y realizaciones en la descripción se dan a conocer solo como ejemplos ilustrativos.

25 Según un aspecto de la presente invención, un aparato de codificación de imagen configurado para dividir una imagen en bloques de división de una pluralidad de tamaños y para llevar a cabo codificaciones en la imagen mientras se controla la calidad de la imagen según un valor de parámetro en unidades de bloques de división, incluye una unidad configurada para obtener un tamaño de bloque de un bloque objetivo a codificar, una unidad configurada para obtener un tamaño de bloque mínimo utilizado para controlar el valor del parámetro, una unidad configurada para obtener un estado de división del bloque objetivo, una unidad configurada para obtener el valor del parámetro, una unidad configurada para determinar si el bloque objetivo se divide según el estado de división del bloque objetivo, una unidad configurada para determinar si el tamaño de bloque del bloque objetivo es mayor o igual que el tamaño de bloque mínimo, una unidad configurada para determinar si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo y una unidad configurada para codificar el valor del parámetro obtenido.

30 35 40 Según una realización a modo de ejemplo de la presente invención, es posible controlar la calidad de la imagen en unidades de bloques apropiados y hacer que los bordes de los bloques sean más pequeños para generar una imagen en la que los bordes son menos apreciables. Puesto que no es necesario introducir el código del parámetro de control de la calidad de la imagen en todos los bloques transformados ortogonalmente, la cantidad de código del parámetro de cuantificación no aumentará más de lo necesario.

45 Otras características y aspectos de la presente invención resultarán evidentes a partir de la siguiente descripción detallada de realizaciones a modo de ejemplo, haciendo referencia a los dibujos adjuntos.

[Breve descripción de los dibujos]

50 Los dibujos adjuntos, que están incorporados en una parte de la memoria descriptiva y constituyen la misma, muestran realizaciones a modo de ejemplo, características y aspectos de la invención y, junto con la descripción, sirven para explicar los principios de la invención.

[figura 1]

55 La figura 1 es un diagrama de flujo que muestra un procedimiento de codificación de un parámetro de calidad de la imagen, según una primera realización a modo de ejemplo de la presente invención.

[figura 2]

La figura 2 es un diagrama de flujo que muestra un procedimiento de decodificación de un parámetro de calidad de la imagen, según una segunda realización a modo de ejemplo de la presente invención.

60 [figura 3]

La figura 3 es un diagrama de flujo que muestra un procedimiento de codificación de un parámetro de calidad de la imagen, según una tercera realización a modo de ejemplo de la presente invención.

[figura 4]

65 La figura 4 es un diagrama de flujo que muestra un procedimiento de decodificación de un parámetro de calidad de la imagen, según una cuarta realización a modo de ejemplo de la presente invención.

[figura 5]

La figura 5 es un diagrama de flujo que muestra un procedimiento de codificación de un parámetro de calidad de la imagen, según una quinta realización a modo de ejemplo de la presente invención.

5

[figura 6]

La figura 6 es un diagrama de flujo que muestra un procedimiento de decodificación de un parámetro de calidad de la imagen, según una sexta realización a modo de ejemplo de la presente invención.

10

[figura 7]

La figura 7 es un diagrama que muestra un ejemplo de un bloque de retícula dividido.

[figura 8A]

La figura 8A es un diagrama que muestra un ejemplo de división de un bloque de retícula y un indicador de división.

15

[figura 8B]

La figura 8B es un diagrama que muestra un ejemplo de división de un bloque de retícula e indicadores de división.

20

[figura 8C]

La figura 8C es un diagrama que muestra un ejemplo de división de un bloque de retícula e indicadores de división.

[figura 8D]

La figura 8D es un diagrama que muestra un ejemplo de división de un bloque de retícula e indicadores de división.

25

[figura 9]

La figura 9 es un diagrama que muestra un ejemplo de una configuración de hardware para llevar a cabo un procedimiento de codificación y un procedimiento de decodificación, según una realización a modo de ejemplo de la presente invención.

30

[figura 10]

La figura 10 es un diagrama que muestra un ejemplo de una configuración de hardware para llevar a cabo un procedimiento de codificación, según una realización a modo de ejemplo de la presente invención.

[figura 11]

La figura 11 es un diagrama que muestra un ejemplo de una configuración de hardware para llevar a cabo un procedimiento de decodificación, según una realización a modo de ejemplo de la presente invención.

[figura 12A]

La figura 12A es un diagrama que muestra un ejemplo de control de la calidad de la imagen en bloques de división.

40

[figura 12B]

La figura 12B es un diagrama que muestra un ejemplo de control de la calidad de la imagen en bloques de división.

[figura 12C]

La figura 12C es un diagrama que muestra un ejemplo de control de la calidad de la imagen en bloques de división.

[figura 13]

La figura 13 es un diagrama de flujo que muestra un procedimiento de procesamiento de codificación en un fotograma.

50

[figura 14]

La figura 14 es un diagrama de flujo que muestra un procedimiento de procesamiento de decodificación en un fotograma.

55

[figura 15]

La figura 15 es un diagrama que muestra la estructura de datos de un fotograma codificado.

[figura 16]

La figura 16 es un diagrama que muestra la estructura de datos de datos codificados en un bloque de retícula.

60

[figura 17]

La figura 17 es un diagrama de flujo que muestra los detalles de un procedimiento de codificación de bloques de división.

[figura 18]

La figura 18 es un diagrama de flujo que muestra los detalles de un procedimiento de decodificación de bloques de división.

5 [figura 19]

La figura 19 es un diagrama de flujo que muestra un procedimiento de cálculo de un parámetro de calidad de la imagen, según la primera realización a modo de ejemplo.

10 [figura 20]

La figura 20 es un diagrama que muestra bloques de división que van acompañados por parámetros de cuantificación.

15 [figura 21A]

La figura 21A es un diagrama que muestra una relación entre un perfil, un nivel y un tamaño de bloque mínimo de control de la calidad de la imagen.

20 [figura 21B]

La figura 21B es un diagrama que muestra una relación entre un perfil, un nivel y un tamaño de bloque mínimo de control de la calidad de la imagen.

25 [figura 22]

La figura 22 es un diagrama de flujo que muestra un procedimiento para calcular un tamaño de bloque mínimo de control de la calidad de la imagen.

30 [figura 23A]

La figura 23A es un diagrama que muestra una estructura de datos codificados de una secuencia.

[figura 23B]

La figura 23B es un diagrama que muestra una estructura de datos codificados de una secuencia.

35 [figura 24]

La figura 24 es un diagrama de flujo que muestra un procedimiento de codificación de imagen, según una séptima realización a modo de ejemplo de la presente invención.

40 [figura 25]

La figura 25 es un diagrama de flujo que muestra un procedimiento de decodificación de imagen, según una octava realización a modo de ejemplo de la presente invención.

[Descripción de las realizaciones]

45 A continuación, se describirán con detalle diversas realizaciones a modo de ejemplo, características y aspectos de la invención, haciendo referencia a los dibujos.

50 Un procedimiento para codificar bloques de división, según una primera realización a modo de ejemplo de la presente invención se describirá haciendo referencia a las figuras 1, 9, 13, 17 y 19.

55 La figura 9 muestra una configuración para ejecutar un programa de codificación que incluye el procedimiento de codificación según la presente realización a modo de ejemplo. El programa de codificación se graba en un dispositivo de disco duro (de aquí en adelante, abreviado como HDD, Hard Disk Device) 903. El programa de codificación se carga en una memoria de acceso aleatorio (RAM, Random Access Memory) 902 tras la activación. Una unidad central de procesamiento (CPU, Central Processing Unit) 901 lleva a cabo las etapas que se describen a continuación para llevar a cabo el procesamiento de codificación de imagen. En la presente realización a modo de ejemplo, los datos de imagen de entrada se almacenan y se leen desde el HDD 903. Los datos de salida se graban en el HDD 903.

60 Se describirá una operación de codificación de la configuración anterior. La figura 13 es un diagrama de flujo que muestra un procedimiento para codificar un fotograma completo. La presente realización a modo de ejemplo trata un caso en el que las pantallas se dividen cada una en una estructura de retícula en bloques de retícula de 64 x 64 píxeles y son procesados secuencialmente en unidades de bloques de retícula. Los bloques de retícula en una pantalla se procesan en un orden de escaneo de fotograma. Se debe observar que el tamaño del bloque de retícula no está limitado a 64.

65 En la etapa S1301, la CPU 901 establece un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S1302, la CPU 901 codifica el tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S1303, la CPU 901 codifica la información de la cabecera del fotograma. En la etapa S1304, la CPU 901 establece un primer bloque de retícula en la parte superior izquierda de la pantalla como un bloque de retícula objetivo. En la

etapa S1305, la CPU 901 lleva a cabo un procesamiento de predicción en el bloque de retícula objetivo para generar datos residuales. En la etapa S1306, la CPU 901 lleva a cabo un procedimiento para codificar datos residuales en el bloque de retícula objetivo. En la etapa S1307, la CPU 901 determina si se procesan todos los bloques de retícula en el fotograma. En la etapa S1308, la CPU 901 establece el siguiente bloque de retícula como un bloque de retícula objetivo.

El procedimiento de las etapas se describirá a continuación. En la etapa S1301, la CPU 901 establece un parámetro llamado tamaño de bloque mínimo de control de la calidad de la imagen. La presente realización a modo de ejemplo se describirá suponiendo que la calidad de la imagen está controlada por un parámetro de cuantificación que se utiliza para cuantificar los coeficientes de transformación ortogonal. El tamaño de bloque mínimo de control de la calidad de la imagen es de 16 x 16 píxeles, de modo que la información del tamaño de bloque mínimo de control de la calidad de la imagen tiene un valor de 16. No obstante, el valor no está limitado a 16. El desarrollador del programa puede incrustar un valor fijo en el programa. El usuario puede especificar un valor a través de una interfaz de usuario del programa. El propio programa puede calcular y establecer un valor óptimo. El programa puede calcular valores óptimos dependiendo de las imágenes de entrada. Por ejemplo, si una imagen de entrada es mayor que 640 x 480 píxeles, el programa puede establecer el tamaño de bloque mínimo de control de la calidad de la imagen en un valor tan grande como 16. Si la resolución es menor o igual que 640 x 480 píxeles, el programa puede establecer el tamaño de bloque mínimo de control de la calidad de la imagen en un valor tan pequeño como 8.

En la etapa S1302, la CPU 901 codifica el tamaño de bloque mínimo de control de la calidad de la imagen establecido en la etapa S1301 para generar el código del tamaño de bloque mínimo de control de la calidad de la imagen. Por ejemplo, la CPU 901 puede proporcionar un índice que expresa información del tamaño de bloque mínimo de control de la calidad de la imagen de 64 como "0" y, de manera similar 32, 16 y 8 como "1", "2" y "3", respectivamente. Entonces, la CPU 901 puede generar un código de longitud fija que representa el índice.

En la etapa S1303, la CPU 901 genera y codifica la información de la cabecera del fotograma. La CPU 901 incluye el código de tamaño de bloque mínimo de control de la calidad de la imagen generado en la etapa S1302 como parte de la información de la cabecera del fotograma. Se debe observar que, si bien la presente realización a modo de ejemplo trata el caso en el que el código del tamaño de bloque mínimo de control de la calidad de la imagen está incluido en una cabecera de fotograma, esto no es restrictivo. El código del tamaño de bloque mínimo de control de la calidad de la imagen puede estar incluido en una secuencia, una imagen, un fragmento o un bloque del nivel jerárquico superior.

En la etapa S1304, la CPU 901 establece un bloque de retícula en la parte superior izquierda de la pantalla como un primer bloque de retícula objetivo a procesar.

En la etapa S1305, la CPU 901 genera una imagen predicha utilizando predicción intrafotograma y predicción de compensación de movimiento, y genera datos residuales de los píxeles del bloque de retícula objetivo. La predicción intrafotograma incluye generar una imagen predicha haciendo referencia a la proximidad del bloque de retícula objetivo. La predicción de compensación de movimiento incluye generar una imagen predicha haciendo referencia a píxeles en otros fotogramas. Los ejemplos incluyen intrapredicción y predicción de compensación de movimiento que se adoptan en el esquema de codificación H.264 mencionado anteriormente. La CPU 901 codifica un modo de predicción intrafotograma y/o vectores de movimiento generados, y envía la resultante a una memoria intermedia de salida como código de la información de la cabecera del bloque de retícula.

En la etapa S1306, la CPU 901 lleva a cabo un procedimiento para codificar datos residuales en un bloque, que se describirá junto con la figura 17, en el bloque de retícula objetivo. La figura 17 se describirá en detalle a continuación.

En la etapa S1307, la CPU 901 determina si todos los bloques de retícula en el fotograma están procesados. Si todos los bloques de retícula están procesados (Sí, en la etapa S1307), la CPU 901 finaliza el procesamiento de codificación en todo el fotograma. Si todavía no se han procesado todos los bloques de retícula (NO, en la etapa S1307), la CPU 901 lleva a cabo la etapa S1308.

En la etapa S1308, la CPU 901 establece el siguiente bloque de retícula en un orden de escaneo de fotograma como un bloque de retícula objetivo, y vuelve a la etapa S1305.

La figura 17 es un diagrama de flujo detallado de la etapa S1306. En la etapa S1306, la CPU 901 codifica una pluralidad de bloques de transformación ortogonal estructurados jerárquicamente, de diferentes tamaños, para generar datos codificados que incluyen código de parámetro de control de calidad de la imagen. Específicamente, la CPU 901 lleva a cabo la división de bloques, una transformación ortogonal y la cuantificación en base a un tamaño de bloque mínimo de control de la calidad de la imagen, a datos de tamaño de bloque y a datos residuales en los bloques. El diagrama de flujo muestra, asimismo, el procesamiento para codificar bloques de división para generar código de indicador de división de bloque, datos de coeficientes codificados y código de QP_DELTA relacionados con la actualización de un parámetro de calidad de la imagen. La presente realización a modo de ejemplo utiliza una estructura de árbol cuaternario de región para la estructuración jerárquica. Los bloques en el nivel jerárquico más

alto son bloques de retícula de 64 x 64 píxeles. A medida que el nivel desciende mediante procesamiento de la división, bloques de 32 x 32 píxeles, 16 x 16 píxeles, y tan pequeños como de 8 x 8 píxeles pueden ser procesados. La estructura de división de bloques basada en árbol cuaternario de región se puede transmitir al lado de un decodificador por medio de indicadores de división de bloques incrustados en un flujo.

5 En la etapa S1700, la CPU 901 calcula los costes cuando un bloque objetivo es dividido y cuando no. En la etapa S1701, la CPU 901 determina si dividir el bloque objetivo. En la etapa S1702, la CPU 901 codifica un indicador de división de bloque establecido como falso y genera el resultado. En la etapa S1703, la CPU 901 lleva a cabo un procedimiento para calcular un parámetro de calidad de la imagen. En la etapa S1704, la CPU 901 lleva a cabo una transformación ortogonal, una cuantificación y una codificación de longitud variable. En la etapa S1705, la CPU 901 lleva a cabo un procedimiento para codificar el parámetro de calidad de la imagen. En la etapa S1706, la CPU 901 clasifica los datos codificados y envía la resultante a una memoria intermedia de salida. En la etapa S1707, la CPU 901 codifica un indicador de división de bloques establecido como verdadero y genera la resultante. En la etapa S1708, la CPU 901 lleva a cabo el procedimiento para calcular un parámetro de calidad de la imagen. En la etapa S1709, la CPU 901 lleva a cabo el procedimiento para codificar el parámetro de calidad de la imagen. En la etapa S1710, la CPU 901 procesa bloques de división incluidos.

10 20 El procedimiento de las etapas se describirá a continuación. En la etapa S1700, la CPU 901 calcula los costes cuando el bloque es dividido y cuando no. Los costes se pueden calcular utilizando los multiplicadores de Lagrange mencionados anteriormente, así como las características de la imagen, estadísticas y longitudes de código predichas.

25 En la etapa S1701, la CPU 901 compara los costes con y sin división, calculados en la etapa S1700. Si el coste con división de bloque es menor que el coste sin división de bloque (Sí, en la etapa S1701), la CPU 901 determina que la comparación es verdadero. Si no (NO, en la etapa S1701), la CPU 901 determina que la comparación es falso. Si el resultado de la determinación en la etapa S1701 es falso (NO, en la etapa S1701), la CPU 901 lleva a cabo las etapas S1702 a S1706. Si el resultado de la determinación en la etapa S1701 es verdadero (Sí, en la etapa S1701), la CPU 901 lleva a cabo las etapas S1707 a S1710.

30 35 40 En la etapa S1702, la CPU 901 codifica un indicador de división de bloque establecido como falso. La CPU 901 habitualmente asigna y codifica un valor de un bit de 0, aunque no está limitado a este.

45 50 En la etapa S1703, la CPU 901 lleva a cabo un procedimiento de cálculo de parámetros de calidad de la imagen mostrado en la figura 19 para calcular un parámetro de calidad de la imagen. La figura 19 se describirá en detalle a continuación.

55 60 En la etapa S1704, la CPU 901 lleva a cabo una transformación ortogonal, una cuantificación y una codificación de longitud variable en datos residuales para generar datos de coeficientes codificados. La transformación ortogonal se implementa mediante una transformada discreta del coseno (DCT, Discrete Cosine Transform). La cuantificación se lleva a cabo utilizando el parámetro de calidad de la imagen calculado en la etapa S1703. La CPU 901 lleva a cabo la cuantificación utilizando el mismo valor del parámetro de calidad de la imagen que el utilizado para la cuantificación anterior si no se calcula un nuevo parámetro de calidad de la imagen mediante el procedimiento de cálculo del parámetro de calidad de la imagen mostrado en la figura 19 que se describirá a continuación. Para la codificación de longitud variable, la CPU 901 puede utilizar la codificación de Huffman y la codificación aritmética. Dichas técnicas de codificación se pueden implementar utilizando procedimientos similares a los de H.264 mencionados anteriormente.

65 70 En la etapa S1705, la CPU 901 lleva a cabo el procedimiento de codificación de parámetros de calidad de la imagen. El procedimiento de codificación de parámetros de calidad de la imagen se describirá en detalle a continuación haciendo referencia a la figura 1.

75 80 En la etapa S1706, la CPU 901 ordena el código de indicador de división de bloques, el código de QP_DELTA y los datos de coeficientes codificados, y envía la resultante a una memoria intermedia de salida en este orden. Si no hay datos codificados para enviar, la CPU 901 solo clasifica y genera los datos codificados existentes. La figura 16 muestra un ejemplo del orden de código.

85 90 A continuación, se proporcionará una descripción del caso en el que el coste es menor con división de bloques. En la etapa S1707, la CPU 901 codifica un indicador de división de bloque establecido como verdadero. La CPU 901 habitualmente asigna y codifica un valor de un bit de 1, aunque no está limitado a este.

95 100 En la etapa S1708, la CPU 901 lleva a cabo el procedimiento de cálculo del parámetro de calidad de la imagen mostrado en la figura 19 que se describirá a continuación, calculando de este modo un parámetro de calidad de la imagen.

En la etapa S1709, la CPU 901 lleva a cabo el procedimiento de codificación de parámetros de calidad de la imagen como en la etapa S1705. El procedimiento de codificación de parámetros de calidad de la imagen se describirá en detalle a continuación haciendo referencia a la figura 1.

- 5 En la etapa S1710, la CPU 901 lleva a cabo recursivamente el presente procedimiento en todos los bloques de división incluidos, es decir, cuatro bloques de división (superior izquierdo, superior derecho, inferior izquierdo e inferior derecho) que están incluidos en el nivel jerárquicamente posterior. Cada uno de los bloques de división tiene un tamaño de 1/2 del bloque objetivo original, tanto en altura como en anchura. El tamaño del bloque objetivo disminuye división a división. El procesamiento recursivo puede ser llevado a cabo hasta un tamaño de bloque tan pequeño como de 8 x 8 píxeles. Mediante dicho procesamiento recursivo, un bloque de retícula es dividido sucesivamente en una estructura de árbol cuaternario de región. Por ejemplo, supóngase que un bloque de retícula está dividido tal como se muestra en la figura 7. Un indicador de división para indicar la presencia o la ausencia de división de bloque se indicará con 1 cuando un bloque correspondiente está dividido, y 0 cuando no. Los indicadores de división en los bloques de división respectivos se expresan tal como se muestra en las figuras 8A, 8B, 8C y 8D.
- 10 Las figuras 8A, 8B, 8C y 8D muestran los niveles y los bloques de 64 x 64 píxeles, 32 x 32 píxeles, 16 x 16 píxeles y 8 x 8 píxeles, respectivamente. Los números indican indicadores de división. “-” indica que no se necesita ningún indicador. Los números entre paréntesis indican el orden estructural. En la presente realización a modo de ejemplo, el orden estructural hace referencia al orden de superior izquierdo, superior derecho, inferior izquierdo e inferior derecho en términos de bloques. En términos de niveles, el orden estructural hace referencia al orden hacia delante de escaneo de la estructura de árbol cuaternario de región. Los conjuntos de código de indicador de división, código de QP_DELTA y datos de coeficientes codificados se almacenan en el orden estructural. No se ajusta ningún indicador de división en bloques de 8 x 8 píxeles, ya que dichos bloques no se dividirán más. Un bloque que tiene un indicador de división de 1 no contiene datos de coeficientes en sí, ya que dicho bloque está subdividido en bloques de división de menor tamaño que contienen datos de coeficientes. Tal como se muestra en la figura 16 que se describirá a continuación, un flujo tiene una estructura de datos tal que los fragmentos de información codificada en bloques de división respectivos están dispuestos en el orden estructural. Un fragmento de información codificada en cada bloque de división incluye un conjunto de código de indicador de división, código de QP_DELTA y/o datos de coeficientes en el bloque de división.
- 15
- 20
- 25
- 30 La figura 1 es un diagrama de flujo detallado del procedimiento de codificación de parámetros de calidad de la imagen llevado a cabo en las etapas S1705 y S1709. En la etapa S101, la CPU 901 obtiene un tamaño del bloque objetivo. En la etapa S102, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S103, la CPU 901 obtiene el estado de división del bloque. En la etapa S104, la CPU 901 determina si el bloque objetivo está dividido. En la etapa S105, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S106, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S107, la CPU 901 obtiene un parámetro de calidad de la imagen. En la etapa S108, la CPU 901 codifica el parámetro de calidad de la imagen.
- 35
- 40 El procedimiento de las etapas se describirá a continuación. En la etapa S101, la CPU 901 obtiene un tamaño del bloque objetivo. Inmediatamente después de que se llame al presente procedimiento desde el procedimiento de la figura 17, el bloque objetivo que se procesará es un bloque de retícula en sí mismo. El valor del tamaño del bloque objetivo es, por lo tanto, 64. A medida que el nivel de la estructura del árbol cuaternario de la región desciende posteriormente debido a llamadas recursivas, el valor del tamaño cambia a 32, 16 y 8.
- 45
- 50 En la etapa S102, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen que se establece desde el exterior. En la presente realización a modo de ejemplo, el valor del tamaño de bloque mínimo de control de la calidad de la imagen es 16. Si el bloque objetivo a codificar está dividido o no, se ha determinado en las etapas S1700 y S1701 de la figura 17. En la etapa S103, la CPU 901 obtiene la información del estado de división que indica el resultado de la determinación de división.
- 55
- 60 En la etapa S104, la CPU 901 hace referencia al estado de división del bloque objetivo obtenido, y determina si el bloque objetivo está dividido. Si el bloque objetivo está dividido (Sí, en la etapa S104), la CPU 901 lleva a cabo la etapa S106. Si el bloque objetivo no está dividido (NO, en la etapa S104), la CPU 901 lleva a cabo la etapa S105.
- 65 En la etapa S105, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. Si se determina que el tamaño del bloque objetivo es mayor o igual (Sí, en la etapa S105), la CPU 901 lleva a cabo la etapa S107 y la etapa siguiente. Si no (NO, en la etapa S105), la CPU 901 finaliza el procesamiento de codificación de parámetros de calidad de la imagen.
- En la etapa S106, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. Si se determina que el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen (Sí, en la etapa S106), la CPU 901 lleva a cabo las etapas S107 y S108. Si no (NO, en la etapa S108), la CPU 901 finaliza el procesamiento de codificación del parámetro de calidad de la imagen.

