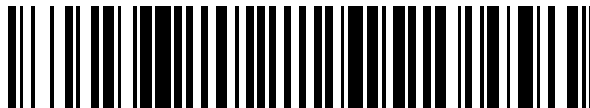


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 600 010**

51 Int. Cl.:

|                    |           |
|--------------------|-----------|
| <b>H04N 11/02</b>  | (2006.01) |
| <b>H04N 19/91</b>  | (2014.01) |
| <b>H04N 19/70</b>  | (2014.01) |
| <b>H04N 19/46</b>  | (2014.01) |
| <b>H04N 19/119</b> | (2014.01) |
| <b>H04N 19/174</b> | (2014.01) |

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **29.06.2011 PCT/SE2011/050876**

87 Fecha y número de publicación internacional: **26.01.2012 WO12011860**

96 Fecha de presentación y número de la solicitud europea: **29.06.2011 E 11809949 (8)**

97 Fecha y número de publicación de la concesión europea: **10.08.2016 EP 2596640**

54 Título: **Codificación y descodificación de imágenes**

30 Prioridad:

**21.07.2010 US 366215 P**

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

**06.02.2017**

73 Titular/es:

**TELEFONAKTIEBOLAGET LM ERICSSON (PUBL)  
(100.0%)  
164 83 Stockholm, SE**

72 Inventor/es:

**SJÖBERG, RICKARD y  
PRIDDLE, CLINTON**

74 Agente/Representante:

**DE ELZABURU MÁRQUEZ, Alberto**

ES 2 600 010 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**

Codificación y decodificación de imágenes

**Campo técnico**

5 Las presentes realizaciones refieren en general al tratamiento de imágenes o tramas de vídeo, y en particular a la codificación y decodificación de tales imágenes o tramas de vídeo.

**Antecedentes**

10 H.264, también conocido como MPEG-4 (Grupo de Expertos de Imágenes en Movimiento) AVC (Codificación Avanzada de Vídeo), es el estado normalizado de la técnica de codificación de vídeo. Es un códec híbrido que se basa en la eliminación de la redundancia entre tramas, denominado inter codificación, y dentro de las tramas, denominado intra codificación. Lo que se obtiene tras el proceso de codificación son los datos VCL (Capa de Codificación de Vídeo), que se encapsulan además dentro de las unidades NAL (Capa de Abstracción de la Red) antes del transporte o del almacenamiento.

15 En H.264/MPEG-4 AVC una imagen de una secuencia de vídeo se compone de macrobloques con un tamaño fijo de 16x16 píxeles y la codificación de la imagen continúa macrobloque por macrobloque. Cada imagen de la secuencia de vídeo se divide en uno o más fragmentos. Un fragmento es una pieza descodificable independientemente de la imagen. Esto quiere decir que si se pierde un fragmento de una imagen, los otros fragmentos de la imagen son aún descodificables. Además, los fragmentos se pueden utilizar para la codificación y descodificación ya que son independientes de los otros fragmentos de la imagen. En H.264/MPEG-4 AVC el límite de un fragmento debe situarse entre los bordes de dos fragmentos contiguos, de acuerdo con el orden de  
20 codificación, macrobloques.

HEVC (Codificación de Vídeo de Alto Rendimiento) es un sucesor de H.264/MPEG-4 AVC. HEVC pretende mejorar sustancialmente el rendimiento de la codificación comparado con H.264/MPEG-4 AVC, es decir, reducir los requisitos de velocidad de bits manteniendo la calidad de la imagen. HEVC es el objetivo en la siguiente generación de pantallas de HDTV (Televisión de Alta Definición) y contiene sistemas de captura que caracterizan las velocidades progresivas de tramas escaneadas y resoluciones de pantalla a partir de QVGA (Disposición Gráfica de Cuarto de Pantalla) (320x240) hasta 1080p y Ultra HDTV (7680x4320), así como calidad mejorada de la imagen.  
25

HEVC permite la utilización de las llamadas unidades mayores de codificación (LCUs) que son bloques de píxeles que tienen un tamaño mayor que los macrobloques de H.264/MPEG-4 AVC para proporcionar un rendimiento mejorado en la codificación. Con objeto de exagerar tanto las grandes áreas homogéneas como las pequeñas áreas detalladas en la misma imagen se ha propuesto una codificación jerárquica para HEVC. Las LCUs en una imagen se escanean según un orden predefinido y cada LCU se puede dividir en menores unidades de codificación (CUs), que por su parte se pueden dividir jerárquicamente en una estructura de árbol en grado 4 descendente a una unidad de codificación menor (SCU). Una imagen puede, por consiguiente, codificarse como una mezcla de unidades de codificación con diferentes tamaños que van desde las LCUs bajando hasta las SCUs.  
30

35 En correspondencia con H.264/MPEG-4 AVC una imagen de una secuencia de vídeo se puede dividir en uno o más fragmentos en HEVC. El límite del fragmento está en HEVC alineado con el borde de dos fragmentos adyacentes, de acuerdo con un orden predefinido, LCUs.