En la etapa S107, la CPU 901 obtiene un parámetro de calidad de la imagen. El parámetro de calidad de la imagen es el determinado en la etapa S1703 o S1708 de la figura 17. En la etapa S108, la CPU 901 codifica el parámetro de calidad de la imagen. En la presente realización a modo de ejemplo, la CPU 901 determina un valor de diferencia entre el parámetro de calidad de la imagen utilizado para codificar el bloque anterior y el parámetro de calidad de la imagen obtenido en la etapa S107 como un valor de QP_DELTA. La CPU 901 codifica el valor de QP_DELTA determinado mediante la codificación de Golomb en el código de QP_DELTA, que es el código del parámetro de calidad de la imagen. En la presente realización a modo de ejemplo, QP_DELTA está codificado mediante codificación de Golomb. No obstante, esto no es restrictivo. QP_DELTA se puede codificar mediante codificación aritmética u otra codificación de longitud variable. El parámetro de calidad de la imagen se utiliza para calcular el siguiente valor de QP_DELTA.

La figura 19 es un diagrama de flujo detallado del procedimiento de cálculo del parámetro de calidad de la imagen llevado a cabo en las etapas S1703 y S1708.

En la etapa S1901, la CPU 901 obtiene un tamaño del bloque objetivo. En la etapa S1902, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S1903, la CPU 901 obtiene el estado de división del bloque. En la etapa S1904, la CPU 901 determina si el bloque objetivo está dividido. En la etapa S1905, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual al tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S1906, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S1907, la CPU 901 calcula un parámetro de calidad de la imagen.

El procedimiento de las etapas se describirá a continuación. En la etapa S1901, la CPU 901 obtiene un tamaño del bloque objetivo. Inmediatamente después de que se llame al presente procedimiento desde el procedimiento de la figura 17, el bloque objetivo a procesar es un bloque de retícula en sí mismo. El valor del tamaño del bloque objetivo es, por lo tanto, 64. A medida que el nivel de la estructura del árbol cuaternario de la región desciende posteriormente debido a llamadas recursivas, el valor del tamaño cambia a 32, 16 y 8.

En la etapa S1902, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen que se establece desde el exterior. En la presente realización a modo de ejemplo, el tamaño de bloque mínimo de control de la calidad de la imagen tiene un valor de 16.

Si el bloque objetivo a codificar está dividido o no se ha determinado en las etapas S1700 y S1701 de la figura 17. En la etapa S1903, la CPU 901 obtiene información del estado de división que indica el resultado de la determinación de división.

En la etapa S1904, la CPU 901 hace referencia al estado de división del bloque objetivo obtenido, y determina si el bloque objetivo está dividido. Si el bloque objetivo está dividido (Sí, en la etapa S1904), la CPU 901 lleva a cabo la etapa S1906. Si el bloque objetivo no está dividido (NO, en la etapa S1904), la CPU 901 lleva a cabo la etapa S1905.

En la etapa S1905, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. Si se determina que el tamaño del bloque objetivo es mayor o igual (Sí, en la etapa S1905), la CPU 901 lleva a cabo la etapa S1907. Si no (NO, en la etapa S1905), la CPU 901 finaliza el procesamiento del cálculo del parámetro de calidad de la imagen.

En la etapa S1906, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. Si se determina que el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen (Sí, en la etapa S1906), la CPU 901 lleva a cabo la etapa S1907. Si no (NO, en la etapa S1906), la CPU 901 finaliza el procesamiento de cálculo del parámetro de calidad de la imagen.

En la etapa S1907, la CPU 901 lleva a cabo el procesamiento de control de velocidad para calcular un parámetro de calidad de la imagen para codificar el bloque objetivo. Por ejemplo, como con TM5, la CPU 901 predice la cantidad de códigos de bloque objetivo y calcula un parámetro de cuantificación óptimo. Si el bloque objetivo está dividido, todos los bloques de división incluidos están sujetos al control de velocidad. Los bloques de división incluidos son codificados con el mismo parámetro de calidad de la imagen.

El procedimiento de codificación según la presente realización a modo de ejemplo se utiliza para generar datos que tienen una estructura de datos mostrada en las figuras 15 y 16. Haciendo referencia a la figura 15, la información de la cabecera del fotograma incluye el código del tamaño de bloque mínimo de control de la calidad de la imagen que se incrusta en la etapa S1303 de la figura 13. Todos los bloques de retícula dentro del fotograma están bajo el control del código. Haciendo referencia a la figura 16, un indicador de división en un bloque de retícula, o un bloque de 64 x 64 píxeles, está seguido por un indicador de división en un bloque superior izquierdo de 32 x 32 píxeles en el nivel posterior. Si el bloque de 32 x 32 píxeles no está dividido, sigue el código del parámetro de calidad de la imagen o el código de QP_DELTA de ese bloque, acompañado del código de datos del coeficiente en el residuo de los 32 x 32 píxeles. Sigue un indicador de división en un bloque superior derecho de 32 x 32 píxeles. Tal como se

dijo anteriormente, si un bloque de 32 x 32 píxeles se divide en bloques de 16 x 16 píxeles, sigue un indicador de división para indicar el estado de la división. Si un bloque no se divide más, sigue el código de QP_DELTA y el código de datos del coeficiente. El tamaño de bloque mínimo de control de la calidad de la imagen es 16. Si un bloque se divide en bloques de menos de 16 x 16 píxeles, sigue un indicador de división y un código de datos de coeficiente sin el código de QP_DELTA agregado a los bloques más pequeños.

Tal como se describió anteriormente, un tamaño del bloque objetivo es comparado con el tamaño de bloque mínimo de control de la calidad de la imagen, y si se codifica un valor de QP_DELTA se controla según el resultado de la comparación. La provisión de dicho mecanismo permite el control de la calidad de la imagen en unidades de bloques en las que una imagen se divide en un patrón de retícula. El control de la calidad de la imagen se puede llevar a cabo en unidades más finas (figura 12C) que en un caso convencional donde el código de QP_DELTA está incrustado en unidades de bloques de retícula (figura 12A). Esto puede estrechar los bordes entre bloques de diferentes calidades de imagen y hacer que las diferencias en la calidad de la imagen sean menos apreciables. En otro caso convencional, el control de la calidad de la imagen se lleva a cabo y el código de QP_DELTA está incorporado en unidades de bloques de transformación ortogonal (figura 12B). A diferencia de este caso, la cantidad de código no aumentará más de lo necesario, porque la unidad de control de la calidad de la imagen puede ser seleccionada y el código de QP_DELTA no necesita ser incrustado en cada bloque de transformación ortogonal. El procedimiento de cálculo del parámetro de calidad de la imagen se lleva a cabo solo si se cumple la misma condición que para introducir un parámetro de calidad de la imagen. De este modo, se evita que el cálculo del parámetro de calidad de la imagen también aumente más de lo necesario.

En la presente realización a modo de ejemplo, los bloques se definen como bloques cuadrados. Se comprenderá que los bloques pueden ser rectangulares, tal como de 8 x 4 píxeles. Aunque la estructura del árbol cuaternario de la región se escanea utilizando un procesamiento recursivo, las estructuras de bucle se pueden utilizar para implementar el escaneo.

Tal como se muestra en la figura 10, el procedimiento de codificación según la presente realización a modo de ejemplo se puede llevar a cabo mediante una configuración que incluye hardware dedicado, tal como un transformador ortogonal 1003, un cuantificador 1004 y un codificador de longitud variable 1005. La CPU 1001 no llevará a cabo la transformación ortogonal, la cuantificación y la etapa de codificación de longitud variable, sino que hace que las unidades anteriores lleven a cabo las partes respectivas del procesamiento. En otros aspectos, el procedimiento es el mismo que el mostrado en la figura 17.

La presente realización a modo de ejemplo ha tratado el caso en el que los datos codificados están grabados en el HDD 903. No obstante, esto no es restrictivo. Los datos codificados pueden ser grabados en un medio de grabación que no sea el dispositivo de disco duro 903. Se apreciará que los datos codificados pueden ser transmitidos a un circuito de comunicación a través de una interfaz de comunicación 905. Un dispositivo de grabación externo puede estar conectado para grabar datos codificados en un medio portátil.

La presente realización a modo de ejemplo también ha tratado el caso en el que las aplicaciones de software están grabadas en el dispositivo de disco duro 903. No obstante, esto no es restrictivo. Las aplicaciones de software pueden estar grabadas en un fragmento de hardware, tal como una memoria de solo lectura (ROM, Read Only Memory). Se apreciará que las aplicaciones de software pueden estar grabadas en un medio portátil, tal como una tarjeta de memoria y un disco.

La presente realización a modo de ejemplo también ha tratado el caso en el que el número de veces de división se codifica para producir código que indica el tamaño de un bloque mínimo de control de la calidad de la imagen. No obstante, esto no es restrictivo. Puesto que el tamaño de bloque máximo es el de un bloque de retícula, el tamaño de un bloque mínimo de control de la calidad de la imagen se puede expresar mediante un código de longitud fija de 6 bits. Un bloque mínimo de control de la calidad de la imagen no está limitado a una configuración cuadrada. Los bloques mínimos de control de la calidad de la imagen para configuraciones horizontales y verticales se pueden proporcionar por separado. Por ejemplo, los bloques horizontales u horizontalmente adyacentes pueden ser agrupados y los parámetros de calidad de la imagen pueden ser controlados en unidades de grupos de bloques. En tal caso, los tamaños de bloque mínimo de control de la calidad de la imagen pueden establecerse de manera efectiva para configuraciones horizontales y verticales de manera separada.

Un procedimiento de decodificación de imagen según una segunda realización a modo de ejemplo de la presente invención se describirá haciendo referencia a las figuras 2, 9, 14 y 18.

La figura 9 muestra una configuración para ejecutar un programa de decodificación que incluye el procedimiento de decodificación según la presente realización a modo de ejemplo. El programa de grabación se graba en el HDD 903. El programa de decodificación se carga en la RAM 902 y la CPU 901 lleva a cabo las etapas de un procedimiento que se describirá a continuación para el procesamiento de decodificación. Los datos codificados de entrada se leen desde el HDD 903. Una imagen decodificada se envía a una pantalla a través de una memoria intermedia de salida de imagen que está dispuesta en una RAM de video (VRAM, Video RAM) 904. La CPU 901 lleva a cabo el procesamiento de decodificación bloque por bloque. Cada bloque se decodifica para producir una imagen

5 decodificada del bloque, que se copia en una posición apropiada en un fotograma cuando se envía a la memoria intermedia de salida de imágenes. Cuando se completa el procesamiento de decodificación en bloques tanto como en un solo fotograma, un fotograma de imagen es emitido a una memoria intermedia de salida. La siguiente descripción se proporcionará con la suposición de que los datos codificados de entrada son datos codificados que se generan en la primera realización a modo de ejemplo.

10 La figura 14 muestra el procedimiento para decodificar datos codificados de entrada. Los bloques de retícula se decodifican en un orden de escaneo de fotograma. En la etapa S1401, la CPU 901 decodifica y obtiene la 15 información de la cabecera del fotograma. La presente realización a modo de ejemplo trata el caso en el que la información de la cabecera del fotograma incluye un tamaño de bloque mínimo de control de la calidad de la imagen. No obstante, esto no es restrictivo. Según un formato de datos predeterminado, se puede obtener un tamaño de 20 bloque mínimo de control de la calidad de la imagen a partir de la información de la cabecera en una secuencia, una imagen, un fragmento o un bloque del nivel jerárquico más alto. En la etapa S1402, la CPU 901 decodifica y obtiene un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S1403, la CPU 901 lleva a cabo una 25 configuración para que el resultado de la decodificación de los datos codificados se almacene primero en un primer bloque de retícula objetivo. En la etapa S1404, la CPU 901 decodifica el modo de predicción del bloque de retícula objetivo y genera una imagen predicha según el modo de predicción. En la etapa S1405, la CPU 901 lleva a cabo un 30 procedimiento de decodificación de bloque de división en el bloque de retícula objetivo. En la etapa S1406, la CPU 901 agrega datos de imagen predichos y datos residuales del bloque de retícula objetivo, y envía la resultante a la memoria intermedia de salida de imagen. En la etapa S1407, la CPU 901 determina si se procesan todos los 35 bloques de retícula en un fotograma. En la etapa S1408, la CPU 901 establece el siguiente bloque de retícula como un bloque de retícula objetivo.

40 El procedimiento de las etapas se describirá a continuación. En la etapa S1401, la CPU 901 obtiene un parámetro llamado tamaño de bloque mínimo de control de la calidad de la imagen a partir de la información de la cabecera en un bloque que es del orden más alto en términos de secuencias, imágenes, fotogramas, fragmentos y estructura jerárquica.

45 En la etapa S1402, la CPU 901 decodifica y obtiene el tamaño de bloque mínimo de control de la calidad de la imagen. Según la primera realización a modo de ejemplo, el tamaño de bloque mínimo de control de la calidad de la imagen se codifica como un índice. El tamaño de bloque mínimo de control de la calidad de la imagen es 16. En la etapa S1403, la CPU 901 establece el bloque de retícula en la parte superior izquierda de la pantalla como un 50 bloque de retícula objetivo.

55 En la etapa S1404, la CPU 901 decodifica datos codificados en el bloque de retícula objetivo. De este modo, la CPU 901 decodifica un modo de predicción intrafotograma y/o vectores de movimiento generados en la primera realización a modo de ejemplo. Con intracodificación, la CPU 901 hace referencia a píxeles decodificados adyacentes y genera datos de imagen predichos basados en el modo de predicción intrafotograma decodificado. Con intercodificación, la CPU 901 lleva a cabo una compensación de movimiento para generar datos de imagen predichos basados en los vectores de movimiento.

60 En la etapa S1405, la CPU 901 lleva a cabo un procedimiento de decodificación de bloque de división, que se describirá a continuación junto con la figura 18, en el bloque de retícula. Puesto que la decodificación de bloques de división está estructurada jerárquicamente, la CPU 901 decodifica indicadores de división en bloques y determina los estados de los bloques de división mientras lleva a cabo la decodificación. La CPU 901 genera con ello parámetros de calidad de la imagen y datos residuales con respecto a las predicciones.

65 En la etapa S1406, la CPU 901 agrega los datos de imagen predichos generados en la etapa S1404 y los datos residuales generados en la etapa S1405 para generar una imagen decodificada del bloque de retícula objetivo. La CPU 901 envía la imagen decodificada a la memoria intermedia de salida de imagen.

70 En la etapa S1407, la CPU 901 determina si todos los bloques de retícula en el fotograma se han procesado para la decodificación. Por ejemplo, la CPU 901 cuenta el número de bloques de retícula decodificados. Si no todos los bloques de retícula están procesados para decodificación (NO, en la etapa S1407), la CPU 901 pasa a la etapa S1408. Si todos los bloques de retícula están decodificados (SÍ, en la etapa S1407), la CPU 901 finaliza el proceso de decodificación en el fotograma.

75 En la etapa S1408, la CPU 901 establece el resultado de la decodificación que viene a continuación en un orden de escaneo de fotograma como el siguiente bloque de retícula objetivo.

80 La figura 18 es un diagrama de flujo detallado de la etapa S1405, que muestra un procedimiento para decodificar una pluralidad de bloques de transformación ortogonal estructurados jerárquicamente, de diferentes tamaños. Específicamente, la figura 18 muestra un procedimiento para llevar a cabo una decodificación de longitud variable, una transformación ortogonal inversa y una cuantificación inversa en datos de coeficientes codificados de longitud variable basados en el código de indicador de división de bloque y el código de QP_DELTA, para generar datos residuales. La presente realización a modo de ejemplo utiliza una estructura de árbol cuaternario de región para la

estructuración jerárquica. Un bloque en el nivel más alto (primer bloque objetivo que se procesará) es un bloque de retícula de 64 x 64 píxeles.

- 5 En la etapa S1801, la CPU 901 decodifica un indicador de división de bloque en el bloque objetivo. En la etapa S1802, la CPU 901 lleva a cabo un procedimiento de decodificación de parámetros de calidad de la imagen. En la etapa S1803, la CPU 901 determina si el bloque objetivo a decodificar está dividido. En la etapa S1804, la CPU 901 lleva a cabo una decodificación de longitud variable, una cuantificación inversa y una transformación ortogonal inversa. En la etapa S1805, la CPU 901 lleva a cabo recursivamente el procesamiento de decodificación en los bloques de división incluidos.
- 10 10 El procedimiento de las etapas se describirá a continuación. En la etapa S1801, la CPU 901 decodifica un indicador de división de bloque, tal como el mostrado en la figura 16, para obtener información sobre si el bloque objetivo está subdividido. La decodificación se lleva a cabo según una estructura jerárquica de árbol cuaternario. De este modo, la CPU 901 puede obtener el estado de división de bloques.
- 15 15 En la etapa S1802, la CPU 901 lleva a cabo un procedimiento de decodificación del parámetro de calidad de la imagen que se describirá a continuación junto con la figura 2. La CPU 901 decodifica de este modo un parámetro de calidad de la imagen que se utiliza en la cuantificación inversa para decodificar el bloque objetivo.
- 20 20 En la etapa S1803, la CPU 901 hace referencia al indicador de división de bloque decodificado para determinar si el bloque objetivo está dividido. Si se determina que el bloque objetivo no está dividido (NO, en la etapa S1803), la CPU 901 lleva a cabo la etapa S1804. Si se determina que el bloque objetivo está dividido (SÍ, en la etapa S1803), la CPU 901 lleva a cabo la etapa S1805.
- 25 25 En la etapa S1804, la CPU 901 lleva a cabo una decodificación de longitud variable, una cuantificación inversa y una transformación ortogonal inversa en datos de coeficientes codificados del bloque objetivo, generando de este modo datos residuales. Para la decodificación de longitud variable, la CPU 901 lleva a cabo la decodificación según el esquema de codificación utilizado en la primera realización a modo de ejemplo. La CPU 901 obtiene de este modo los datos del coeficiente de cuantificación incluidos en el bloque. Para la cuantificación inversa, la CPU 901 utiliza el parámetro de calidad de la imagen decodificado en la etapa S1802. Si no se decodifica un nuevo parámetro de calidad de la imagen en S1802, la CPU 901 lleva a cabo una cuantificación inversa simplemente utilizando el parámetro de calidad de la imagen previamente decodificado. La CPU 901 lleva a cabo una transformación ortogonal inversa en los datos de coeficientes resultantes de la cuantificación inversa, generando de este modo datos residuales.
- 30 30 35 En la etapa S1805, la CPU 901 lleva a cabo recursivamente el procesamiento en las etapas S1801 a S1805 siempre que un indicador de división de bloques decodificado indique la presencia de un nivel jerárquicamente posterior. La CPU 901 restaura con ello los datos residuales hasta que se alcanza el tamaño de bloque mínimo de división.
- 40 40 A continuación, la etapa S1802 destinada a decodificar un parámetro de calidad de la imagen se describirá en detalle haciendo referencia a la figura 2.
- 45 45 La figura 2 muestra un procedimiento para decodificar un parámetro de calidad de la imagen de una pluralidad de bloques de transformación ortogonal estructurados jerárquicamente, de diferentes tamaños. En la etapa S201, la CPU 901 obtiene un tamaño del bloque objetivo. En la etapa S202, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S203, la CPU 901 obtiene el estado de división del bloque. En la etapa S204, la CPU 901 determina si el bloque objetivo está dividido. En la etapa S205, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S206, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S207, la CPU 901 decodifica un parámetro de calidad de la imagen.
- 50 50 55 El procedimiento de las etapas se describirá a continuación. En la etapa S201, la CPU 901 obtiene un tamaño del bloque objetivo. Inmediatamente después de que se llama al presente procedimiento desde el procedimiento de la figura 18, el bloque objetivo a procesar es un bloque de retícula en sí mismo. El valor del tamaño del bloque objetivo es, por lo tanto, 64. A medida que los indicadores de división son decodificados posteriormente, el valor del tamaño cambia a 32, 16 y 8 en consecuencia.
- 60 60 65 En la etapa S202, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen. En la presente realización a modo de ejemplo, la CPU 901 obtiene el tamaño de bloque mínimo de control de la calidad de la imagen decodificado en la etapa S1402 de la figura 14. En la etapa S203, la CPU 901 obtiene el estado de división del bloque. En la presente realización a modo de ejemplo, la CPU 901 obtiene el indicador de división de bloque que fue decodificado en la etapa S1801 de la figura 18.
- 65 65 En la etapa S204, la CPU 901 determina el valor del indicador de división de bloque. Utilizando la misma asignación de código que la descrita en la primera realización a modo de ejemplo, un indicador de división de bloque que tiene un valor de "0" indica que el bloque no está dividido. Un indicador de división de bloque que tiene un valor de "1"

indica que el bloque está dividido. Si el valor del indicador de división de bloque es "0" (NO, en la etapa S204), la CPU 901 lleva a cabo la etapa S205. Si el valor es "1" (SÍ, en la etapa S204), la CPU 901 lleva a cabo la etapa S206.