Tanto las normas H.264/MPEG-4 AVC como HEVC requieren la determinación y el uso de direcciones con el fin de identificar el primer macrobloque o unidad de codificación de un fragmento y de ese modo el comienzo del fragmento en la imagen o trama de vídeo. Tales direcciones, aunque necesarias en el descodificador, sobrecargan los datos codificados de la imagen. Además, con la introducción de la división jerárquica de la LCU en HEVC surgen nuevos retos en relación con la codificación y descodificación de imágenes o tramas de vídeo. Por tanto, existe una necesidad de una codificación y descodificación eficientes que puedan manejar que las direcciones del fragmento comiencen de una manera flexible y eficiente.  
40

45 Wiegand, JVT de ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC 1/SC29/WG11 y ITU-T SG16 Q.6), 12ª Reunión: Redmond WA, EE.UU., 17-12 de julio de 2004, Documento: JVT-K051 describe las enmiendas y el informe de las erratas al aumento del margen de fidelidad relativas a la norma ITU-T Rec. H.264 | ISO/IEC 14496-10 de mayo de 2003.

**Resumen**

50 Es un objetivo general proporcionar una gestión eficiente de los fragmentos en las Imágenes y tramas de vídeo.

Es un objetivo particular marcar las posiciones de comienzo del fragmento de una manera eficiente.

Estos y otros objetivos los cumplen las realizaciones de la invención como se define por las reivindicaciones independientes. Aspectos adicionales se describen en el presente documento.

Un aspecto de las realizaciones define un método de codificar una imagen que comprende varios fragmentos. Se genera una representación codificada del fragmento para cada fragmento en la imagen basándose en los valores de píxel dentro de los píxeles del fragmento. Un indicador del fragmento correspondiente se asigna y se fija para cada uno de los fragmentos. El primer fragmento en la imagen tiene un indicador del fragmento fijado en un primer valor definido, mientras que los fragmentos restantes tienen su respectivo indicador del fragmento fijado en un segundo valor definido. Las direcciones del fragmento permiten la identificación de la posición de una primera unidad de codificación de un fragmento y con ello se generan los comienzos de los fragmentos dentro de la imagen para el resto de los fragmentos con exclusión del primer fragmento en la imagen. Estas direcciones del fragmento se incluyen junto con las representaciones codificadas del fragmento y los indicadores del fragmento en una representación codificada de la imagen.

Otro aspecto de las realizaciones se refiere a un dispositivo para codificar una imagen que comprende múltiples fragmentos. Un generador de la representación del dispositivo genera una representación correspondiente codificada del fragmento para cada fragmento en la imagen. El dispositivo comprende un fijador del indicador configurado para fijar un indicador del fragmento asociado con un primer fragmento en la imagen a un primer valor, mientras que el(los) indicador(es) del(de los) fragmento(s) restante(s) está(n) fijados a un segundo valor definido. Un generador de la dirección genera una dirección correspondiente del fragmento para cada fragmento del(de los) fragmento(s) restante(s) para permitir la identificación de una posición respectiva de una primera unidad de codificación y el comienzo del fragmento dentro de la imagen. Un gestor de la representación genera una representación codificada de la imagen para la imagen que comprende las representaciones codificadas del fragmento, las direcciones del fragmento y los indicadores del fragmento.

Un aspecto adicional de las realizaciones define un método de descodificar una representación codificada de imagen de una imagen que comprende múltiples fragmentos. Un indicador del fragmento asociado con un fragmento en la imagen se recupera de la representación codificada. El valor de este indicador del fragmento se emplea con el fin de determinar si una dirección del fragmento necesita ser recuperada de la representación codificada del fragmento con el fin de identificar la posición de comienzo del fragmento dentro de la imagen y la parte del fragmento que pertenece al fragmento. Si el indicador del fragmento tiene un primer valor definido, el fragmento presente es el primer fragmento en la imagen. Los valores de píxel de los píxeles del fragmento generados al descodificar una representación codificada del fragmento asociada con el fragmento y recuperados de la representación codificada de la imagen se asignan entonces a una primera porción de la imagen empezando con un comienzo del fragmento definido dentro de la imagen, tal como la esquina superior izquierda. Si el indicador del fragmento, en cambio, tiene un segundo valor definido, se emplea la dirección del fragmento para identificar el comienzo del fragmento dentro de la imagen y los valores de píxel generados al descodificar la representación codificada del fragmento del píxel se asignan a la porción de la imagen empezando por el comienzo del fragmento identificado.