- 5 En la etapa S205, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. Si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen (SÍ, en la etapa S205), la CPU 901 lleva a cabo la etapa S207. Si no (NO, en la etapa S205), la CPU 901 finaliza el procesamiento.
- 10 En la etapa S206, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. Si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen (SÍ, en la etapa S206), la CPU 901 lleva a cabo la etapa S207. Si no (NO, en la etapa S206), la CPU 901 finaliza el procesamiento.
- 15 En la etapa S207, la CPU 901 decodifica el código del parámetro de calidad de la imagen. En la presente realización a modo de ejemplo, la CPU 901 decodifica el código de QP_DELTA en un valor de QP_DELTA mediante decodificación de Golomb. En la presente realización a modo de ejemplo, QP_DELTA se decodifica mediante decodificación de Golomb. No obstante, esto no es restrictivo. QP_DELTA puede ser decodificado mediante decodificación aritmética u otra decodificación de longitud variable, dependiendo del formato de datos predeterminado. A continuación, la CPU 901 agrega el valor de QP_DELTA para el parámetro de calidad de la imagen que se utiliza para codificar el bloque anterior, generando de este modo un parámetro de calidad de la imagen.
- 20
- 25 En la presente realización a modo de ejemplo, si decodificar o no un parámetro de calidad de la imagen se controla dependiendo de la misma condición que para comparar el tamaño del bloque objetivo en el que se codifica el parámetro de calidad de la imagen y el tamaño de bloque mínimo de control de la calidad de la imagen. Esto permite decodificar un flujo generado por el procedimiento de codificación descrito en la primera forma de realización a modo de ejemplo.
- 30 Tal como se muestra en la figura 11, el procedimiento de decodificación según la presente realización a modo de ejemplo puede ser llevado a cabo mediante una configuración que incluye hardware dedicado, tal como un transformador ortogonal inverso 1103, un cuantificador inverso 1104 y un decodificador de longitud variable 1105, en lugar de la configuración mostrada en la figura 9. La CPU 1101 no llevará a cabo la transformación ortogonal inversa, la cuantificación inversa y la etapa de decodificación de longitud variable, sino que hace que las unidades anteriores lleven a cabo las partes respectivas de procesamiento. En otros aspectos, el procedimiento es el mismo que el mostrado en la figura 18.
- 35
- 40 La presente realización a modo de ejemplo ha tratado el caso en el que los datos codificados están grabados en el dispositivo de disco duro 903. No obstante, esto no es restrictivo. Se pueden recibir datos codificados desde un circuito de comunicación a través de una interfaz de comunicación 1006. Se puede conectar un dispositivo de grabación externo para introducir datos codificados desde un medio portátil.
- 45 La presente realización a modo de ejemplo también ha tratado el caso en el que las aplicaciones de software están grabadas en el dispositivo de disco duro 903. No obstante, esto no es restrictivo. Las aplicaciones de software pueden estar grabadas en un fragmento de hardware tal como una memoria de solo lectura (ROM, Read Only Memory). Se apreciará que las aplicaciones de software pueden estar grabadas en un medio portátil, tal como una tarjeta de memoria y un disco.
- 50 Se describirá un procedimiento de codificación de un parámetro de calidad de la imagen según una tercera realización a modo de ejemplo de la presente invención haciendo referencia a la figura 3. A menos que se especifique lo contrario, la configuración para llevar a cabo el procedimiento de codificación y la operación de las etapas es la misma que en la primera realización a modo de ejemplo. En la presente realización a modo de ejemplo, un bloque que incluye un bloque objetivo y tiene el mismo tamaño que un tamaño de bloque mínimo de control de la calidad de la imagen se definirá como un bloque mínimo de control de la calidad de la imagen. La información que indica si un parámetro de calidad de la imagen está codificado en un bloque mínimo de control de la calidad de la imagen se definirá como información de estado de codificación del parámetro de calidad de la imagen. La información del estado de codificación del parámetro de calidad de la imagen es equivalente a la información que indica si un parámetro de calidad de la imagen del primer bloque de división en el bloque mínimo de control de la calidad de la imagen está codificado.
- 55
- 60 La figura 3 muestra un procedimiento para codificar un parámetro de calidad de la imagen. La figura 3 es un diagrama de flujo detallado del procedimiento de codificación del parámetro de calidad de la imagen que se llevará a cabo en las etapas S1705 y S1709 de la figura 17. Este procedimiento se lleva a cabo en lugar del procedimiento de codificación del parámetro de calidad de la imagen mostrado en la figura 1. En la etapa S301, la CPU 901 obtiene información de estado de codificación de los parámetros de calidad de la imagen. En la etapa S302, la CPU 901 hace referencia a la información del estado de codificación del parámetro de calidad de la imagen y determina si un parámetro de calidad de la imagen ya ha sido codificado en un bloque mínimo de control de la calidad de la imagen.
- 65

El procedimiento de las etapas se describirá a continuación. En la etapa S104, como la primera realización a modo de ejemplo, la CPU 901 hace referencia al estado de división del bloque objetivo obtenido, es decir, si el bloque objetivo está dividido adicionalmente mediante un árbol cuaternario. Si se determina que el bloque objetivo está dividido (Sí, en la etapa S104), la CPU 901 finaliza el procesamiento de manera diferente a la primera realización a modo de ejemplo. Si se determina que el bloque objetivo no está dividido (NO, en la etapa S104), la CPU 901 lleva a cabo la etapa S105 como la primera realización a modo de ejemplo.

5 En la etapa S105, como la primera realización a modo de ejemplo, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. Si se determina que el tamaño del bloque objetivo es mayor o igual (Sí, en la etapa S105), la CPU 901 lleva a cabo las etapas S107 y S108 como la primera realización a modo de ejemplo. Si no (NO, en la etapa S105), la CPU 901 lleva a cabo el procesamiento de las etapas S301 y S302.

10 15 En la etapa S301, la CPU 901 obtiene información del estado de codificación del parámetro de calidad de la imagen. En la presente realización a modo de ejemplo, el tamaño de bloque mínimo de control de la calidad de la imagen es 16. Si el bloque objetivo es un bloque superior izquierdo de 8 x 8 píxeles, se deduce que la CPU 901 obtiene la información de que ningún parámetro de calidad de la imagen está codificado. Si el bloque objetivo es un bloque superior derecho, inferior izquierdo o inferior derecho de 8 x 8 píxeles, un parámetro de calidad de la imagen ya ha sido codificado en un bloque superior izquierdo de 8 x 8 píxeles. En tal caso, la CPU 901 obtiene la información de que un parámetro de calidad de la imagen está codificado.

20 25 En la etapa S302, la CPU 901 hace referencia a la información del estado de codificación del parámetro de calidad de la imagen y determina si un parámetro de calidad de la imagen ya ha sido codificado en el bloque mínimo de control de la calidad de la imagen. Si no se determina que ningún parámetro de calidad de la imagen ha sido codificado (NO, en la etapa S302), la CPU 901 lleva a cabo las etapas S107 y S108. Si se determina que un parámetro de calidad de la imagen ha sido codificado (Sí, en la etapa S302), la CPU 901 finaliza el procesamiento de codificación del parámetro de calidad de la imagen.

30 35 40 En la etapa S301 de la presente realización a modo de ejemplo, el código de QP_DELTA se incrusta como datos que acompañan a un bloque objetivo (un conjunto de bloques de división) si "el bloque objetivo está dividido y el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen". En la presente realización a modo de ejemplo, el código de QP_DELTA se agrega a un bloque superior izquierdo, que es el primer bloque en un bloque objetivo, si "el bloque objetivo está dividido y el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen". En la presente realización a modo de ejemplo, el tamaño de bloque mínimo de control de la calidad de la imagen es 16. Supóngase que los bloques divididos tal como se muestra en la figura 20 constituyen una secuencia. En tal caso, el código de QP_DELTA es introducido inmediatamente antes de los fragmentos de datos del coeficiente codificado de longitud variable que corresponden a los bloques sombreados de Nº 0, 1, 7, 5, 6, 8, 9, 13, 14 y 15. En otras palabras, el código de QP_DELTA acompaña a los bloques de 16 x 16 píxeles o más. Si un bloque de 16 x 16 píxeles está dividido, el código de QP_DELTA acompaña el bloque superior izquierdo de 8 x 8 píxeles, es decir, el primer bloque subdividido del bloque de 16 x 16 píxeles.

45 50 55 60 La presente realización a modo de ejemplo puede proporcionar de este modo los mismos efectos que los de la primera realización a modo de ejemplo, mientras que los procedimientos y las estructuras de los datos tienen diferentes conceptos. En otra posible configuración, no se codifica ningún parámetro de calidad de la imagen a menos que los coeficientes de cuantificación tengan un valor distinto de cero. Si un bloque superior izquierdo en un bloque objetivo no incluye un coeficiente de cuantificación de un valor distinto de cero, un parámetro de calidad de la imagen se adjunta a un primer bloque de división estructuralmente posterior en el que un coeficiente de cuantificación tiene un valor distinto de cero. En la etapa S301 de la presente realización a modo de ejemplo, la CPU 901 calcula y obtiene la información de estado de codificación de los parámetros de calidad de la imagen a partir del tamaño y la posición del bloque objetivo a codificar. No obstante, esto no es restrictivo. Por ejemplo, puede estar dispuesto un indicador que indique si un parámetro de calidad de la imagen está codificado en un tamaño de bloque mínimo de control de la calidad de la imagen. El indicador es restablecido cada vez que se lleva a cabo el procesamiento de división de un bloque mínimo de control de la calidad de la imagen, y es establecido cada vez que se codifica un parámetro de calidad de la imagen. Para obtener la información del estado de codificación de los parámetros de calidad de la imagen, la CPU 901 hace referencia al indicador. Si el indicador está restablecido, la CPU 901 obtiene la información de que ningún parámetro de calidad de la imagen está codificado. Si el indicador está establecido, la CPU 901 obtiene la información de que un parámetro de calidad de la imagen está codificado.

65 Se describirá un procedimiento para decodificar un parámetro de calidad de la imagen según una cuarta realización a modo de ejemplo de la presente invención haciendo referencia a la figura 4. A menos que se especifique lo contrario, la configuración para llevar a cabo el procedimiento de decodificación y el funcionamiento de las etapas son los mismos que en la segunda realización a modo de ejemplo. En la presente realización a modo de ejemplo, un bloque que incluye un bloque objetivo y tiene el mismo tamaño que un tamaño de bloque mínimo de control de la

calidad de la imagen se definirá como un bloque mínimo de control de la calidad de la imagen. La información que indica si un parámetro de calidad de la imagen está decodificado en un bloque mínimo de control de la calidad de la imagen se definirá como la información del estado de decodificación de los parámetros de calidad de la imagen.

- 5 La figura 4 muestra un procedimiento para decodificar un parámetro de calidad de la imagen. La figura 4 es un diagrama de flujo detallado del procedimiento de decodificación de los parámetros de calidad de la imagen que se llevará a cabo en la etapa S1802 de la figura 18. Este procedimiento se lleva a cabo en lugar del procedimiento de decodificación de parámetros de calidad de la imagen mostrado en la figura 2. En la etapa S401, la CPU 901 obtiene la información del estado de decodificación de los parámetros de calidad de la imagen. En la etapa S402, la CPU 901 hace referencia a la información del estado de decodificación del parámetro de calidad de la imagen y determina si un parámetro de calidad de la imagen ya ha sido decodificado en un bloque mínimo de control de la calidad de la imagen.
- 10 15 El procedimiento de las etapas se describirá a continuación. En la etapa S204, la CPU 901 determina el valor del indicador de división de bloque. Si el valor del indicador de división de bloque es "0", lo que indica que el bloque no está dividido (NO, en la etapa S204), la CPU 901 lleva a cabo la etapa S205 como la segunda realización a modo de ejemplo. Si el valor del indicador de división de bloque es "1", lo que indica que el bloque está dividido (Sí, en la etapa S204), la CPU 901 finaliza el procesamiento de decodificación de los parámetros de calidad de la imagen de manera diferente a la segunda realización a modo de ejemplo.
- 20 25 En la etapa S205, como la segunda realización a modo de ejemplo, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. Si se determina que el tamaño del bloque objetivo es mayor o igual (Sí, en la etapa S205), la CPU 901 continúa con la etapa S207. En la etapa S207, la CPU 901 decodifica un parámetro de calidad de la imagen como la segunda realización a modo de ejemplo. Si no (NO, en la etapa S205), la CPU 901 lleva a cabo las etapas S401 y S402.
- 30 35 En la etapa S401, la CPU 901 obtiene la información del estado de decodificación del parámetro de calidad de la imagen. En la presente realización a modo de ejemplo, el tamaño de bloque mínimo de control de la calidad de la imagen es 16. Si el bloque objetivo es un bloque superior izquierdo de 8 x 8 píxeles, se deduce que la CPU 901 obtiene la información de que ningún parámetro de calidad de la imagen está decodificado. Si el bloque objetivo es un bloque superior derecho, inferior izquierdo o inferior derecho de 8 x 8 píxeles, un parámetro de calidad de la imagen ya ha sido decodificado en un bloque superior izquierdo de 8 x 8 píxeles. En tal caso, la CPU 901 obtiene la información de que un parámetro de calidad de la imagen está decodificado.
- 40 45 50 55 60 65 En la etapa S402, la CPU 901 hace referencia a la información del estado de decodificación del parámetro de calidad de la imagen y determina si un parámetro de calidad de la imagen ya ha sido decodificado en el bloque mínimo de control de la calidad de la imagen. Si no se ha decodificado ningún parámetro de calidad de la imagen (NO, en la etapa S402), la CPU 901 lleva a cabo la etapa S207. Si se ha decodificado un parámetro de calidad de la imagen (Sí, en la etapa S402), la CPU 901 finaliza el procesamiento de decodificación de los parámetros de calidad de la imagen.
- En la presente realización a modo de ejemplo, decodificar o no realmente un parámetro de calidad de la imagen se controla en base a la condición para comparar un tamaño del bloque objetivo en el que se codifica el parámetro de calidad de la imagen y el tamaño de bloque mínimo de control de la calidad de la imagen, y a una condición de determinación para determinar si el parámetro de calidad de la imagen ya ha sido decodificado en el bloque mínimo de control de la calidad de la imagen. Dicho modo permite la decodificación de un flujo generado por el procedimiento de codificación descrito en la tercera realización a modo de ejemplo. El procedimiento de codificación puede emplear una configuración tal que no se codifique ningún parámetro de calidad de la imagen a menos que los coeficientes de cuantificación tengan un valor distinto de cero. En tal caso, el procedimiento de decodificación puede emplear una configuración similar. Si un bloque superior izquierdo en un bloque objetivo no incluye ningún coeficiente de cuantificación de un valor distinto de cero, se decodifica un parámetro de calidad de la imagen en un primer bloque de división estructuralmente posterior en el que un coeficiente de cuantificación tiene un valor distinto de cero.
- En la etapa S401 de la figura 4 según la presente realización a modo de ejemplo, la CPU 901 calcula y obtiene la información de estado de la decodificación de los parámetros de calidad de la imagen a partir del tamaño y la posición del bloque objetivo a decodificar. No obstante, esto no es restrictivo. Por ejemplo, puede estar dispuesto un indicador que indique si un parámetro de calidad de la imagen está decodificado dentro de un tamaño de bloque mínimo de control de la calidad de la imagen. El indicador se restablece cada vez que se lleva a cabo el procesamiento de dividir un bloque mínimo de control de la calidad de la imagen, y se establece cada vez que se decodifica un parámetro de calidad de la imagen. Para obtener la información del estado de la decodificación de los parámetros de calidad de la imagen, la CPU 901 hace referencia al indicador. Si el indicador se ha restablecido, la CPU 901 obtiene la información de que ningún parámetro de calidad de la imagen está decodificado. Si el indicador se ha establecido, la CPU 901 obtiene la información de que un parámetro de calidad de la imagen está decodificado.

Se describirá un procedimiento para codificar datos residuales en un bloque haciendo referencia a la figura 5. Este procedimiento pertenece a un procedimiento de codificación de un parámetro de calidad de la imagen según una quinta realización a modo de ejemplo de la presente invención. A menos que se especifique lo contrario, la configuración para llevar a cabo el procedimiento de codificación y la operación de las etapas es la misma que en la primera realización a modo de ejemplo.

La figura 5 muestra un procedimiento para codificar un parámetro de calidad de la imagen y codificar datos residuales. El procedimiento implementa el procesamiento mostrado en las figuras 17 y 1. Este procedimiento se llama desde la etapa S1306 de la figura 13. En la etapa S501, la CPU 901 obtiene un tamaño del bloque objetivo. En la etapa S502, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S503, la CPU 901 calcula los costes cuando el bloque objetivo se divide y cuando no, respectivamente. En la etapa S504, la CPU 901 determina si dividir el bloque objetivo. En la etapa S505, la CPU 901 envía un indicador de división de bloque establecido como falso. En la etapa S506, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S507, la CPU 901 calcula un parámetro de calidad de la imagen. En la etapa S508, la CPU 901 lleva a cabo un procedimiento para codificar el parámetro de calidad de la imagen. En la etapa S509, la CPU 901 lleva a cabo la predicción y lleva a cabo una transformación ortogonal, una cuantificación y una codificación de longitud variable. En la etapa S510, la CPU 901 envía un indicador de división de bloque establecido como verdadero. En la etapa S511, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S512, la CPU 901 calcula un parámetro de calidad de la imagen. En la etapa S513, la CPU 901 codifica el parámetro de calidad de la imagen. En la etapa S514, la CPU 901 procesa los bloques de división incluidos.

El procedimiento de procesamiento se describirá concretamente a continuación. En la etapa S501, la CPU 901 obtiene un tamaño del bloque objetivo. Inmediatamente después de que se llame al presente procedimiento desde el procedimiento de la figura 13, el bloque objetivo a procesar es un bloque de retícula en sí mismo. En la presente realización a modo de ejemplo, el valor del tamaño del bloque objetivo es de 64, aunque no está limitado a este. A medida que el nivel de la estructura de árbol cuaternario de la región desciende posteriormente debido a llamadas recursivas, el valor del tamaño cambia a 32, 16 y 8. Se debe observar que el número de niveles no está limitado a los mismos.

En la etapa S502, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen que se establece desde el exterior. En la presente realización a modo de ejemplo, el valor del tamaño de bloque mínimo de control de la calidad de la imagen es de 16, aunque no está limitado a este.

En la etapa S503, la CPU 901 calcula los costes cuando el bloque se divide y cuando no, respectivamente. Los costes se calculan utilizando los multiplicadores de Lagrange mencionados anteriormente, así como las características de la imagen, las estadísticas y las longitudes de código predichas.

En la etapa S504, la CPU 901 compara los costes con y sin división, calculados en la etapa S503. Si el coste con división de bloque es menor que el coste sin división de bloque (Sí, en la etapa S504), la CPU 901 lleva a cabo las etapas S510 a S514. Si no (NO, en la etapa S504), la CPU 901 lleva a cabo las etapas S505 a S509.

En la etapa S505, la CPU 901 envía un indicador de división de bloque establecido como falso. En la etapa S506, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. Si el tamaño del bloque objetivo es mayor o igual (Sí, en la etapa S506), la CPU 901 lleva a cabo las etapas S507 a S509. Si no (NO, en la etapa S506), la CPU 901 lleva a cabo la etapa S509 sin llevar a cabo las etapas S507 y S508.

En la etapa S507, la CPU 901 lleva a cabo el control de velocidad en el bloque objetivo para calcular un parámetro de calidad de la imagen. En la etapa S508, la CPU 901 codifica el parámetro de calidad de la imagen calculado en la etapa S507. En la presente realización a modo de ejemplo, la CPU 901 determina un valor de diferencia entre el parámetro de calidad de la imagen utilizado en el bloque anterior y el parámetro de calidad de la imagen calculado en la etapa S507 como un valor de QP_DELTA. La CPU 901 codifica el valor de QP_DELTA determinado mediante codificación de Golomb en el código de QP_DELTA, que es el código del parámetro de calidad de la imagen.

En la etapa S509, la CPU 901 lleva a cabo una transformación ortogonal, una cuantificación y una codificación de longitud variable en datos residuales para generar datos de coeficientes codificados. La CPU 901 lleva a cabo la cuantificación utilizando el parámetro de calidad de la imagen calculado en la etapa S507. Si en la etapa S506 se determina que el tamaño del bloque objetivo es menor que el tamaño de bloque mínimo de control de la calidad de la imagen (NO, en la etapa S506), la CPU 901 utiliza el mismo parámetro de calidad de la imagen que el utilizado para la cuantificación previa.

En la etapa S510, la CPU 901 codifica un indicador de división de bloque establecido como verdadero. En la etapa S511, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. Si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad

de la imagen (Sí, en la etapa S511), la CPU 901 lleva a cabo las etapas S512 a S514. Si no (NO, en la etapa S511), la CPU 901 lleva a cabo la etapa S514 sin llevar a cabo las etapas S512 y S513.

- 5 En la etapa S512, la CPU 901 lleva a cabo el procesamiento de control de la velocidad para calcular un parámetro de calidad de la imagen con el que codificar el bloque objetivo. Todos los bloques de división incluidos están sujetos al control de velocidad. Los bloques de división incluidos están codificados con el mismo parámetro de calidad de la imagen.
- 10 En la etapa S513, la CPU 901 lleva a cabo la misma operación que en la etapa S508 para codificar el parámetro de calidad de la imagen calculado en la etapa S512. En la etapa S514, la CPU 901 lleva a cabo recursivamente el presente procedimiento en cuatro bloques de división (superior izquierdo, superior derecho, inferior izquierdo e inferior derecho) que están incluidos en el nivel jerárquicamente subsiguiente. Cada uno de los bloques de división tiene un tamaño de 1/2 del bloque objetivo original tanto en altura como en anchura. El tamaño del bloque objetivo disminuye división a división. Dicho procesamiento recursivo se puede llevar a cabo hasta un tamaño de bloque tan pequeño como de 8 x 8 píxeles. Mediante el procesamiento recursivo, el código de indicador de división, el código de QP_DELTA y los datos de coeficiente en los bloques de división respectivos se envían como una secuencia en un orden estructural, tal como se muestra en la figura 16.
- 15 Tal como se describió anteriormente, el procesamiento de control de la velocidad y el procesamiento de codificación de los parámetros de calidad de la imagen pueden integrarse uno con otro. Incluso, dicha configuración permite el control de la calidad de la imagen en unidades más finas como el procedimiento de codificación de un parámetro de calidad de la imagen descrito en la tercera realización a modo de ejemplo.
- 20 Un procedimiento para decodificar un parámetro de calidad de la imagen según una sexta realización a modo de ejemplo de la presente invención se describirá haciendo referencia a la figura 6.
- 25 La figura 6 muestra un procedimiento para decodificar un parámetro de calidad de la imagen. Este procedimiento se aplica a la etapa S1405 de la figura 14. En la etapa S601, la CPU 901 obtiene el tamaño de un bloque objetivo a decodificar. En la etapa S602, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S603, la CPU 901 decodifica un indicador de división de bloque. En la etapa S604, la CPU 901 determina si el bloque objetivo a decodificar está dividido. En la etapa S605, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S606, la CPU 901 decodifica un parámetro de calidad de la imagen. En la etapa S607, la CPU 901 lleva a cabo una decodificación de longitud variable, una cuantificación inversa y una transformación ortogonal inversa. En la etapa S608, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S609, la CPU 901 decodifica un parámetro de calidad de la imagen. En la etapa S610, la CPU 901 llama de manera recursiva al presente procedimiento de decodificación, dirigido a bloques de división incluidos.
- 30 El procedimiento de las etapas se describirá concretamente a continuación. En la etapa S601, la CPU 901 obtiene el tamaño de un bloque objetivo para decodificar. Inmediatamente después de que se llame al presente procedimiento desde el procedimiento de la figura 14, el bloque objetivo a procesar es un bloque de retícula en sí mismo. En la presente realización a modo de ejemplo, un bloque de retícula tiene un valor de tamaño de 64, aunque no está limitado a este. A medida que el nivel de la estructura de árbol cuaternario de la región desciende posteriormente debido a llamadas recursivas, el valor del tamaño cambia a 32, 16 y 8. Se debe tener en cuenta que el número de niveles no está limitado a estos.
- 35 En la etapa S602, la CPU 901 obtiene un tamaño de bloque mínimo de control de la calidad de la imagen. En la presente realización a modo de ejemplo, la CPU 901 obtiene el tamaño de bloque mínimo de control de la calidad de la imagen que está incrustado en la información de la cabecera del fotograma.
- 40 En la etapa S603, la CPU 901 obtiene el estado de división del bloque. En la presente realización a modo de ejemplo, la CPU 901 hace referencia al valor del indicador de división de bloque que es decodificado en la etapa S1801 de la figura 18.
- 45 En la etapa S604, la CPU 901 determina si el valor del indicador del bloque división es verdadero o falso. Si es falso (NO, en la etapa S604), la CPU 901 lleva a cabo las etapas S605 a S607. Si es verdadero (Sí, en la etapa S604), la CPU 901 lleva a cabo las etapas S608 a S610.
- 50 En la etapa S605, la CPU 901 determina si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen. Si el tamaño del bloque objetivo es mayor o igual que el tamaño de bloque mínimo de control de la calidad de la imagen (Sí, en la etapa S605), la CPU 901 lleva a cabo las etapas S606 y S607. Si no (NO, en la etapa S605), la CPU 901 lleva a cabo la etapa S607 sin llevar a cabo la etapa S606.
- 55 En la etapa S606, la CPU 901 decodifica el código del parámetro de calidad de la imagen. En la presente realización a modo de ejemplo, la CPU 901 decodifica el código de QP_DELTA en un valor de QP_DELTA mediante la

decodificación de Golomb. A continuación, la CPU 901 agrega el valor de QP_DELTA a un parámetro de calidad de la imagen que se ha utilizado en la cuantificación inversa anterior, generando de este modo un parámetro de calidad de la imagen.

5 En la etapa S607, la CPU 901 lleva a cabo una decodificación de longitud variable, una cuantificación inversa y una transformación ortogonal inversa. Para la cuantificación inversa, la CPU 901 utiliza el parámetro de calidad de la imagen decodificado en la etapa S606. Si el tamaño del bloque objetivo es menor que el tamaño de bloque mínimo de control de la calidad de la imagen (NO, en la etapa S608), la CPU 901 utiliza el parámetro de calidad de la imagen que se ha utilizado para cuantificar inversamente el bloque objetivo anterior. Posteriormente, la CPU 901 finaliza la decodificación del bloque objetivo.

10 En la etapa S608, la CPU 901 determina si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen. Si el tamaño del bloque objetivo es igual al tamaño de bloque mínimo de control de la calidad de la imagen (SI, en la etapa S608), la CPU 901 lleva a cabo la etapa S609. Si no (NO, en la etapa S608), la CPU 901 continúa con la etapa S610 sin llevar a cabo la etapa S609.

15 En la etapa S609, la CPU 901 decodifica el código del parámetro de calidad de la imagen como en la etapa S606. En la etapa S610, la CPU 901 lleva a cabo recursivamente el presente procedimiento en cuatro bloques de división (superior izquierdo, superior derecho, inferior izquierdo, superior derecho) que están incluidos en el nivel jerárquicamente posterior. Cada uno de los bloques de división tiene un tamaño de 1/2 del bloque objetivo original tanto en altura como en anchura. El tamaño del bloque objetivo disminuye de división a división. Mediante dicho procesamiento recursivo, la CPU 901 decodifica el código de indicador de división, el código de QP_DELTA y los datos de los coeficientes en cada bloque de división en el orden de la estructura de árbol cuaternario de la región tal como se muestra en la figura 16.

20 25 Tal como se describió anteriormente, el procesamiento recursivo para escanear la estructura de árbol cuaternario de la región y el procesamiento de decodificación de los parámetros de calidad de la imagen se puede integrar uno con otro. Incluso, dicha configuración puede decodificar una secuencia codificada mediante el procedimiento de codificación según la primera realización a modo de ejemplo o la quinta realización a modo de ejemplo.

30 Una séptima realización a modo de ejemplo de la presente invención trata con un procedimiento de codificación de bloque de división. El procedimiento de codificación de bloque de división según la presente realización a modo de ejemplo se implementa utilizando la configuración mostrada en la figura 9.

35 40 45 Antes de la codificación, la CPU 901 establece un perfil y un nivel. Los perfiles describen las características de un flujo que se generará en la presente realización a modo de ejemplo. Los niveles son un conjunto de parámetros que están permitidos según el rendimiento del procesamiento del decodificador. Ya que el rendimiento del procesamiento depende principalmente del tamaño de la imagen, los niveles están definidos para los tamaños de imagen respectivos. No obstante, las unidades de definición de nivel no están limitadas a los tamaños de imagen. Por ejemplo, los niveles se pueden establecer según el número de bloques de retícula a procesar, porque el rendimiento de procesamiento depende del número. En la presente realización a modo de ejemplo, los tamaños de bloque mínimo de control de la calidad de la imagen se establecen en un valor mínimo en cada nivel. Las figuras 21A y 21B muestran tablas que definen los tamaños mínimos de bloque de control de la calidad de la imagen limitados por perfil y por nivel.

50 55 60 65 La presente realización a modo de ejemplo trata un caso en el que los perfiles incluyen un perfil de referencia que utiliza solo herramientas básicas y un perfil principal que utiliza otras herramientas. No obstante, los perfiles no están limitados a los mismos.

50 55 60 65 La figura 21A muestra una tabla que muestra los niveles y los límites de los tamaños mínimos de bloque de control de la calidad de la imagen en el perfil de referencia. La figura 21B muestra una tabla que muestra los niveles y los límites del tamaño de bloque mínimo de control de la calidad de la imagen en el perfil principal.

50 55 60 65 En la presente realización a modo de ejemplo, los niveles se clasifican por el número máximo de bloques de retícula (64 x 64 píxeles) incluidos en un fotograma. Específicamente, el nivel 1 está destinado a imágenes relativamente pequeñas, tal como de 640 x 480 píxeles. El nivel 2 está destinado a imágenes de tamaño medio de 1920 x 1080 píxeles. El nivel 3 está destinado a imágenes grandes de 4096 x 2160 píxeles.

50 55 60 65 Se describirá el perfil de referencia mostrado en la figura 21A. El nivel 1 muestra que el valor mínimo de un tamaño de bloque mínimo de control de la calidad de la imagen es 8 y los tamaños mínimos de bloque de control de la calidad de la imagen se pueden establecer en el intervalo de 8 a 64. El nivel 2 muestra que el valor mínimo de un tamaño de bloque mínimo de control de la calidad de la imagen es 16 y los tamaños mínimos de bloque de control de la calidad de la imagen se pueden establecer en el intervalo de 16 a 64. El nivel 3 muestra que el valor mínimo de un tamaño de bloque mínimo de control de la calidad de la imagen es 32 y los tamaños mínimos de bloque de control de la calidad de la imagen se pueden establecer en el intervalo de 32 a 64. El perfil principal mostrado en la

figura 21B muestra que los tamaños mínimos de bloque de control de la calidad de la imagen están controlados en el intervalo de 8 a 64, independientemente del nivel.

5 La CPU 901 determina el perfil y el nivel dependiendo de las condiciones de un procedimiento o aparato de codificación, tales como herramientas y capacidades de memoria, y/o considerando las condiciones de un procedimiento o aparato de decodificación. No obstante, el procedimiento para determinar el perfil y el nivel no está limitado especialmente. Un usuario no mostrado puede establecer un perfil y un nivel. El perfil y nivel determinados pueden codificarse, incluirse en una cabecera de secuencia que describe la información sobre una imagen en movimiento completa, y enviarse desde la interfaz de comunicación 905. Dicho perfil y nivel pueden estar grabados en el HDD 903.

10 A continuación, se describirá un procedimiento para codificar fotogramas. En la etapa S1301 de la figura 13, como la primera realización a modo de ejemplo, la CPU 901 establece inicialmente un parámetro de un tamaño de bloque mínimo de control de la calidad de la imagen. La figura 22 es un diagrama de flujo que muestra un procedimiento para calcular un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S2201, la CPU 901 determina un tipo de perfil. Si el tipo de perfil es un perfil de referencia (PERFIL DE REFERENCIA en la etapa S2202), la CPU 901 lleva a cabo la etapa S2204 y las etapas posteriores. Si el tipo de perfil es un perfil principal (PERFIL PRINCIPAL en la etapa S2202), la CPU 901 lleva a cabo la etapa S2203.

15 20 El caso con el perfil principal se describirá en primer lugar. En la etapa S2203, la CPU 901 establece el tamaño de bloque mínimo de control de la calidad de la imagen en 8. A continuación, la CPU 901 finaliza el procesamiento.

25 A continuación, se describirá el caso con el perfil de referencia. En la etapa S2204, la CPU 901 obtiene un valor de nivel. En la presente realización a modo de ejemplo, el valor del nivel se calcula según un tamaño de imagen de entrada. El valor del nivel indica un nivel mínimo en el que se puede decodificar una secuencia del tamaño de la imagen de entrada.

30 35 En la etapa S2205, la CPU 901 determina si el valor de nivel es 1. Si la determinación es verdadero (Sí, en la etapa S2205), la CPU 901 lleva a cabo la etapa S2206. Si la determinación es falso (NO, en la etapa S2205), la CPU 901 lleva a cabo la etapa S2207. En la etapa S2206, la CPU 901 establece el tamaño de bloque mínimo de control de la calidad de la imagen en 8. En la etapa S2207, la CPU 901 determina si el valor del nivel es 2. Si la determinación es verdadero (Sí, en la etapa S2207), la CPU 901 lleva a cabo la etapa S2208. Si la determinación es falso (NO, en la etapa S2207), la CPU 901 lleva a cabo la etapa S2209. En la etapa S2208, la CPU 901 establece el tamaño de bloque mínimo de control de la calidad de la imagen en 16. En la etapa S2209, la CPU 901 establece el tamaño de bloque mínimo de control de la calidad de la imagen a 32.

40 45 A continuación, la CPU 901 vuelve al proceso mostrado en la figura 13. En la etapa S1302, como la primera realización a modo de ejemplo, la CPU 901 codifica el tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S1303, la CPU 901 genera la información de la cabecera de fotograma que incluye los datos codificados. La figura 23A muestra el flujo de bits generado. Una cabecera de secuencia incluye datos codificados en un perfil y un nivel, seguidos de códigos que indican el tamaño de la imagen. Una cabecera de fotograma en cada pieza de datos de fotograma incluye el código de tamaño de bloque mínimo de control de la calidad de la imagen, con una configuración de tamaño mayor o igual al valor mínimo de un tamaño de bloque mínimo de control de la calidad de la imagen definido en la figura 21A.

50 55 60 La figura 24 es un diagrama de flujo que muestra un procedimiento completo. En la etapa S2401, la CPU 901 introduce el tamaño de una imagen a codificar. En la etapa S2402, la CPU 901 selecciona un perfil. La CPU 901 también selecciona un nivel según la figura 21A o 21B. En la etapa S2403, la CPU 901 obtiene el valor mínimo de un tamaño de bloque mínimo de control de la calidad de la imagen según el nivel seleccionado. En la etapa S2404, la CPU 901 codifica el perfil y nivel seleccionados. En la etapa S2405, la CPU 901 genera una cabecera de secuencia que incluye los datos codificados en el perfil y el nivel. En la etapa S2406, la CPU 901 lleva a cabo la codificación en unidades de fotogramas. Los tamaños posibles de los bloques mínimos de control de la calidad de la imagen en cada fotograma son mayores o iguales que el valor mínimo anterior de un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S2407, la CPU 901 determina si todos los fotogramas están codificados. Si no todos los fotogramas están codificados (NO, en la etapa S2407), la CPU 901 continúa con la etapa S2406 para codificar un fotograma siguiente. Si todos los fotogramas están codificados (Sí, en la etapa S2407), la CPU 901 finaliza el proceso de codificación.

65 Proporcionar perfiles y niveles y la limitación de los tamaños mínimos de bloque de control de la calidad de la imagen permite la determinación de tamaños mínimos de bloque de control de la calidad de la imagen dependiendo del tamaño de la imagen. Por lo tanto, es posible codificar una imagen, aunque se omiten las operaciones en bloques más pequeños sin causar una disminución en la calidad de la imagen. Esto permite diseñar un circuito óptimo para un aparato deseado y diseñar una configuración para ciertos consumos de energía suprimidos. Por ejemplo, un perfil de referencia se puede utilizar para comunicaciones móviles, de modo que se pueden imponer restricciones de carga severas en las operaciones de codificación a la vista del tiempo de funcionamiento de la batería. Se puede utilizar un perfil principal para aparatos estacionarios sin batería, por lo que se imponen

restrictiones más flexibles sobre la carga de operación. En términos de software, el tiempo de operación para llevar a cabo el control de cuantificación en bloques más pequeños que los tamaños mínimos de bloque de control de la calidad de la imagen se puede omitir para agilizar.

5 Se comprenderá que un procedimiento de codificación basado en perfil y nivel, tal como la presente realización a modo de ejemplo, se puede aplicar a la tercera y quinta realizaciones a modo de ejemplo.

10 Los valores mínimos de un tamaño de bloque mínimo de control de la calidad de la imagen determinado para los niveles respectivos, mostrados en las figuras 21A y 21B, siempre se pueden utilizar en los niveles respectivos. En tal caso, se puede omitir el código de tamaño de bloque mínimo de control de la calidad de la imagen que se incluirá en cada fotograma. La figura 23B muestra dicho flujo de bits. El flujo de bits mostrado en la figura 23B difiere del mostrado en la figura 23A en que las cabeceras de fotograma no incluyen el código de tamaño de bloque mínimo de control de la calidad de la imagen. Los tamaños de bloque mínimo de control de la calidad de la imagen en la secuencia se fijan en el valor mínimo de un tamaño de bloque mínimo de control de la calidad de la imagen que está definido para el perfil y el nivel. Esto proporciona el efecto de eliminar el código redundante para mejorar la eficiencia de la codificación.

15 Las capacidades de memoria para almacenar matrices de cuantificación más pequeñas que el tamaño de bloque mínimo de control de la calidad de la imagen y el procesamiento para actualizar dichas matrices de cuantificación también pueden estar omitidos.

20 Una octava realización a modo de ejemplo de la presente invención trata un procedimiento de decodificación de imagen para decodificar un flujo de bits que se genera en la séptima realización a modo de ejemplo. El procedimiento de decodificación de imagen según la presente realización a modo de ejemplo se implementa utilizando la configuración mostrada en la figura 9.

25 Antes de decodificar fotogramas, la CPU 901 introduce y decodifica una cabecera de secuencia. La CPU 901 decodifica el código relacionado con un perfil y un nivel según la figura 23A. La CPU 901 también decodifica el código de la anchura de la imagen y el código de la altura de imagen. Si el perfil decodificado es un perfil de referencia, la CPU 901 hace referencia a la tabla de la figura 21A. La CPU 901 selecciona y determina adicionalmente el valor mínimo de un tamaño de bloque mínimo de control de la calidad de la imagen correspondiente al nivel. Si el perfil decodificado es un perfil principal, la CPU 901 hace referencia a la tabla de la figura 21B y establece el valor mínimo de un tamaño de bloque mínimo de control de la calidad de la imagen en 8.

30 35 En este momento, la CPU 901 reserva memorias intermedias y otras memorias necesarias en la RAM 902. Por ejemplo, con un perfil de referencia en el nivel 2, no se llevará a cabo ningún control de la calidad de la imagen en un tamaño de bloque de 8 x 8 píxeles. En tal caso, se pueden omitir áreas de memoria para almacenar parámetros de calidad de la imagen en unidades de bloques de 8x8 y 4x4 píxeles.

40 45 La CPU 901 decodifica los fotogramas según el diagrama de flujo de la figura 14, que se ha descrito en la segunda realización a modo de ejemplo. En la etapa S1402, la CPU 901 decodifica un tamaño de bloque mínimo de control de la calidad de la imagen incluido en una cabecera de fotograma. En la etapa S1405, la CPU 901 lleva a cabo la decodificación jerárquica con un bloque de retícula como un bloque objetivo. En la etapa S1805 de la figura 18 descrita en la segunda realización a modo de ejemplo, la CPU 901 lleva a cabo el procesamiento recursivo según el nivel jerárquico, decodificando de este modo los parámetros de control de la calidad de la imagen hasta bloques del tamaño de bloque mínimo de control de la calidad de la imagen.

50 55 La figura 25 es un diagrama de flujo que muestra un procedimiento completo. En la etapa S2501, la CPU 901 introduce una cabecera de secuencia. En la etapa S2502, la CPU 901 decodifica datos codificados en un perfil y un nivel, obteniendo de este modo el perfil y el nivel. En la etapa S2503, la CPU 901 obtiene el valor mínimo de un tamaño de bloque mínimo de control de la calidad de la imagen según el nivel seleccionado. En la etapa S2504, la CPU 901 lleva a cabo la decodificación en unidades de fotogramas. Esto garantiza que los posibles tamaños de bloques mínimos de control de la calidad de la imagen en cada fotograma sean mayores o iguales al valor mínimo anterior de un tamaño de bloque mínimo de control de la calidad de la imagen. En la etapa S2505, la CPU 901 determina si todos los fotogramas están decodificados. Si no todos los fotogramas están codificados (NO, en la etapa S2505), la CPU 901 continúa con la etapa S2504 para decodificar un fotograma siguiente. Si se han decodificado todos los fotogramas (SÍ, en la etapa S2505), la CPU 901 finaliza el proceso de decodificación.

60 65 Proporcionar perfiles y niveles y la limitación de los tamaños mínimos de bloque de control de la calidad de la imagen permite la determinación de tamaños mínimos de bloque de control de la calidad de la imagen dependiendo del tamaño de la imagen. Por lo tanto, es posible decodificar una imagen, aunque se omiten las operaciones en bloques pequeños sin causar una disminución en la calidad de la imagen. La determinación de los tamaños mínimos de bloque de control de la calidad de la imagen puede reducir el número de veces que se lleva a cabo un procesamiento recursivo, por lo que se pueden suprimir las variaciones en el tiempo para llevar a cabo la decodificación en unidades de bloques de retícula.

En consecuencia, es posible diseñar un circuito óptimo para un aparato previsto y diseñar una configuración para suprimir cierto consumo de energía. Por ejemplo, un perfil de referencia se puede utilizar para comunicaciones móviles, de modo que se pueden imponer restricciones de carga severas en las operaciones de codificación a la vista del tiempo de funcionamiento de la batería. Se puede utilizar un perfil principal para aparatos estacionarios sin

5 batería, por lo que se imponen restricciones más flexibles sobre la carga de operación. En términos de software, el tiempo de operación para llevar a cabo el control de cuantificación en bloques más pequeños que los tamaños mínimos de bloque de control de la calidad de la imagen se puede omitir para agilizar.