Otro aspecto más adicional de las realizaciones se refiere a un dispositivo para descodificar una representación codificada de imagen de una imagen que comprende múltiples fragmentos. Un recuperador de la dirección del dispositivo recupera un indicador del fragmento asociado con un fragmento a descodificar de la representación codificada de la imagen. Un recuperador de la dirección recupera una dirección del fragmento a partir de la representación codificada de la imagen si el indicador del fragmento tiene un segundo valor definido. Si el indicador del fragmento en cambio tiene un primer valor definido, un generador de la representación genera una representación descodificada de los valores de píxel de los píxeles en el fragmento basándose en una representación codificada del fragmento asociada con el fragmento y recuperada de la presentación codificada de la imagen. Estos valores de los píxeles son asignadas a continuación por un asignador de valores a una primera porción de la imagen a partir de una posición de partida definida que se identifica sin la necesidad de ninguna dirección del fragmento marcado, por ejemplo, la esquina superior izquierda: Si el indicador del fragmento en cambio tiene el segundo valor definido, la dirección recuperada del fragmento la emplea el asignador con el fin de identificar el comienzo de una porción de la imagen que pertenezca al fragmento actual y a continuación asigna los valores del píxel a partir del generador de la representación a esta porción identificada.

Las realizaciones proporcionan una gestión eficiente de los fragmentos dentro de las imágenes o tramas de vídeo en cuanto a que proporcionan una forma eficiente de señalización y de identificación de posiciones de comienzo del fragmento en una imagen o trama de vídeo. Los indicadores del fragmento de las realizaciones proporcionan una mejora significativa en la identificación de los comienzos del fragmento para los primeros fragmentos en la imagen, pero sin ninguna necesidad de marcar y calcular la dirección del fragmento en el descodificador.

**Breve descripción de los dibujos**

La invención, junto con otros objetos y ventajas de la misma, puede entenderse mejor haciendo referencia a la siguiente descripción considerada junto con los dibujos adjuntos, en los que:

La figura 1 es un diagrama de flujo que ilustra un método de codificar una imagen de acuerdo con una realización;

La figura 2 ilustra una realización de una imagen dividida en múltiples fragmentos y que comprende las múltiples unidades mayores de codificación (LCUs);

Las figuras 3A y 3B ilustran realizaciones del comienzo de un fragmento que coincide con un borde entre las unidades de codificación dentro de una LCU;

La figura 4 ilustra esquemáticamente una realización de un orden de codificación y descodificación para el tratamiento de unidades de codificación;

- 5 La figura 5 es un diagrama de secuencia que ilustra etapas adicionales al método de codificación de la figura 1 de acuerdo con una realización;

La figura 6 ilustra esquemáticamente una realización de una representación codificada de una imagen;

La figura 7 es un diagrama de flujo que ilustra una realización de descodificar una representación codificada de una imagen;

- 10 La figura 8 es un diagrama de flujo que ilustra etapas adicionales del método de la figura 7 de acuerdo con una realización;

La figura 9 es un diagrama de flujo que ilustra etapas adicionales del método de la figura 7 de acuerdo con una realización;

- 15 La figura 10 es un diagrama esquemático de bloques de un dispositivo para codificar una imagen de acuerdo con una realización;

La figura 11 es un diagrama esquemático de bloques de un dispositivo para descodificar una representación codificada de una imagen de acuerdo con una realización; y

La figura 12 es un diagrama esquemático de bloques de un terminal de medios de acuerdo con una realización.

### Descripción detallada

- 20 A lo largo de todos los dibujos, se utilizan los mismos números de referencia para elementos similares o correspondientes.

Las realizaciones se refieren generalmente a la gestión de los fragmentos dentro de las imágenes o de las tramas de vídeo. Con mayor detalle, las realizaciones proporcionan un modo flexible y mínimamente eficiente de marcar las direcciones de comienzo del fragmento. Las realizaciones se pueden aplicar a cualquier codificación y descodificación de imagen o de trama de vídeo en las cuales una imagen o una trama de vídeo, por ejemplo de una secuencia de vídeo, pueda comprender múltiples, es decir al menos dos, fragmentos y en donde se le debe marcar al descodificador el comienzo de los fragmentos. La realización puede asimismo aplicarse al estado de la técnica de imágenes o codificación de vídeo, tal como H.264/MPEG-4 AVC, pero es en particular ampliamente aplicable a la codificación y descodificación de imágenes que utilizan división jerárquica de bloques de píxeles en bloques menores de píxeles. Las realizaciones son por consiguiente apropiadas para ser usadas en conexión con (HEVC) Codificación de Vídeo de Alto Rendimiento pero no se limitan a eso.

- Una característica de las realizaciones reside en diferenciar el primer fragmento en la imagen o en la trama de vídeo y los fragmentos restantes. En la técnica anterior, una dirección del fragmento se determina durante la codificación para cada fragmento en la imagen y se incluye en la representación codificada de la imagen para ser utilizada por un descodificador con el fin de identificar el comienzo del fragmento dentro del área de la imagen o de la trama de vídeo. Dependiendo del tamaño total de la imagen, estas direcciones de los fragmentos pueden ser bastante largas y por consiguiente pueden añadir una magnitud significativa de costos a los datos codificados de la imagen. Por ejemplo, una imagen o trama de vídeo, H.264/MPEG-4 AVC podría constar de 1280x960 píxeles. Si el comienzo del fragmento se alinea con los bordes del macrobloque y el comienzo de fragmentos se expresa como coordenadas X e Y relativas a un origen fijo, normalmente la esquina superior izquierda de la imagen, la dirección del fragmento necesitaría ser  $\log_2(\text{ceil}(1280/16))+\log_2(\text{ceil}(960/16))=7+6=13$  bits en este simple ejemplo.  $\text{Ceil}()$  indica la función techo definida como  $\text{ceil}(x) = \lceil x \rceil$  y extrae el entero menor no mayor de x. Esto podrían no ser muchos datos pero con las velocidades de tramas y bits actuales para la descodificación y presentación del vídeo, se acceden normalmente a varios cientos de fragmentos por segundo de modo que el tamaño total de la magnitud de los datos de la dirección de fragmento que necesita ser generada y enviada en el codificador al descodificador es bastante significativa para un secuencia de vídeo. Adicionalmente, las direcciones del fragmento necesitan ser localizadas, recuperadas y tratadas en el descodificador antes de que los datos reales del píxeles de un fragmento puedan ser descodificados y asignados a la parte correcta de la imagen.

- Los problemas mencionados anteriormente al manejar las direcciones de los fragmentos pueden llegar a ser incluso más destacables cuando se emplean la codificación y descodificación jerárquicas, tal como en HEVC, si los comienzos del fragmento se pueden alinear en unidades de codificación de un nivel inferior que las mayores unidades de codificación. La figura 2 ilustra esquemáticamente este concepto.

Una imagen 1, tal como una trama de vídeo en un secuencia de video, se puede dividir en un número de unidades mayores de codificación (LCUs) 10, también denominadas en la técnica mayores bloques de decodificación en árbol (LCTBs) o mayores bloques de codificación (LCBs). La LCU 10 es el mayor bloque posible de píxeles que se puede manejar durante la codificación y la decodificación, y se puede gestionar, por ejemplo, de acuerdo con los modos conocidos intra o inter codificación/decodificación. La LCU 10 puede por su parte dividirse jerárquicamente en un cierto número de menores, con respecto al número de píxeles, bloques de píxeles, denominados normalmente unidades de codificación (CUs) 20A, 20B, bloques de codificación en árbol (CTBs) o bloques de codificación (CBs). Estas CUs 20A, 20B pueden por su parte dividirse además incluso en bloques menores 30A, 30B de píxeles de una manera jerárquica descendente hasta el menor bloque posible de píxeles, denominado unidad menor de codificación (SCU) bloque menor de codificación en árbol (SCTB) o bloque menor de codificación (SCB).

En claro contraste con las técnicas precedentes que limitan el posicionamiento del comienzo del fragmento 2, 3 en una imagen 1 al alinearse con el borde entre dos LCUs 10A, 10B contiguas, algunas realizaciones proporcionan un posicionamiento mucho más flexible del comienzo del fragmento habilitando potencialmente el comienzo del fragmento 4, 5 al alinearlos con el borde entre cualesquiera CUs 10A, 10B, 20A, 20B, 30A, 30B contiguas en la imagen desde el nivel SCU hasta el nivel LCU. Por lo tanto, es posible entonces que el comienzo del fragmento se pueda posicionar dentro de una LCU con la última CU del fragmento precedente y la primera CU del fragmento actual en la misma LCU. La figura 2 ilustra esquemáticamente esto. El borde entre un primer fragmento 2 y el segundo fragmento 3 coincide con el borde de dos contiguos, de acuerdo con el orden de tratamiento, LCUs 10A, 10B. El borde entre el segundo fragmento 3 y el tercer fragmento 4 está en cambio alineado con el borde de dos unidades de codificación 20A, 20B que son cada una la cuarta parte de tamaño comparadas con una LCU 10. Correspondientemente, el borde entre en el tercer fragmento 4 y un cuarto fragmento 5 está en la figura 2 alineado con el borde de dos CUs 30A, 30B, cada una 1/16 de tamaño comparado con una LCU 10.

Sin embargo, la división jerárquica y los comienzos del fragmento alineados con las pequeñas CUs, incluso posiblemente con las SCU, pueden dar lugar incluso a mayores direcciones del fragmento comparándolas con MPEG-4/AVC. Por ejemplo y en el ejemplo expuesto previamente, una imagen de 1280x960 píxeles puede tener 19200 posiciones potenciales de comienzo del fragmento si los comienzos del fragmento se pueden alinear con el borde de las SCUs de 8x8 píxeles. Si las direcciones del fragmento están entonces en forma de coordenadas X e Y necesitarían ser de  $8 + 7 = 15$  bits.

De acuerdo con las realizaciones, la palabra "fragmento" se emplea para indicar una porción codificable y decodificable independientemente de una imagen o trama de vídeo. Una imagen puede por consiguiente estar compuesta de un único fragmento o múltiples fragmentos, es decir, al menos dos fragmentos.