10 Se comprenderá que la CPU 901 puede decodificar código relacionado con el perfil y el nivel según la figura 23B. Como en la descripción anterior, la CPU 901 introduce y decodifica una cabecera de secuencia. Según el perfil y el nivel resultantes, la CPU 901 determina un tamaño de bloque mínimo de control de la calidad de la imagen, tal como se ha descrito anteriormente. Cuando se decodifican fotogramas, la CPU 901 omite la etapa S1402 y utiliza el tamaño de bloque mínimo de control de la calidad de la imagen.

15 La determinación única de un tamaño de bloque mínimo de control de la calidad de la imagen según el perfil y el nivel hace posible la decodificación de imagen sin el código de tamaño de bloque mínimo de control de la calidad de la imagen en cada cabecera de fotograma.

20 La CPU 901 puede comparar un tamaño de bloque mínimo de control de la calidad de la imagen decodificado con un tamaño de bloque mínimo de control de la calidad de la imagen determinado por el perfil y el nivel. Si el último es mayor que el primero, la CPU 901 puede detectar el flujo de bits en busca de un error.

REIVINDICACIONES

1. Aparato de codificación de imagen, configurado para dividir una imagen en bloques y codificar la imagen en unidades de bloques, comprendiendo el aparato de codificación de imagen:
- 5 un medio de obtención (901), configurado para obtener información sobre un tamaño de bloque de un bloque a codificar y configurado para obtener información sobre un tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación;
- 10 un medio de codificación de información (901), configurado para codificar la información sobre el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación;
- 15 un medio de verificación (901), configurado para verificar si el bloque a codificar es un bloque que está dividido y, para un bloque que no está dividido, verificar si el tamaño del bloque a codificar es mayor o igual que el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación; y
- 20 un medio de codificación (901),
- 25 en el que, si el tamaño de bloque del bloque a codificar es mayor o igual que el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación (S105):
- 30 el medio de obtención (901) está configurado para obtener un primer parámetro de cuantificación para el bloque a codificar (S107) y el medio de codificación de información (901) está configurado para codificar un primer valor de diferencia entre el primer parámetro de cuantificación y un parámetro de cuantificación utilizado para codificar un bloque anterior (S108); y
- 35 el medio de codificación (901) está configurado para codificar el bloque a codificar utilizando el primer parámetro de cuantificación,
- 40 de otra manera:
- 30 el medio de obtención está configurado para obtener un segundo parámetro de cuantificación para el bloque a codificar (S107) y el medio de codificación de información está configurado para codificar un segundo valor de diferencia entre el segundo parámetro de cuantificación y un parámetro de cuantificación utilizado para codificar un bloque anterior (S108), en el caso de que (i) el segundo valor de diferencia no haya sido codificado previamente para un bloque en un grupo de bloques que se obtiene por división de un bloque que tiene un tamaño de bloque igual al tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación, y comprende el bloque a codificar (S302) y (ii) el bloque a codificar es el primer bloque en el grupo de bloques, en un orden de codificación, que incluye un coeficiente de transformación cuantificado que no es igual a cero; y
- 45 el medio de codificación (901) está configurado para codificar el bloque a codificar en el grupo de bloques utilizando el segundo parámetro de cuantificación, y para codificar un bloque posterior del bloque a codificar en el grupo de bloques, en el orden de codificación, utilizando el segundo parámetro de cuantificación en el caso de que el siguiente bloque incluya un coeficiente de transformación cuantificado que no es igual a cero,
- 50 en el que, el primer valor de diferencia y el segundo valor de diferencia están codificados mediante codificación de Golomb.
2. El aparato de codificación de imagen, según la reivindicación 1, en el que la información sobre el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación es un índice correspondiente a un tamaño de bloque.
- 55 3. El aparato de codificación de imagen, según la reivindicación 1, en el que el medio de obtención obtiene la información sobre el tamaño de bloque del bloque a codificar haciendo referencia a la información que indica si dividir o no el bloque a codificar.
- 60 4. Aparato de decodificación de imagen para descodificar datos codificados que se codifican dividiendo una imagen en bloques, comprendiendo el aparato de decodificación de imagen:
- 55 un primer medio de obtención (901), configurado para obtener un tamaño de bloque de un bloque a decodificar;
- 60 un segundo medio de obtención (901), configurado para obtener un tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación para la cuantificación inversa;
- 65 un medio de verificación (901), configurado para verificar si el bloque a decodificar es un bloque que está dividido y, para un bloque que no está dividido, verificar si el tamaño de bloque del bloque a decodificar es mayor o igual que el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación;
- 70 un medio de obtención de parámetros de cuantificación; y
- 75 un medio de decodificación,
- 80 en el que, si el tamaño de bloque del bloque a decodificar es mayor o igual que el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación:
- 85 el medio de obtención del parámetro de cuantificación está configurado para decodificar un primer valor de diferencia entre un primer parámetro de cuantificación para el bloque a decodificar y un parámetro de cuantificación

utilizado para decodificar un bloque anterior y configurado para obtener el primer parámetro de cuantificación para el bloque a decodificar utilizando el primer valor de diferencia; y el medio de decodificación está configurado para decodificar el bloque a decodificar utilizando el primer parámetro de cuantificación;

- 5 en caso contrario:
- el medio de obtención del parámetro de cuantificación está configurado para decodificar un segundo valor de diferencia entre un segundo parámetro de cuantificación para el bloque a decodificar y un parámetro de cuantificación utilizado para decodificar un bloque anterior en el caso de que (i) el segundo valor de diferencia no haya sido decodificado previamente para un bloque en un grupo de bloques que se obtiene por división de un bloque que tiene un tamaño de bloque igual al tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación, y comprende el bloque a decodificar y (ii) el bloque a decodificar es el primer bloque del grupo de bloques, en un orden de decodificación, que incluye un coeficiente de transformación cuantificado que no es igual a cero;
- 10 el medio de obtención del parámetro de cuantificación está configurado para obtener el segundo parámetro de cuantificación para el bloque a decodificar utilizando el segundo valor de diferencia; y el medio de decodificación (901) está configurado para decodificar el bloque a decodificar en el grupo de bloques utilizando el segundo parámetro de cuantificación, y para decodificar un bloque posterior del bloque a codificar en el grupo de bloques, en el orden de decodificación, utilizando el segundo parámetro de cuantificación en el caso de que el siguiente bloque incluya un coeficiente de transformación cuantificado que no es igual a cero,
- 15 en el que, datos del primer valor de diferencia y del segundo valor de diferencia son datos codificados mediante codificación de Golomb en los datos codificados.

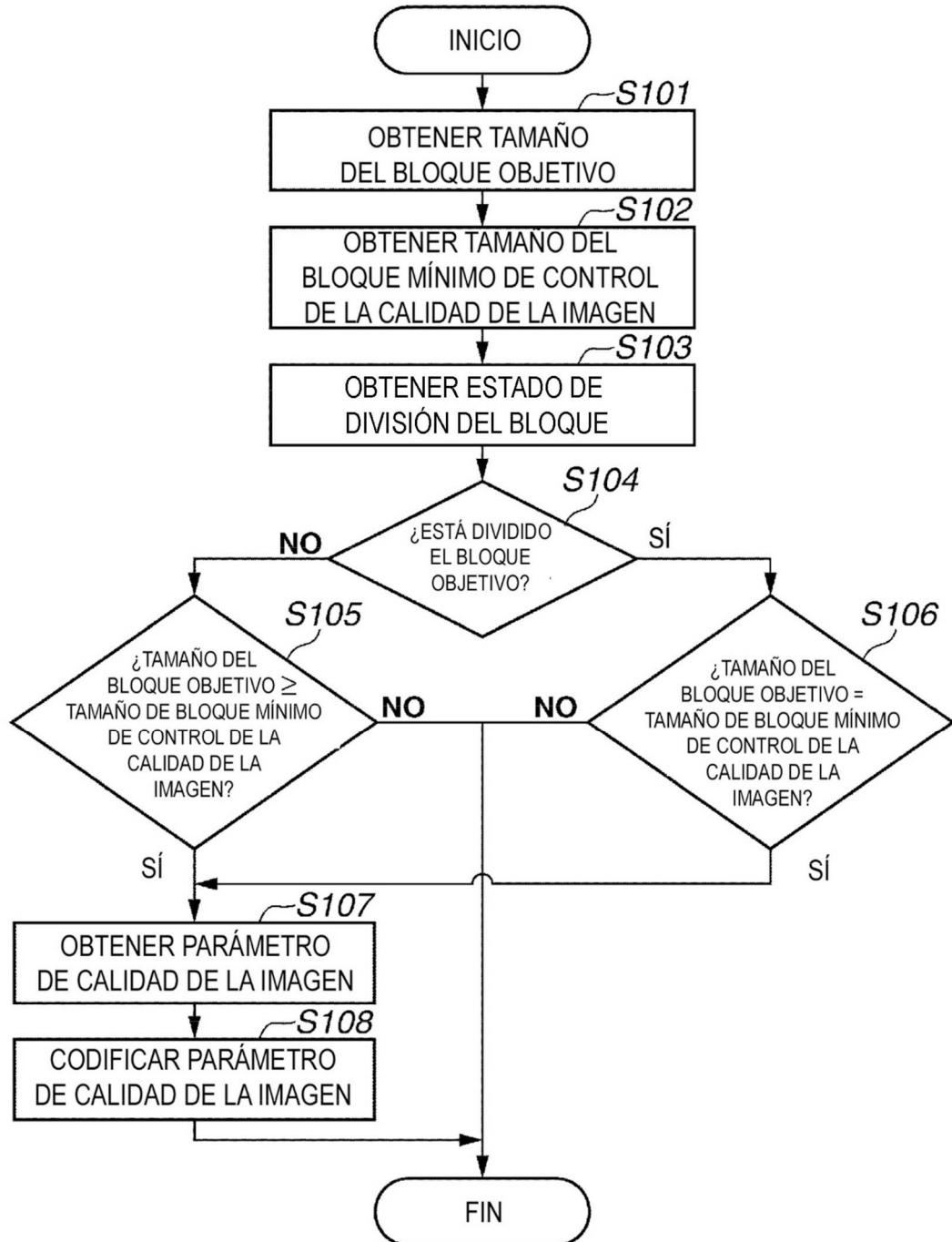
- 20 25 5. Aparato de decodificación de imagen, según la reivindicación 4, en el que la información sobre el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación es un índice correspondiente a un tamaño de bloque.
- 30 6. Aparato de decodificación de imagen, según la reivindicación 4, en el que el medio de obtención obtiene la información sobre el tamaño de bloque del bloque a decodificar haciendo referencia a la información que indica si se debe dividir o no el bloque a decodificar.
- 35 7. Procedimiento de codificación de imagen para codificar imágenes, que divide una imagen en bloques y codifica la imagen en unidades de bloques, comprendiendo el procedimiento de codificación de imagen:

- 40 obtener información sobre un tamaño de bloque de un bloque a codificar (S101) y obtener información sobre un tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación (S102); codificar información sobre el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación utilizado para controlar la cuantificación;
- 45 verificar si el bloque a codificar es un bloque que está dividido (S104) y, para un bloque que no está dividido, verificar si el tamaño del bloque del bloque a codificar es mayor o igual que el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación (S105);
- 50 55 si el tamaño de bloque del bloque a codificar es mayor o igual que el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación (S105):
- 60 obtener un primer parámetro de cuantificación para el bloque a codificar (S107) y codificar un primer valor de diferencia entre el primer parámetro de cuantificación y un parámetro de cuantificación utilizado para codificar un bloque anterior (S108); y codificar el bloque a codificar utilizando el primer parámetro de cuantificación; de lo contrario:
- 65 obtener un segundo parámetro de cuantificación para el bloque a codificar (S107) y codificar un segundo valor de diferencia entre el segundo parámetro de cuantificación y un parámetro de cuantificación utilizado para codificar un bloque anterior (S108), en el caso de que (i) el segundo valor de diferencia no haya sido previamente codificado para un bloque en un grupo de bloques que se obtiene por división de un bloque que tiene un tamaño de bloque igual al tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación y comprende el bloque a codificar (S302) y (ii) el bloque a codificar sea el primer bloque del grupo de bloques, en un orden de codificación, que incluye un coeficiente de transformación cuantificado que no es igual a cero; y
- 70 codificar el bloque a codificar en el grupo de bloques, utilizando el segundo parámetro de cuantificación, y codificar un bloque posterior del bloque a codificar en el grupo de bloques, en el orden de codificación, utilizando el segundo parámetro de cuantificación en el caso de que el siguiente bloque incluya un coeficiente de transformación cuantificado que no es igual a cero,

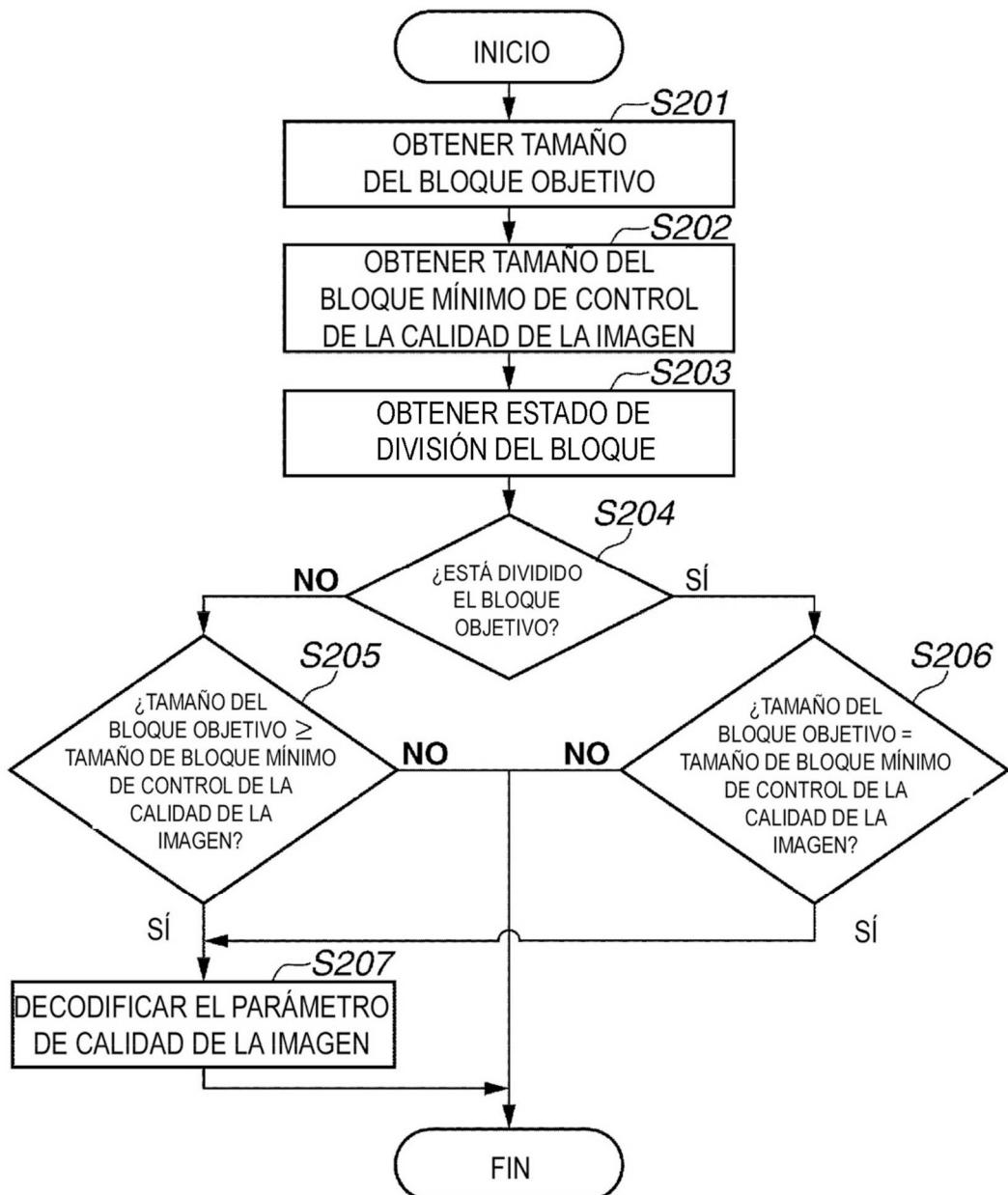
en el que, el primer valor de diferencia y el segundo valor de diferencia están codificados mediante codificación de Golomb.

- 5 8. Procedimiento de decodificación de imagen para decodificar datos que se codifican dividiendo una imagen en bloques, comprendiendo el procedimiento de codificación de imagen:
- obtener un tamaño de bloque de un bloque a decodificar (S201);
 obtener un tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación para la cuantificación inversa (S202);
 10 verificar si el bloque a decodificar es un bloque que está dividido (S204) y, para un bloque que no está dividido, verificar si el tamaño del bloque a decodificar es mayor o igual que el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación;
- 15 en el que, si el tamaño de bloque del bloque a decodificar es mayor o igual que el tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación:
- 20 decodificar un primer valor de diferencia entre un primer parámetro de cuantificación para el bloque a decodificar y un parámetro de cuantificación utilizado para decodificar un bloque anterior y obtener el primer parámetro de cuantificación para el bloque a decodificar utilizando el primer valor de diferencia; y
 decodificar el bloque a decodificar utilizando el primer parámetro de cuantificación;
 en caso contrario:
- 25 decodificar un segundo valor de diferencia entre un segundo parámetro de cuantificación para el bloque a decodificar y un parámetro de cuantificación utilizado para decodificar un bloque anterior en el caso de que (i) el segundo valor de diferencia no haya sido decodificado previamente para un bloque en un grupo de bloques que se obtiene mediante la división de un bloque que tiene un tamaño de bloque igual al tamaño de bloque mínimo utilizado para controlar un parámetro de cuantificación y comprende el bloque a decodificar y (ii) el bloque a decodificar es el primer bloque del grupo de bloques, en un orden de decodificación, que incluye un coeficiente de transformación cuantificado que no es igual a cero;
- 30 obtener el segundo parámetro de cuantificación para el bloque a decodificar utilizando el segundo valor de diferencia; y
 decodificar el bloque a decodificar en el grupo de bloques utilizando el segundo parámetro de cuantificación, y decodificar un bloque posterior del bloque a decodificar en el grupo de bloques, en el orden de decodificación, utilizando el segundo parámetro de cuantificación en el caso de que el siguiente bloque incluya un coeficiente de transformación cuantificado que no es igual a cero,
 en el que, datos del primer valor de diferencia y datos del segundo valor de diferencia son datos codificados mediante codificación de Golomb en los datos codificados.
- 40 9. Programa informático que, cuando es ejecutado en un ordenador, hace que el ordenador lleve a cabo el procedimiento de codificación de imagen de la reivindicación 7.
- 45 10. Programa informático que, cuando es ejecutado en un ordenador, hace que el ordenador lleve a cabo el procedimiento de decodificación de imagen de la reivindicación 8.

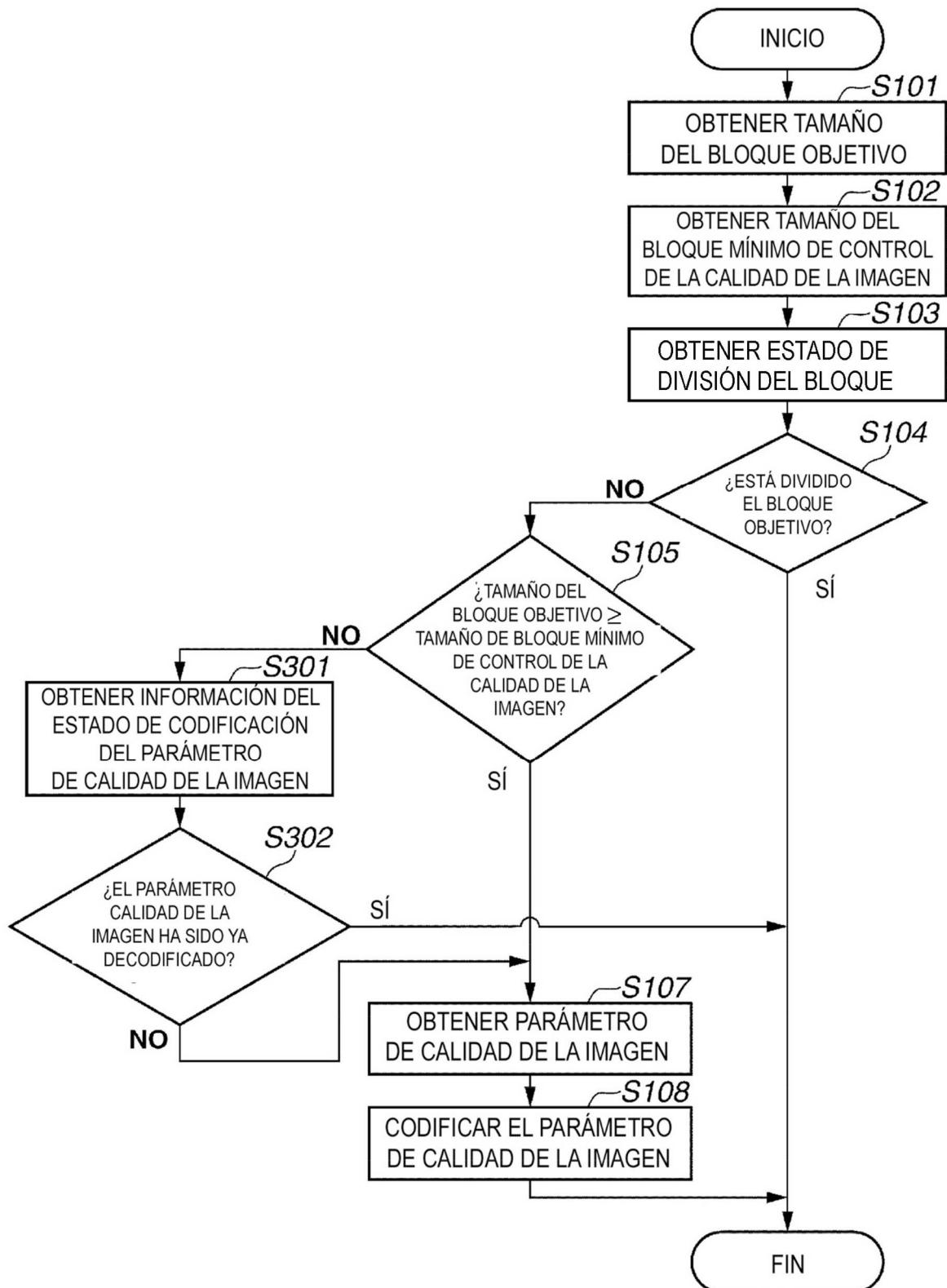
[Fig. 1]



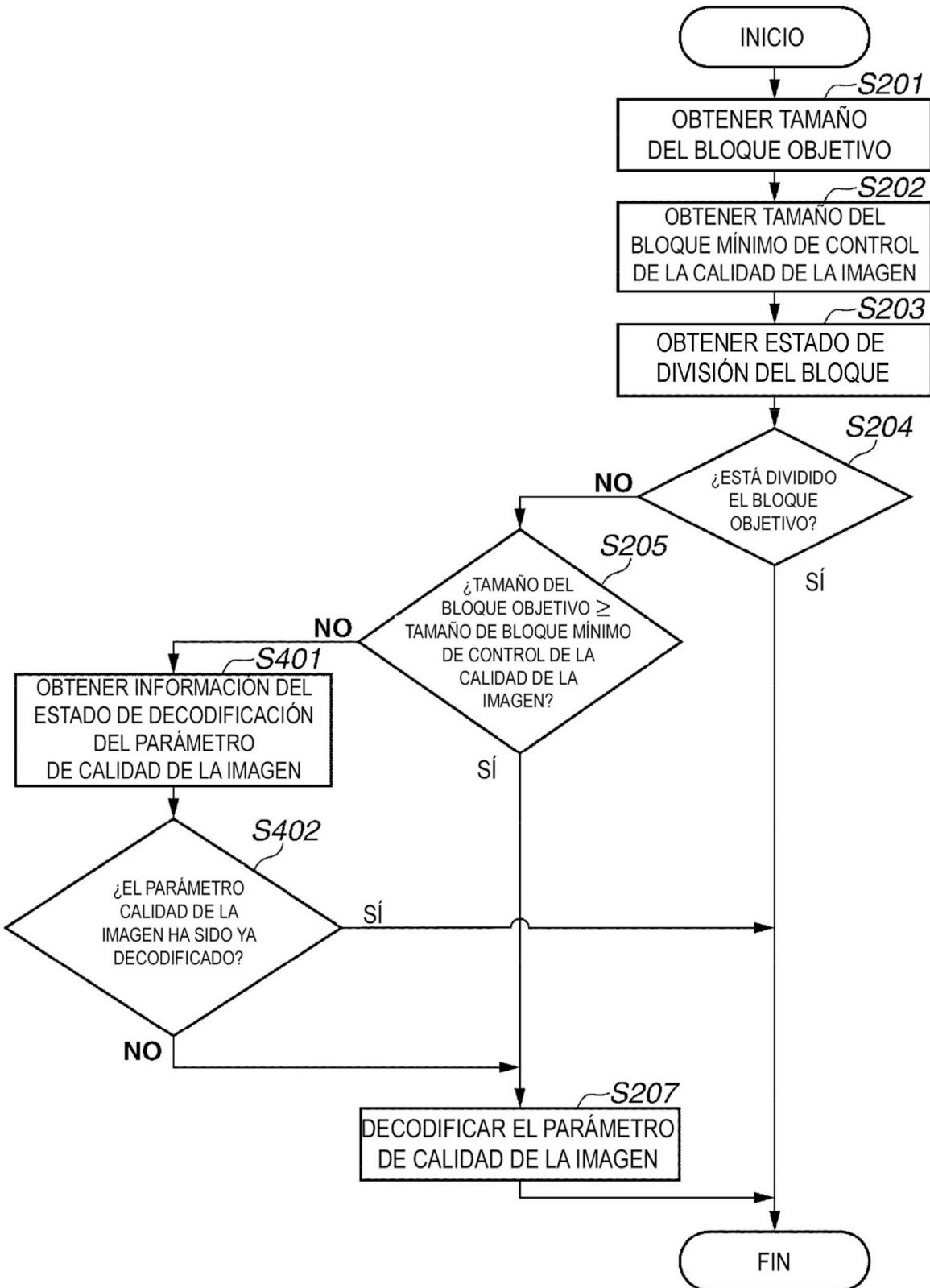
[Fig. 2]



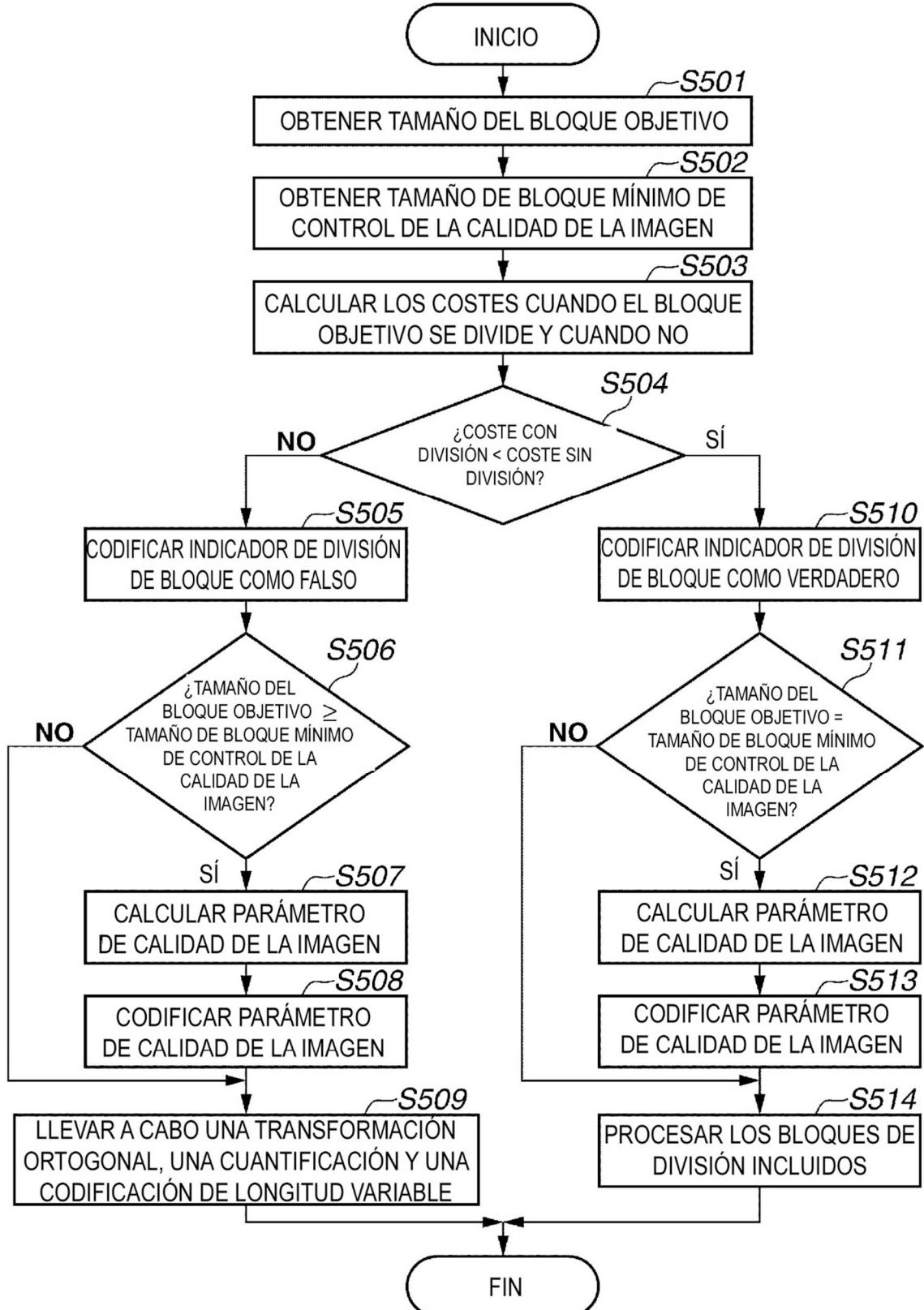
[Fig. 3]



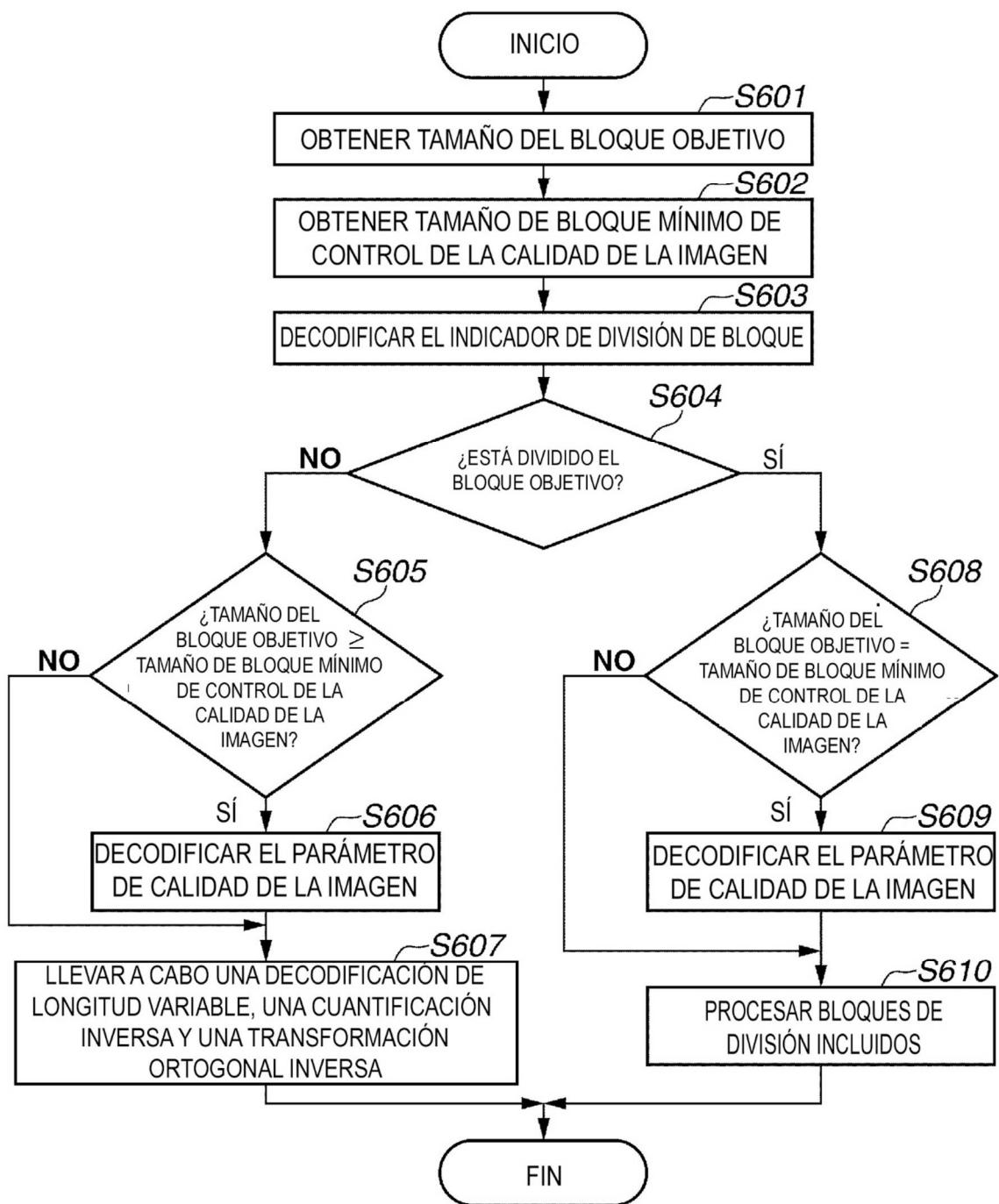
[Fig. 4]



[Fig. 5]



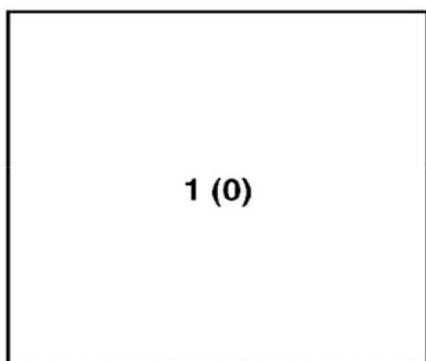
[Fig. 6]



[Fig. 7]

| | | | |
|-------------------------|-------------------------|------------------------|--------------------------|
| 0 (16×16) | 1 (8×8) | 2 (8×8) | 7 (32×32) |
| | 3 (8×8) | 4 (8×8) | |
| 5 (16×16) | 6 (16×16) | | |
| 8 (32×32) | 9 (8×8) | 10 (8×8) | 13 (16×16) |
| | 11 (8×8) | 12 (8×8) | 14 (16×16) |
| | | | 15 (16×16) |

[Fig. 8A]



[Fig. 8C]

| | | |
|-------|-------|---------------|
| 0 (2) | 1 (3) | |
| 0 (4) | 0 (5) | |
| | | 1 (9) 0 (10) |
| | | 0 (11) 0 (12) |
| | | |

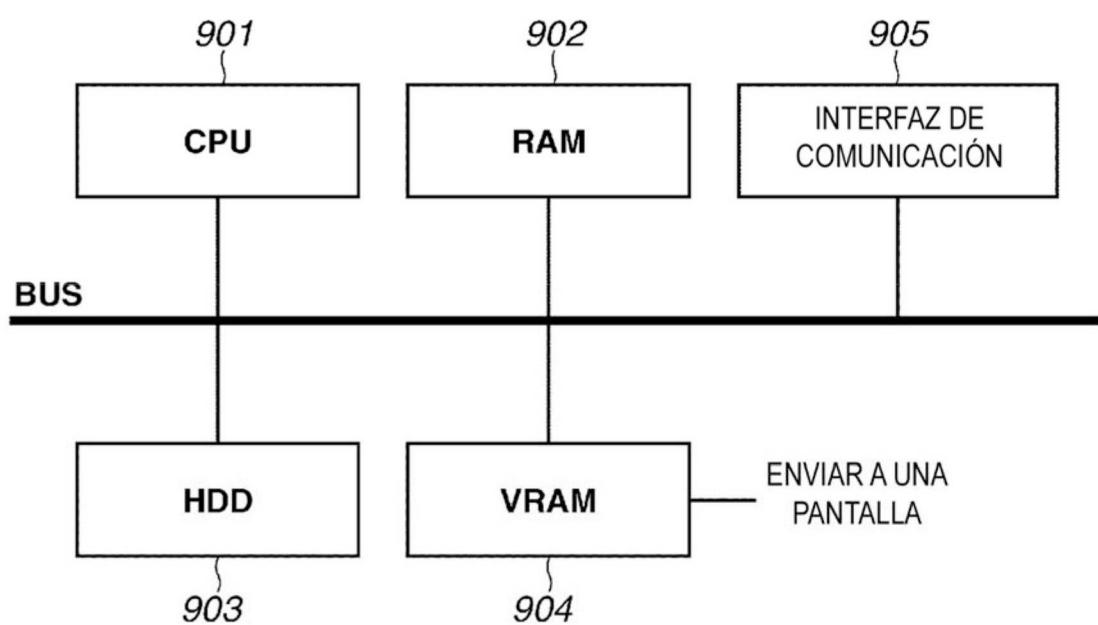
[Fig. 8B]

| | |
|-------|-------|
| 1 (1) | 0 (6) |
| 0 (7) | 1 (8) |

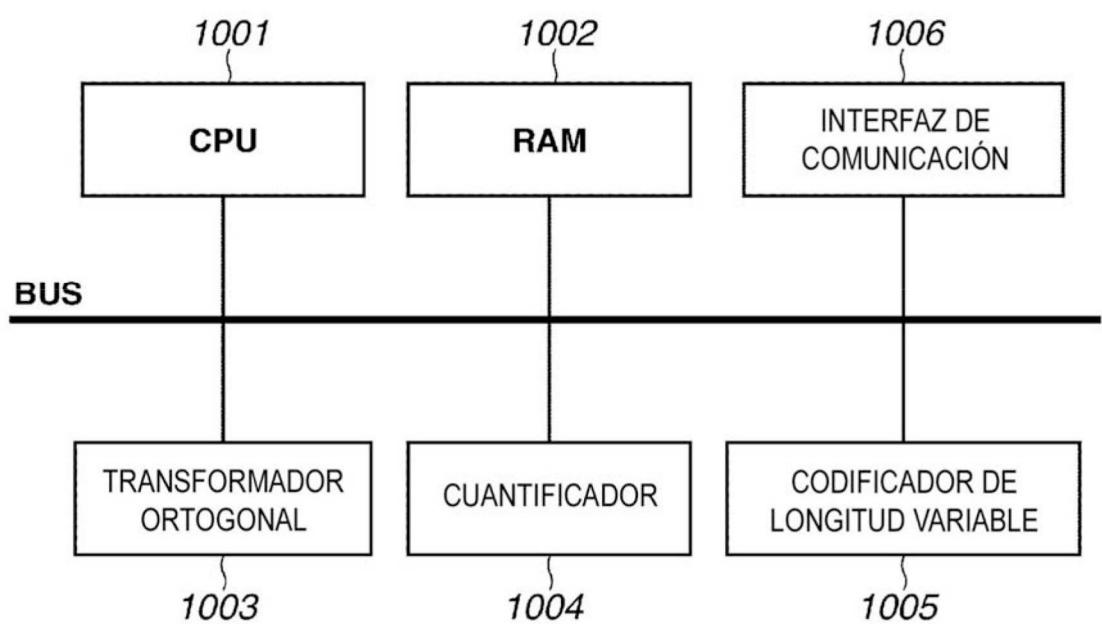
[Fig. 8D]

| | | | |
|---|---|---|---|
| - | - | - | |
| - | - | - | |
| - | - | | - |
| | | - | - |
| | | - | - |

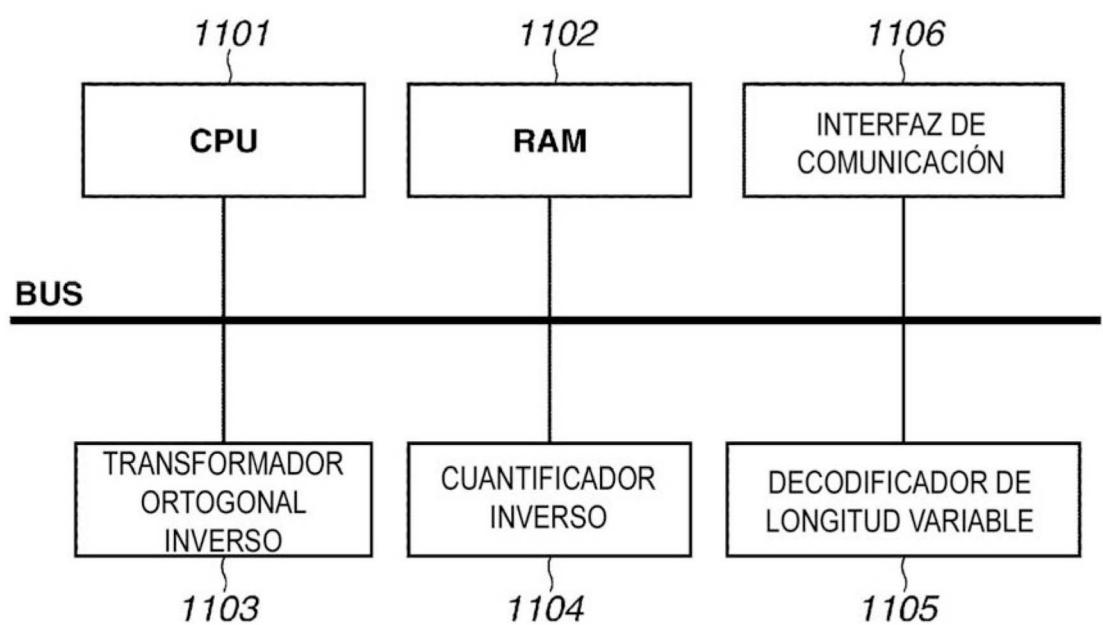
[Fig. 9]



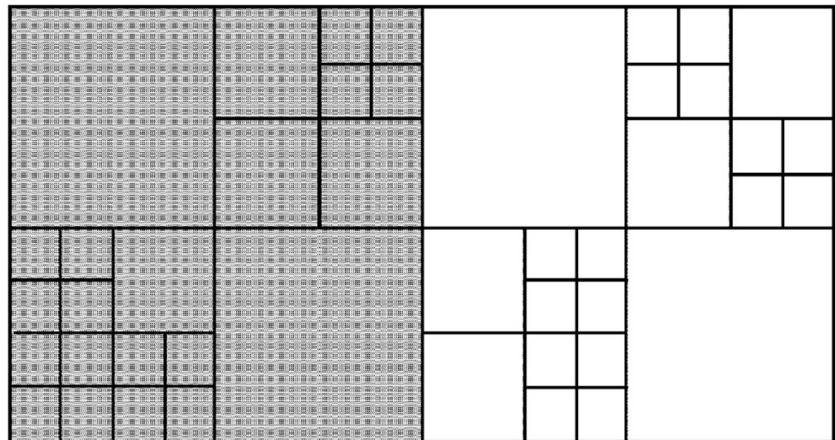
[Fig. 10]