La figura 1 es un diagrama de flujo que ilustra un método de codificar una imagen o trama de vídeo que comprende  $N$  fragmentos. El parámetro  $N$  es un entero positivo igual o mayor que dos. El método se inicia en la etapa S1, que genera una representación codificada del fragmento para cada fragmento en la imagen. Esta representación codificada del fragmento se genera basándose en los valores de píxel de los píxeles en el fragmento de acuerdo con los esquemas de codificación conocidos, tales como intra o inter codificación.

Las siguientes etapas S2 y S3 fijan los llamados indicadores del fragmento para los fragmentos en la imagen. Para mayor detalle, la etapa S2 fija un indicador del fragmento asociado con el primer fragmento de los  $N$  fragmentos en la imagen a un primer valor definido, tal como  $1_{bin}$  o un uno lógico o algún otro símbolo definido, para indicar que el fragmento presente es el primer fragmento de la imagen y que por consiguiente tiene su comienzo del fragmento en una posición definida en la imagen, normalmente la esquina superior izquierda de la imagen. La etapa S3 fija correspondientemente el indicador del fragmento asociado con cada fragmento de los  $N-1$  fragmentos restantes de la imagen a un segundo valor definido, tal como  $0_{bin}$  o un cero lógico o algún otro símbolo definido. Esto quiere decir que los indicadores de los fragmentos se pueden utilizar como información adicional con el fin de diferenciar entre el primer fragmento de la imagen y los fragmentos restantes.

Estos fragmentos restantes necesitan tener direcciones del fragmento asociadas con el fin de habilitar al decodificador para identificar donde se inicia el fragmento dentro del área de la imagen. Esto no es necesario para el primer fragmento, el cual en cambio tiene una posición de comienzo definida dentro de la imagen y comienza preferiblemente en el primer píxel de la imagen en la esquina superior izquierda. Alternativamente, una imagen se puede dividir en regiones en un nivel mayor que los fragmentos; en tal caso, la posición de comienzo definida podría entonces ser el origen de tal región en la imagen.

La etapa S4 por consiguiente genera una dirección del fragmento para cada fragmento de los  $N-1$  fragmentos restantes, excluyendo así el primer fragmento para el cual no se necesita dirección de fragmento de acuerdo con las realizaciones. La dirección del fragmento generada en la etapa S4 para un fragmento permite la identificación de una posición de una primera unidad de codificación del fragmento y así el comienzo del fragmento dentro de la imagen. La primera unidad codificada constituye entonces la primera unidad de codificación del fragmento y una unidad precedente, de acuerdo con un orden de tratamiento definido, es entonces la última unidad codificada del fragmento precedente en la imagen. La primera unidad de codificación podría ser cualquier bloque de píxeles y en MPEG-4/AVC la primera unidad de codificación es un primer macrobloque del fragmento. En correspondencia, con HEVC la primera unidad de codificación podría ser una LCU pero con la división jerárquica puede ser ventajosamente

cualquier unidad de codificación desde el tamaño de la LCU bajando hasta el tamaño de una SCU a menos que se impongan algunas limitaciones de donde se pueden encontrar las posiciones de comienzo del fragmento en la imagen, lo cual se describirá adicionalmente en este documento.

5 El orden de tratamiento en el cual se tratan las unidades de codificación de la imagen, es decir, codificadas y seguidamente descodificadas, podría ser cualquier orden conocido de tratamiento. Un ejemplo de tal orden de tratamiento es el orden de barrido de las tramas o cualquier otro orden de codificación/descodificación, tales como el orden Morton o el orden Z, que se explicarán adicionalmente en ese documento.

10 Las  $N$  representaciones codificadas del fragmento, las  $N-1$  direcciones del fragmento y los  $N$  indicadores del fragmento se emplean entonces para generar una representación codificada de la imagen en la etapa S5. La representación codificada de la imagen tiene normalmente la forma de una secuencia o flujo de bits, aunque se pueden utilizar otros alfabetos de símbolos además del alfabeto binario y que estén dentro del alcance de las realizaciones, tal como el alfabeto hexadecimal o decimal. La figura 6 es una ilustración esquemática de una realización de tal representación codificada de una imagen 50. Generalmente la representación codificada de la imagen 50 comprende dos partes principales para cada fragmento, un encabezamiento 54 del fragmento y los datos codificados 56. El encabezamiento 54 del fragmento comprende normalmente el indicador del fragmento fijado para el fragmento la etapa S2 o en la etapa S3, tal como en forma de palabra código *first\_slice\_in\_pic\_flag*. El encabezamiento 54 del fragmento de cada fragmento restante excepto el primer fragmento en la imagen comprende preferiblemente también la dirección del fragmento generada en la etapa S4, tal como en forma de palabra código *first\_cu\_in\_slice* o la palabra código *slice\_address*. En realizaciones en particular, la información adicional se puede incluir en el encabezamiento 54 del fragmento incluyendo, por ejemplo, el tipo de codificación del fragmento.