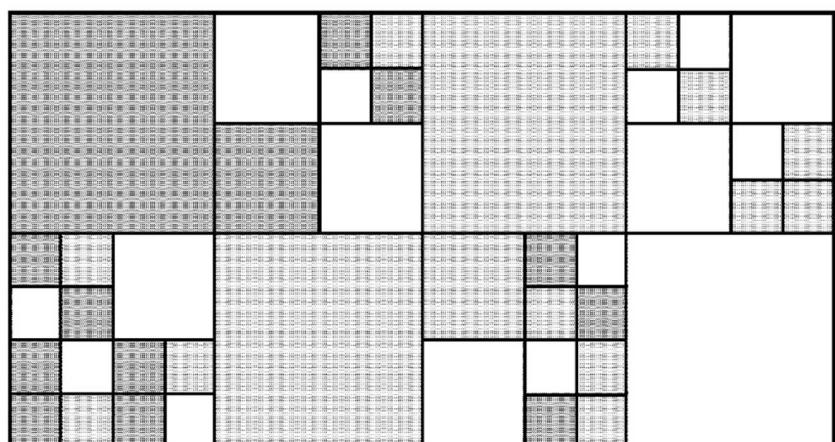
[Fig. 11]



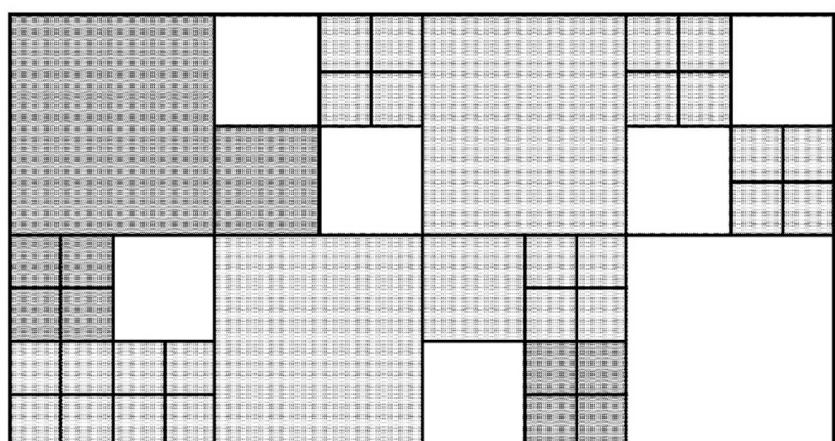
[Fig. 12A]



[Fig. 12B]

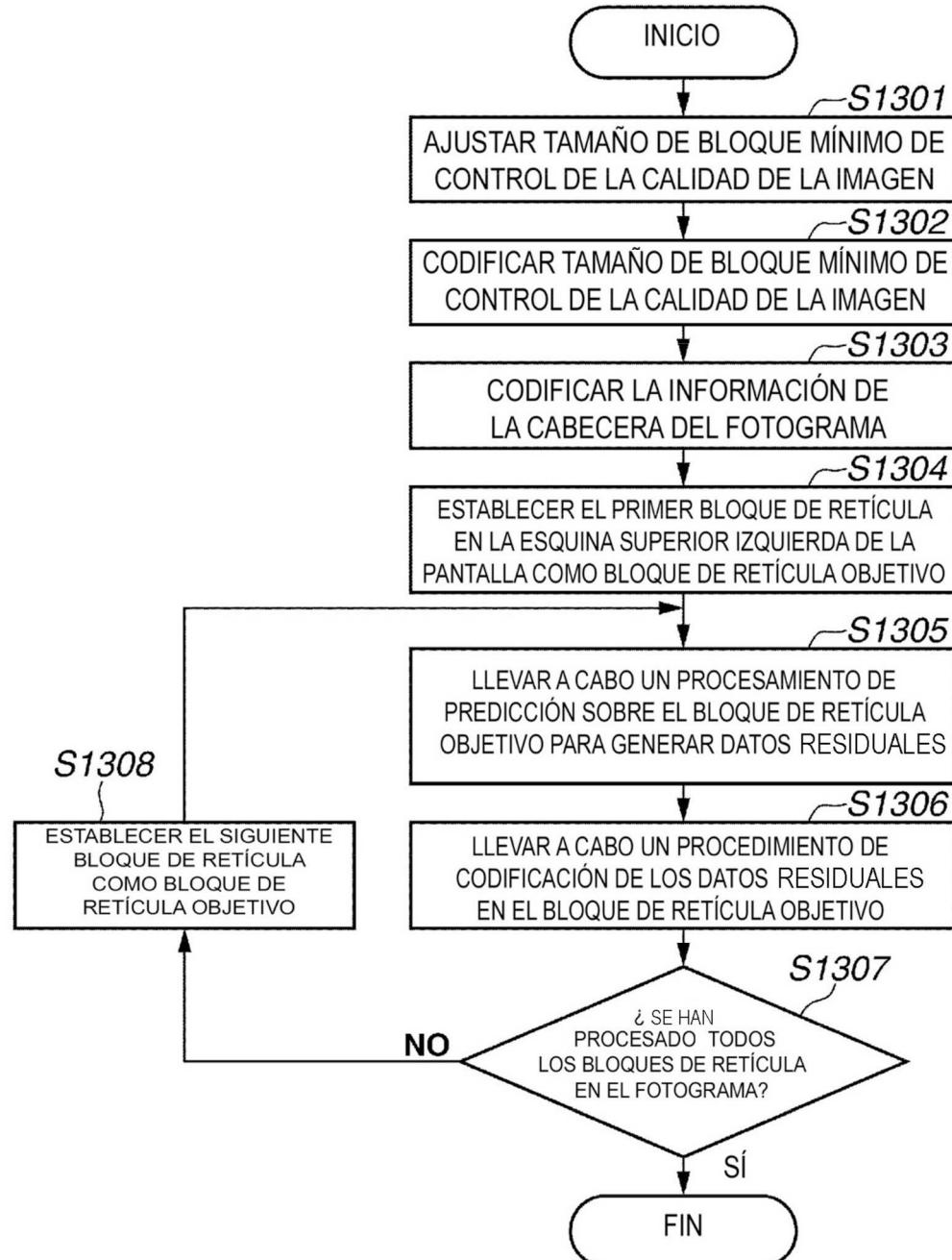


[Fig. 12C]

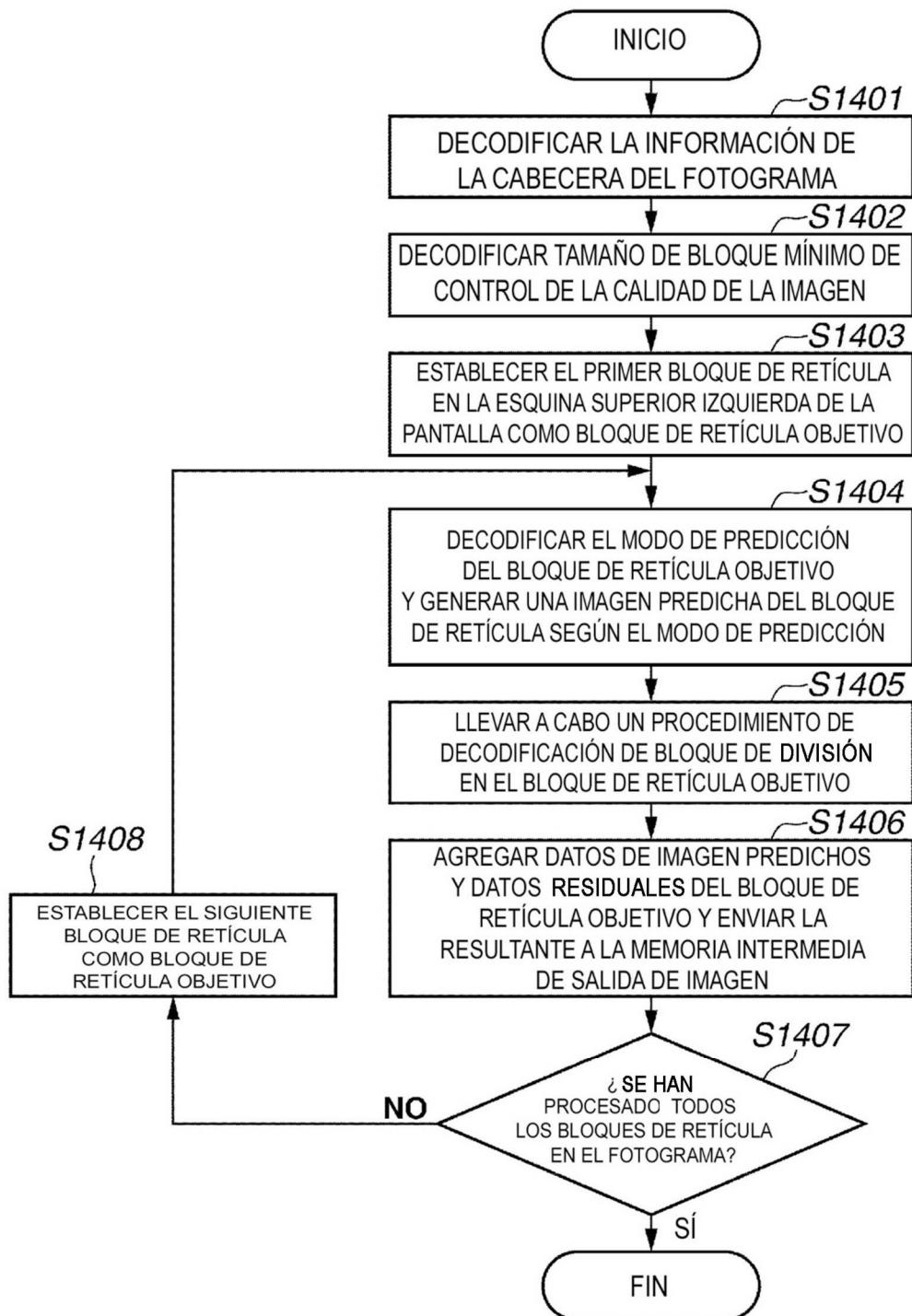


GRIS OSCURO: ALTA CALIDAD DE LA IMAGEN
GRIS CLARO: BAJA CALIDAD DE LA IMAGEN

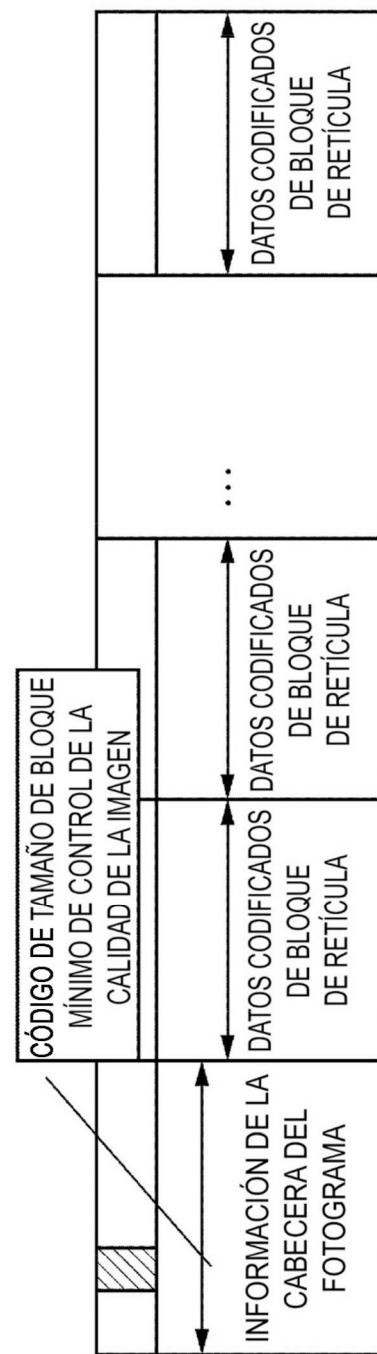
[Fig. 13]



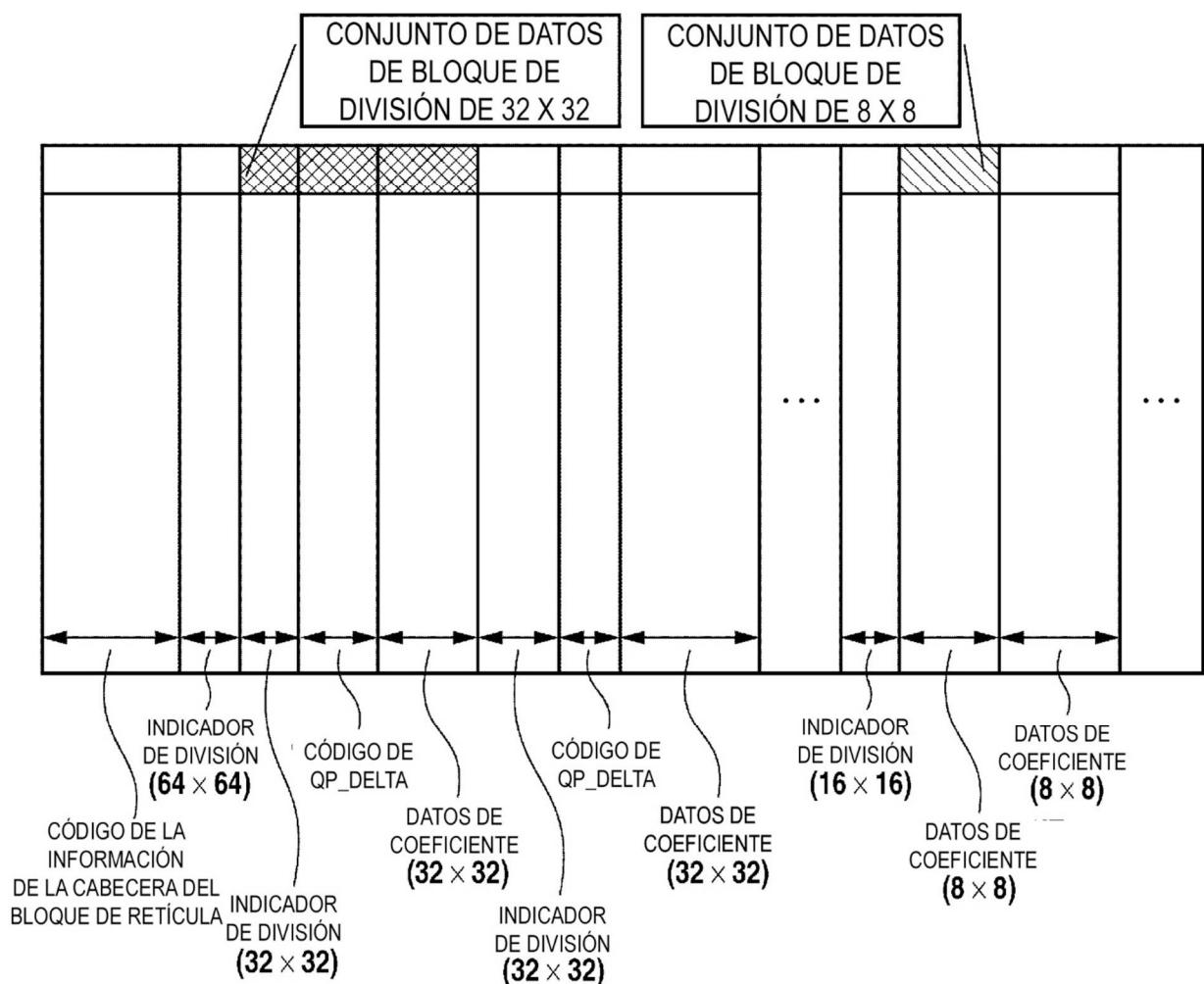
[Fig. 14]



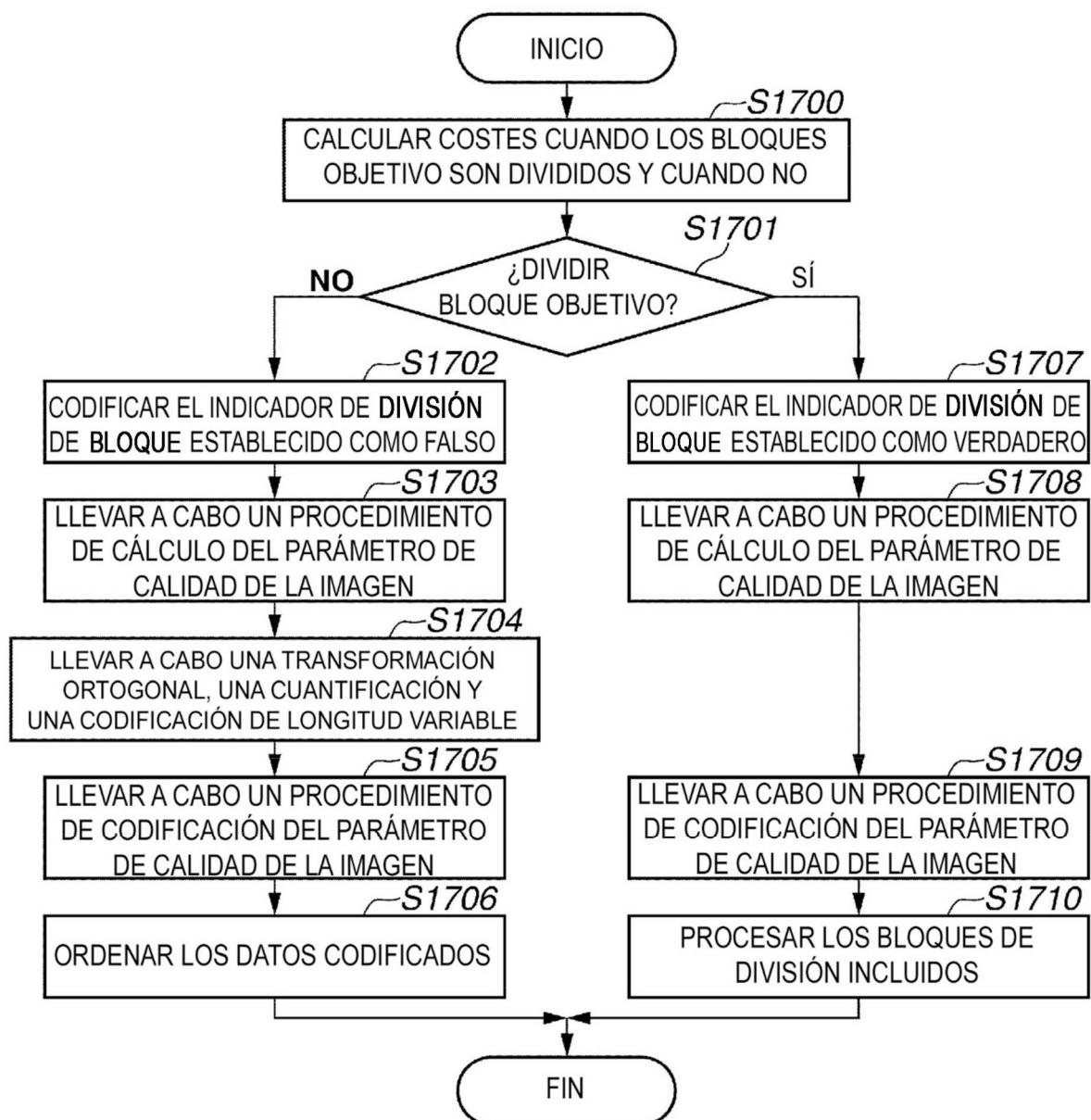
[Fig. 15]



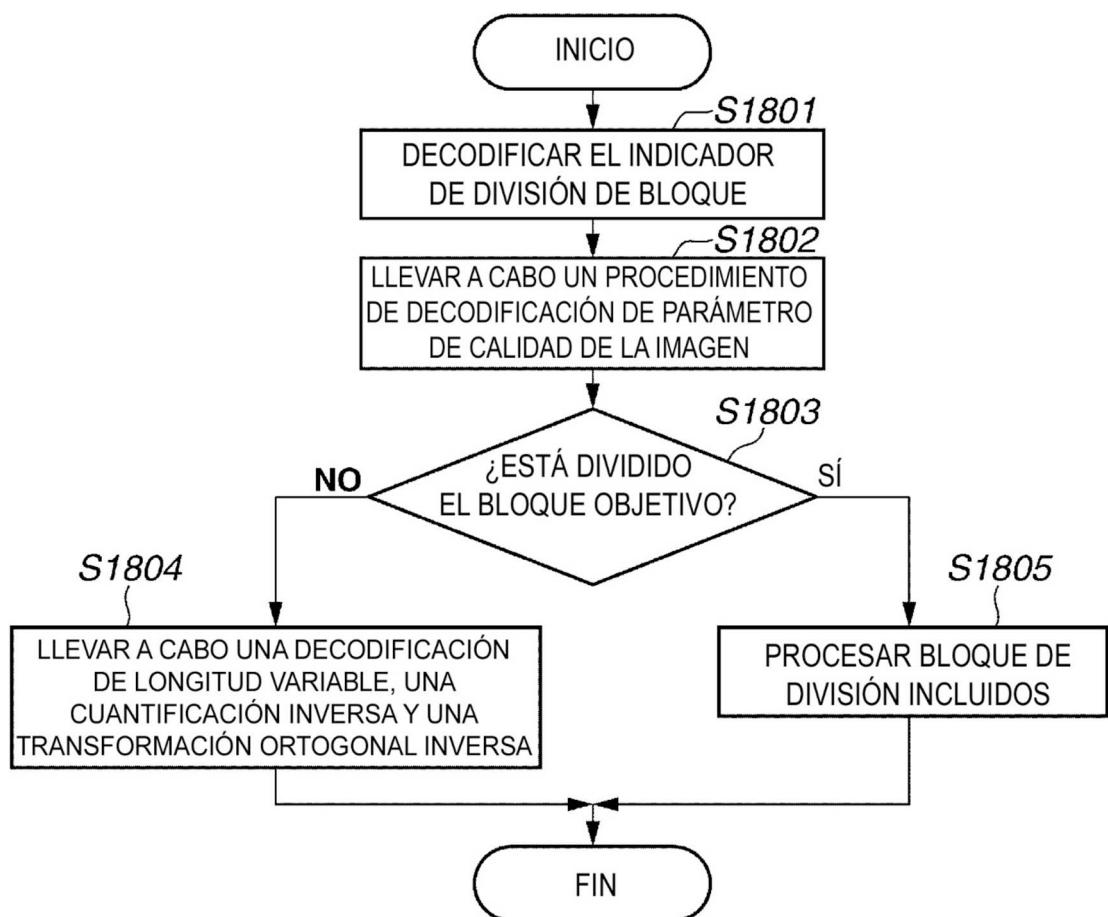
[Fig. 16]



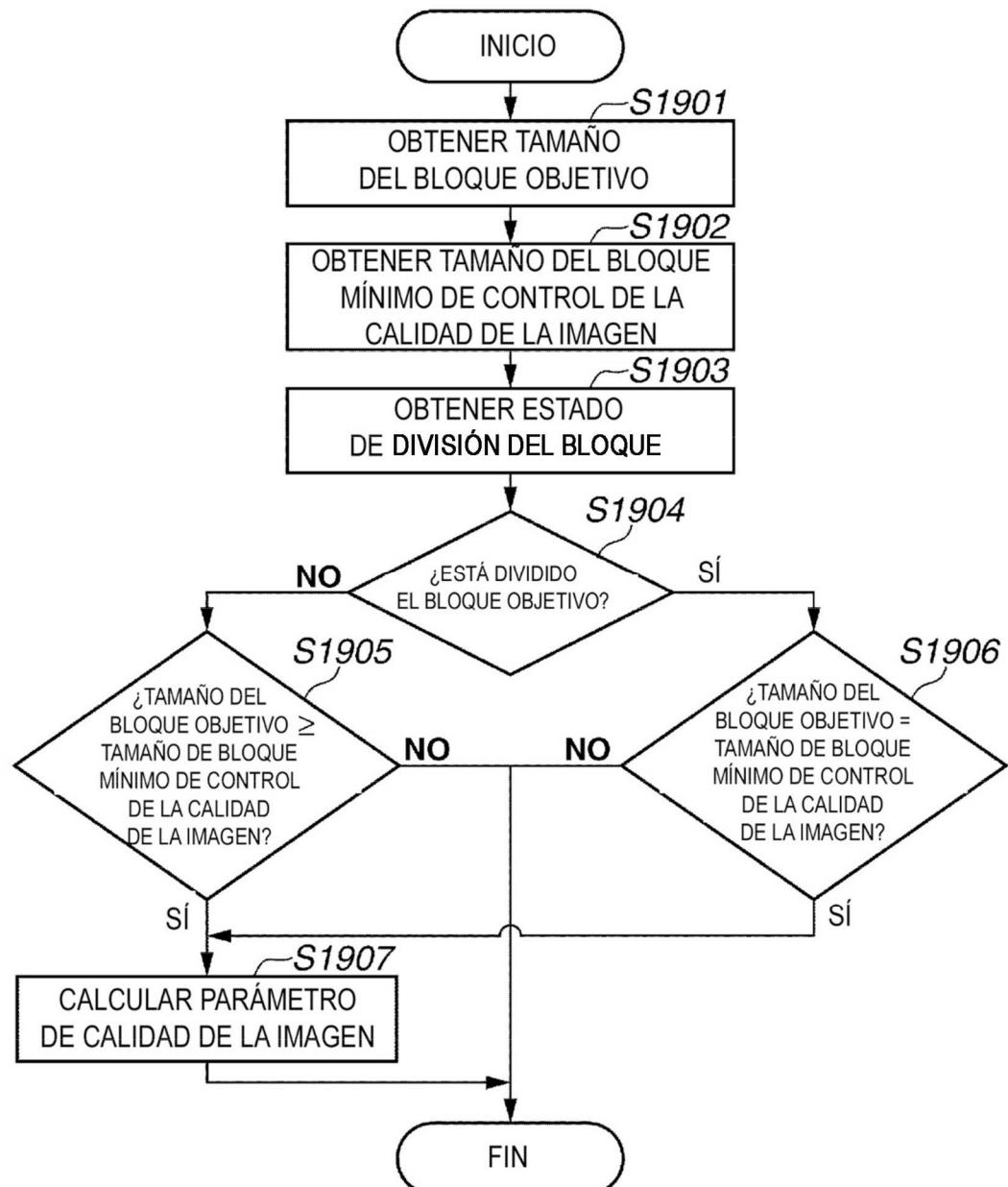
[Fig. 17]



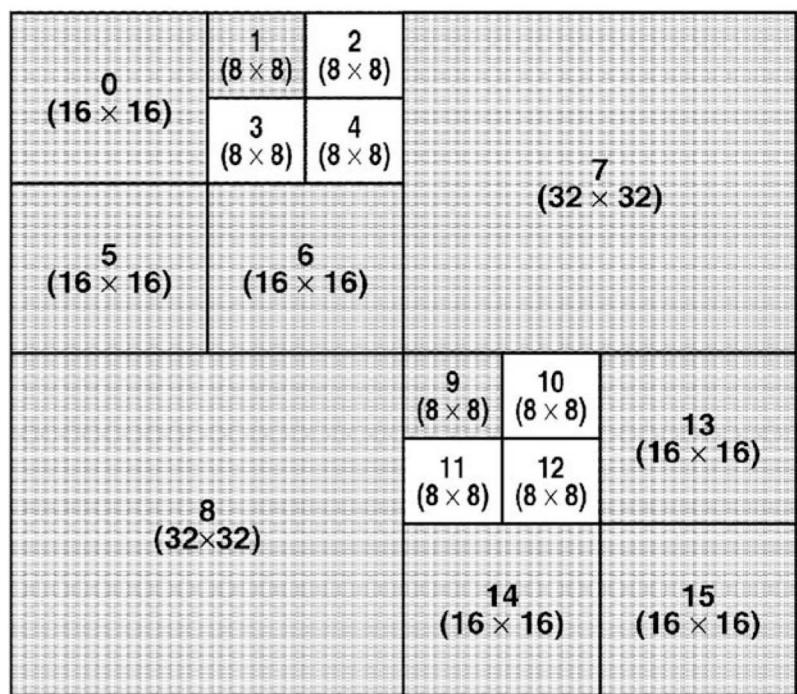
[Fig. 18]



[Fig. 19]



[Fig. 20]



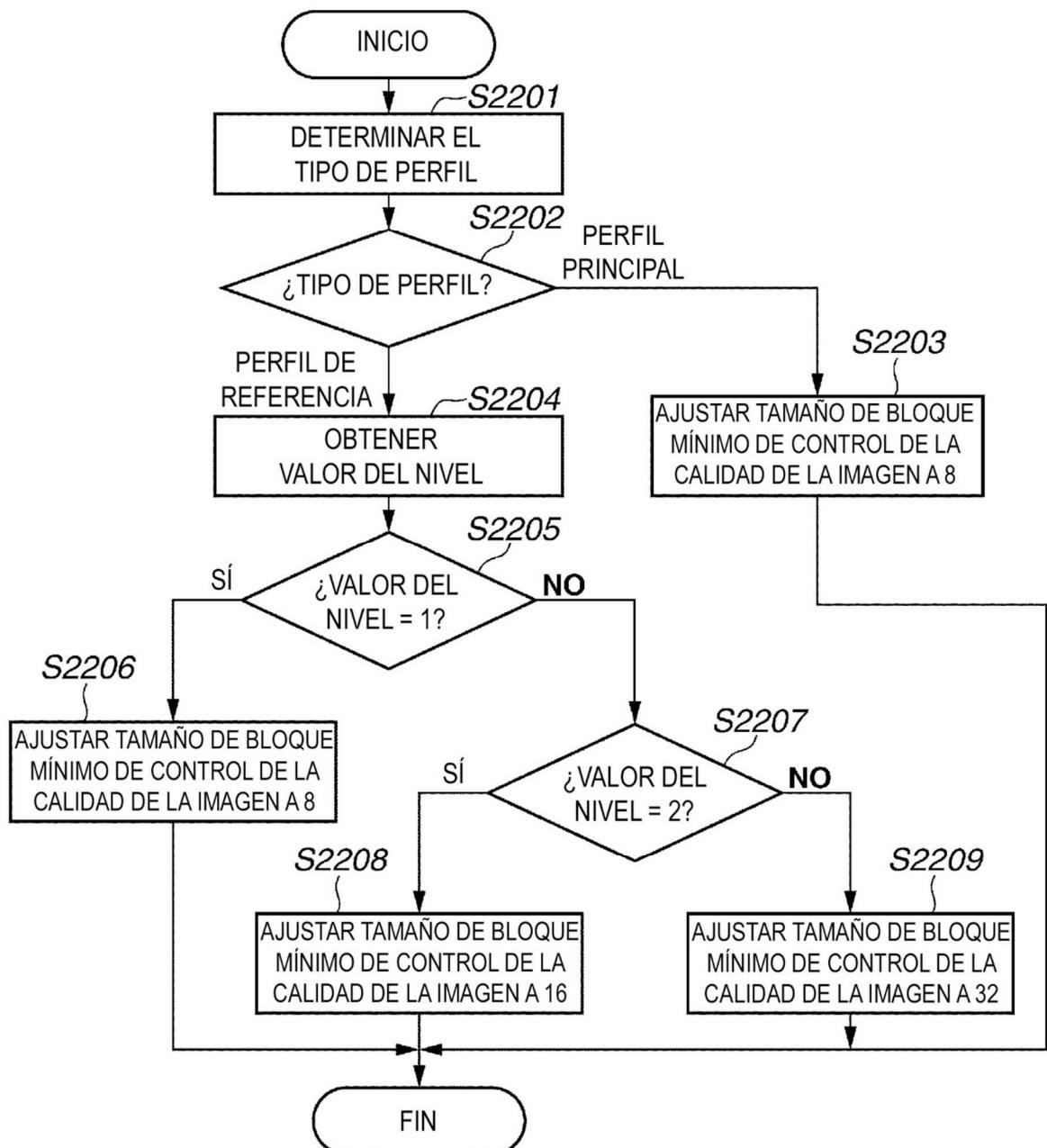
[Fig. 21A]

| NIVEL | NÚMERO MÁXIMO DE BLOQUES DE RETÍCULA POR FOTOGRAMA | VALOR MÍNIMO DEL TAMAÑO DE BLOQUE MÍNIMO DE CONTROL DE LA CALIDAD DE LA IMAGEN | TAMAÑO OBJETIVO DE LA IMAGEN |
|-------|--|--|------------------------------|
| 1 | 80 | 8 | 640 × 480 |
| 2 | 510 | 16 | 1920 × 1080 |
| 3 | 2176 | 32 | 4096 × 2160 |

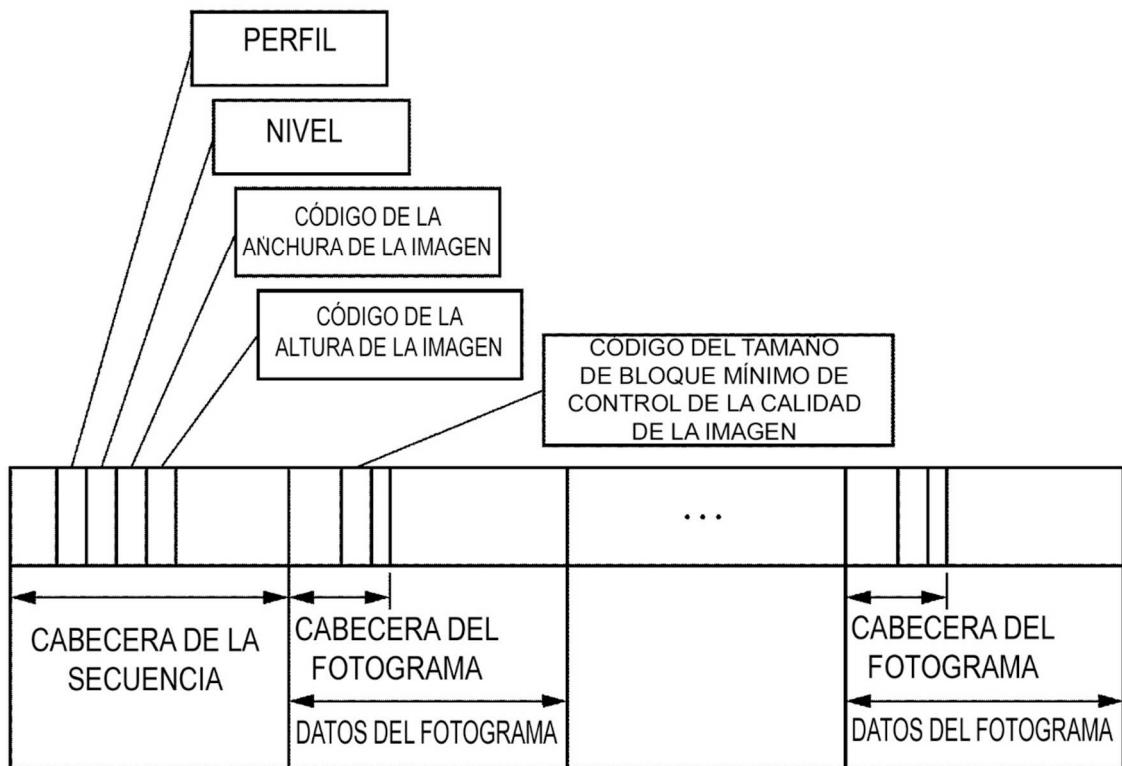
[Fig. 21B]

| NIVEL | NÚMERO MÁXIMO DE BLOQUES DE RETÍCULA POR FOTOGRAMA | VALOR MÍNIMO DEL TAMAÑO DE BLOQUE MÍNIMO DE CONTROL DE LA CALIDAD DE LA IMAGEN | TAMAÑO OBJETIVO DE LA IMAGEN |
|-------|--|--|------------------------------|
| 1 | 80 | 8 | 640 × 480 |
| 2 | 510 | 8 | 1920 × 1080 |
| 3 | 2176 | 8 | 4096 × 2160 |

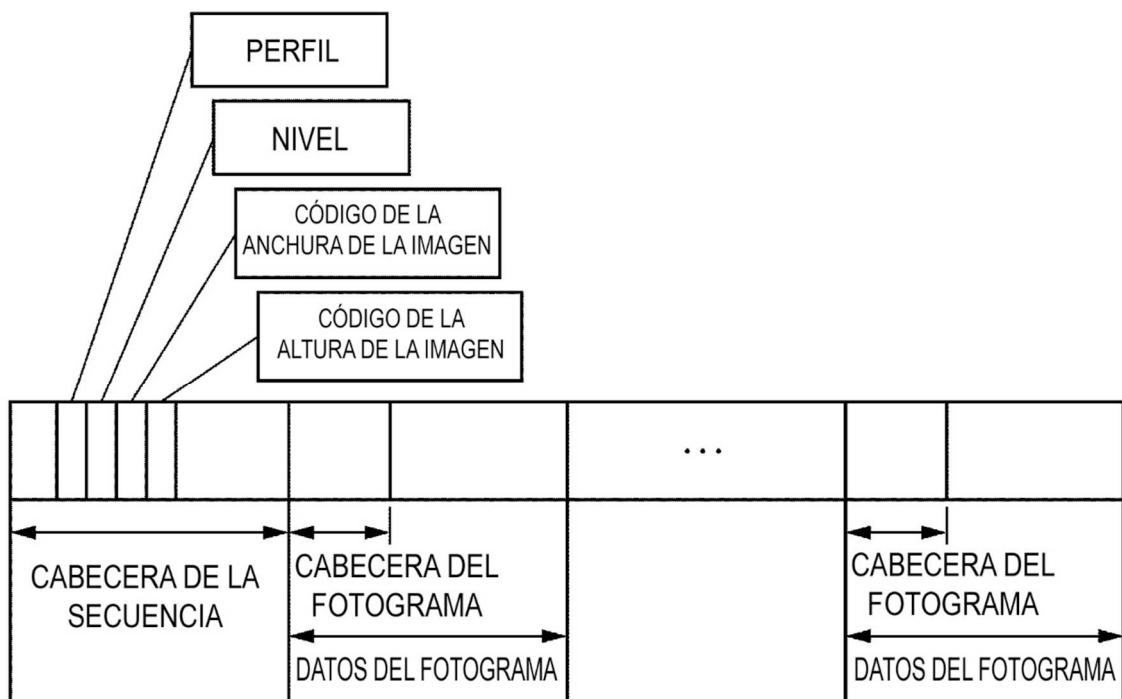
[Fig. 22]



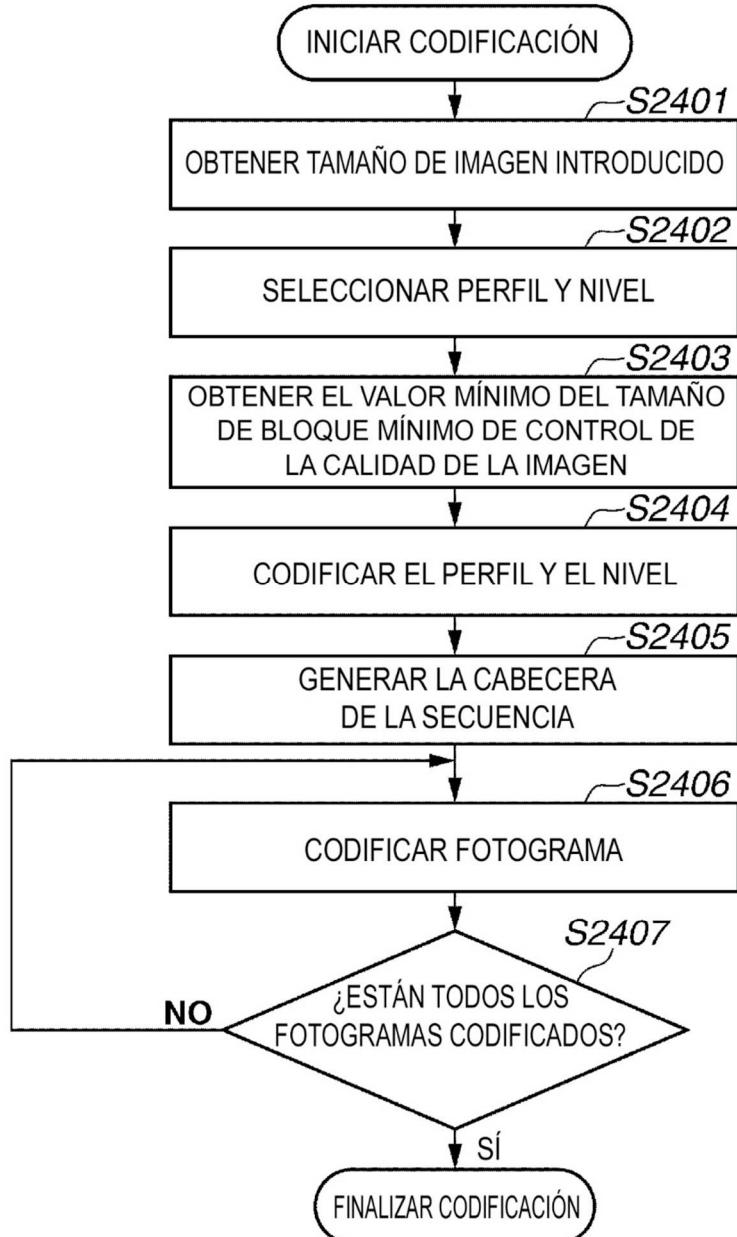
[Fig. 23A]



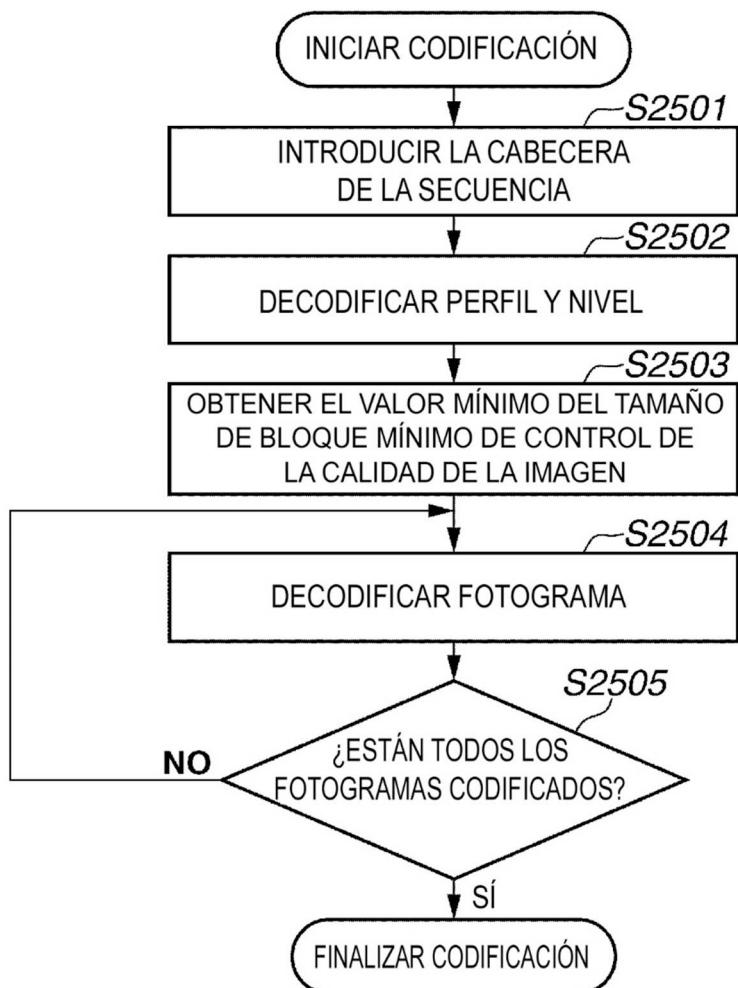
[Fig. 23B]



[Fig. 24]



[Fig. 25]



REFERENCIAS CITADAS EN LA DESCRIPCIÓN

Esta lista de referencias citada por el solicitante es únicamente para mayor comodidad del lector. No forman parte del documento de la Patente Europea. Incluso teniendo en cuenta que la compilación de las referencias se ha efectuado con gran cuidado, los errores u omisiones no pueden descartarse; la EPO se exime de toda responsabilidad al respecto.

Documentos de patentes citados en la descripción

- JP 2005191706