Los datos codificados 56 transportan los datos codificados de la imagen de los píxeles en el fragmento, es decir, las representaciones codificadas del fragmento generadas en la etapa S1.

25 La representación codificada 50 puede comprender también opcionalmente o de otra manera estar asociada con un parámetro fijado de la imagen (PPS) y/o por un segundo parámetro fijado de la secuencia (SPS) 52. Los PPS/SPS 52 podrían formar parte de la representación codificada 50 de la imagen. En tal caso, cada representación codificada de la imagen 50 de una secuencia de video podría tener un campo 52 respectivo PPS y/o SPS. En un enfoque alternativo, no todas dichas representaciones codificadas 50 de la imagen de la secuencia de video necesitan transportar el campo 52 PPS y/o SPS. Por ejemplo, la primera representación codificada 50 de la imagen de la secuencia de video podría incluir el campo 52 PPS y/o SPS y entonces tales campos se incluyen sólo en otra representación codificada de la imagen de la secuencia de video si cualquiera de los parámetros en el campo 52 PPS y/o SPS se actualiza o se cambia. Una variante adicional es marcar el campo 52 PPS y/o SPS fuera de la banda con respecto a la representación codificada 50 de la imagen. En tal caso, el campo 52 PPS y/o SPS se podría enviar separadamente de la representación codificada 50 de la imagen pero en tal caso el descodificador puede identificar a qué secuencia de video o representación codificada de la imagen pertenece el campo 52 PPS y/o SPS. Esto se puede lograr incluyendo una sesión, secuencia y/o identificador de la imagen tanto de la representación codificada 50 de la imagen como del campo 52 PPS y/o SPS.

40 Como se ha mencionado anteriormente, los fragmentos son unidades codificables y descodificables independientemente de la imagen. Esto quiere decir, que la generación de las representaciones codificadas del fragmento en la etapa S1, la fijación del indicador del fragmento de las etapas S2, S3 y la generación de la dirección de la etapa S4 se pueden en realizar en serie o al menos parcialmente en paralelo para los diferentes fragmentos en la imagen. Una codificación en paralelo de los fragmentos reduciría normalmente el tiempo total de codificación de la imagen. El método de las etapas S1 a S5 se repite normalmente para cualquiera de las imágenes restantes o tramas de video, tales como una secuencia de video. Adicionalmente, la etapa S2 o las etapas S3/S4 se pueden realizar después, antes o al menos parcialmente en paralelo con la etapa S1.

45 A continuación, se describirán adicionalmente realizaciones de la presente invención en conexión con HEVC como un ejemplo de una codificación y descodificación de video normalizadas a las cuales se pueden aplicar las realizaciones. Esto debe, sin embargo, simplemente verse como un ejemplo ilustrativo de las codificación/descodificación de video normalizadas que se pueden utilizar con las realizaciones y que tales realizaciones no se limitan a ello.

50 De acuerdo con HEVC, una imagen o trama de video comprende múltiples LCU que tienen un tamaño seleccionado con respecto al número de píxeles. Esto significa que cada LCU de la imagen tiene preferiblemente el mismo número de píxeles. Las LCUs podrían ser rectangulares pero son preferiblemente cuadráticas, es decir, comprenden  $M \times M$  píxeles, en donde  $M$  es un entero positivo definido igual a o preferiblemente mayor de dos y preferiblemente  $M=2^m$ , en donde  $m$  es un entero positivo. Ejemplos no limitativos de valores adecuados de  $M$  son 64 o 128. Cada LCU de la imagen puede dividirse potencialmente en múltiples CUs menores teniendo tamaño respectivos menores que el tamaño seleccionado de las LCUs.

En general, la división jerárquica de una LCU implica dividir la LCU en forma de estructura de árbol grado cuatro. Como es bien conocido en la técnica, una estructura de árbol grado cuatro es una estructura de datos en árbol en la cual cada nodo interno tiene exactamente cuatro descendientes. Jerárquicamente dividir la LCU implica por

consiguiente particionar los dos espacios dimensionales de la imagen ocupados por la LCU sub dividiéndola recurrentemente en cuatro cuadrantes o regiones. En una realización preferida, la división recurrente implica la división en cuatro CUs de igual tamaño. De acuerdo con las realizaciones, si una unidad de codificación, es decir, bien una LCU o una CU menor, se divide en el llamado indicador de unidad de codificación dividida asociado con la unidad de codificación se fija a un valor definido, preferiblemente  $1_{bin}$  o a un uno lógico o a cualquier otro símbolo definido, indica que la unidad de codificación está dividida jerárquicamente en múltiples, preferiblemente cuatro, CU menores. De manera correspondiente, si una unidad de codificación divisible, es decir, una unidad de codificación que es mayor que la SCU, no se divide en un indicador de unidad de codificación dividida asociada a la unidad de codificación se fija preferiblemente en cambio a  $0_{bin}$  o a un cero lógico o a cualquier otro símbolo definido. Una unidad de codificación “divisible” se refiere en este documento a una unidad de codificación que es capaz de ser dividida jerárquicamente en múltiples, preferiblemente cuatro, unidades de codificación menores. Generalmente cualquier unidad de codificación excepto las SCU es una unidad de codificación divisible. Aunque una unidad de codificación se puede dividir en unidades de codificación menores no tienen que ser divididas, por ejemplo, si tal división no va a mejorar la calidad de codificación de la imagen.

La división jerárquica de las realizaciones trata preferiblemente LCU por LCU según un orden de tratamiento definido, tal como el orden de barrido de tramas. El orden de barrido de tramas se hace generalmente de izquierda a derecha y de arriba a abajo. Alternativamente, se podría utilizar un orden de codificación/descodificación, tal como el orden Morton o el orden Z. La figura 4 ilustra los principios del orden Morton. Si una LCU se divide en, preferiblemente cuatro CUs de igual tamaño, estas CUs se pueden tratar además en un orden de tratamiento con el fin de seleccionar si se pueden dividir jerárquicamente en, preferiblemente cuatro de igual tamaño, incluso menores CUs. Este orden de tratamiento podría ser el mismo orden que cuando se tratan las LCUs en la imagen. En un enfoque alternativo, las LCU se tratan en el orden de barrido de tramas estando las CUs tratadas en el orden de codificación/descodificación, tales como el orden Morton. Los órdenes de tratamiento presentados anteriormente son simplemente ejemplos de órdenes que se pueden utilizar y que no limitan por ello las realizaciones.

Por consiguiente, se determina para cada unidad de codificación dividir la unidad de codificación adicionalmente en unidades menores de codificación a menos que la unidad de codificación sea la SCU, que no se puede dividir más jerárquicamente. Cada vez que una unidad de codificación se divide en un indicador de la unidad de codificación dividida asociado con la unidad de codificación se fija preferiblemente a uno y si se determina que una unidad de codificación no se va a dividir más en unidades de codificación menores su indicador de unidad de codificación dividida asociado se fija preferiblemente a cero. Una SCU normalmente no necesita tener ningún indicador de unidad de codificación dividida asociado ya que no se puede dividir más.

Esta decisión de dividir o no una unidad de codificación se basa en el tratamiento de la codificación. Por ejemplo, un área de imagen que represente un fondo perfectamente homogéneo se representa más eficientemente utilizando grandes tamaño de CU, tales como LCUs, comparándolo con dividir la imagen en unidades de codificación menores. Sin embargo, áreas de imagen con pequeños detalles o con numerosos detalles pueden generalmente no estar correctamente representadas si se utilizan grandes unidades de codificación. En tal caso, es más eficiente y preferido desde el punto de vista de la calidad de la codificación utilizar varias CUs menores para el área de la imagen. La selección de dividir adicionalmente una CU puede por ello realizarse de acuerdo con las técnicas descritas en la técnica y basándose preferiblemente en el rendimiento de la codificación y en la calidad.

Los indicadores de la unidad de codificación dividida generados durante la codificación de los fragmentos en la etapa S1 de la figura 1 se incluyen normalmente en la porción 56 de los datos codificados de la representación 50 de la imagen codificada como se ilustra en la figura 6.

La figura 5 es un diagrama de flujo que ilustra etapas adicionales del método de codificación de la figura 1. El método comienza en la etapa opcional S10 en la que se determina una granularidad jerárquica para la imagen. La granularidad jerárquica define un nivel jerárquico para los alineamientos de los bordes del fragmento dentro de la imagen. Este nivel jerárquico define y limita el tamaño de una unidad de codificación posible direccionable más pequeña en la cual se puede alinear un comienzo de un fragmento en la imagen. El nivel jerárquico y la granularidad jerárquica determinada definen por tanto el máximo número de posiciones de comienzo del fragmento que son disponibles potencialmente en la imagen y en las cuales se puede posicionar un comienzo del fragmento. Esto quiere decir que la granularidad jerárquica define el número de CUs direccionables dentro de la imagen, en las que el comienzo de un fragmento se puede alinear entre el borde de tal CU direccionable y una CU anterior en la imagen, de acuerdo con un orden de tratamiento definido.

Por ejemplo, una LCU que tenga un tamaño seleccionado de 64x64 píxeles tendría una granularidad del fragmento definiendo un nivel de granularidad de 0 a 3 con un tamaño de la SCU de 8x8 píxeles. En tal caso, un nivel de granularidad de cero indica que los comienzos del fragmento pueden ser sólo alineados con bordes entre las LCUs. Con una imagen de 1280x960 píxeles esto implica  $20 \times 15 = 300$  posibles posiciones de comienzo del fragmento. Si el nivel de granularidad es por el contrario de uno, la menor unidad de codificación posible a la cual se puede alinear un comienzo del fragmento es en cambio de 32x32 píxeles con un total de  $40 \times 30 = 1200$  posiciones posibles del comienzo del fragmento. En correspondencia, un nivel de granularidad de dos significa que hay  $80 \times 60 = 4800$  posibles posiciones del comienzo del fragmento, ya que los comienzos del fragmento se pueden alinear con CUs de 16x16 píxeles o mayores. Finalmente, un nivel de granularidad de tres indica que los comienzos del fragmento se

pueden alinear con los bordes de las SCU dando un total de  $160 \times 120 = 19200$  posiciones posibles de comienzo del fragmento.

Una razón por lo cual se desearía tener la posibilidad de seleccionar entre estos casos desde 220 hasta 14400 posibles posiciones de comienzo del fragmento en el presente ejemplo es que cuanto más se disponga de más posiciones de comienzo del fragmento en la imagen se necesitan más direcciones del fragmento mayores, aumentando como consecuencia los costos de los datos de la imagen codificada. Por ello, si no existen demandas específicas en cuanto a los tamaños de fragmento objetivo, tales como ajustar los datos del fragmento en un único paquete de datos IP, podría ser ventajoso limitar el número de posiciones de comienzo del fragmento en una imagen para reducir por lo tanto la magnitud de datos de la dirección que necesitan ser generados y transmitidos al descodificador.

Una etapa siguiente S11 determina la longitud de la dirección del fragmento para los  $N-1$  fragmentos en la imagen, en la que la longitud es con respecto al número de bits u otros símbolos para la dirección del fragmento. La longitud de la dirección del fragmento depende del número de posiciones de comienzo del fragmento potenciales y del número de unidades de codificación direccionables dentro de la imagen. En el caso de MPEG-41AVC este número de posiciones de comienzo del fragmento depende del tamaño de la imagen ya que los comienzos del fragmento sólo se pueden alinear en los bordes del macrobloque. Esto quiere decir que dado el tamaño total de la imagen, se puede calcular el número de posibles soluciones de comienzo del fragmento dado el tamaño fijo del macrobloque. La longitud de la dirección del fragmento se puede entonces calcular a partir de este número, tal como  $\log_2(P)$  o  $\log_2(P-1)$ , en donde  $P$  representa el número de las posibles posiciones de comienzo del fragmento y por lo tanto el número total de posibles direcciones del fragmento en la imagen. El tamaño de la imagen se incluye normalmente en un campo de encabezamiento o asociado a las representaciones codificadas de la imagen o se podría encontrar en el previamente mencionado campo 52 PPS o SPS de o asociado con la representación 50 codificada de la imagen, véase figura 6.

En HEVC, la longitud de la dirección del fragmento se determina preferiblemente en la etapa S11 basándose en la granularidad jerárquica determinada en la etapa S10 para la imagen. La granularidad jerárquica se puede utilizar entonces para definir el tamaño de las unidades direccionables de codificación y por ello el tamaño de la menor unidad posible de codificación a la cual se puede alinear un comienzo del fragmento. Esta información de la granularidad se emplea preferiblemente junto con la información del tamaño de la imagen o del número total de LCUs en la imagen, con el fin de determinar la longitud de la dirección del fragmento en la etapa S11.

En una realización, el tamaño de las LCU en la imagen podría ser predefinido y por ello conocido para el codificador y el descodificador. Por ejemplo, el tamaño de la LCU podría ser  $128 \times 128$  o  $64 \times 64$  píxeles. No se necesita por lo tanto determinar o marcar el tamaño de la LCU. En correspondencia, el tamaño de las SCUs en la imagen se podría predefinir. Ejemplo de tales tamaños de tales SCUs fijas y predefinida que se pueden utilizar son  $16 \times 16$  o  $8 \times 8$  píxeles.

En realizaciones alternativas, el proceso de codificación puede determinar adicionalmente el tamaño de la LCU y/o el tamaño de la SCU a emplear para la(s) imagen(es) actual(es). Esto podría ser beneficioso para adaptar por lo tanto estos tamaños de LCU y/o de SCU a las características particulares de la imagen presente. Por ejemplo, para algunas imágenes que fueran básicamente una vista uniforme homogénea del fondo, se podría preferir mayores tamaños de LCU y SCU y llevarían a una codificación más eficiente si se comparasen con las imágenes con un gran número de pequeños detalles en las que se podrían preferir tamaños de LCU y SCU menores.

En una realización, el tamaño de la LCU y/o el tamaño de la SCU se determinan por lo tanto durante la codificación, tal como se basan en los valores de píxel de la imagen. Una notificación del tamaño determinado de la LCU y/o una notificación del tamaño determinado de la SCU se asocian entonces con la representación codificada de la imagen. La asociación de la(s) notificación(es) y la representación codificada de la imagen se pueden realizar de acuerdo con varias realizaciones. Por ejemplo, se pueden incluir las notificaciones en la representación codificada de la imagen. Una alternativa es incluir las notificaciones en los PPS o en los SPS.

El tamaño de la SCU podría definirse entonces basándose en el parámetro  $\log_2\_min\_coding\_block\_size\_minus3$  y preferiblemente calculando el parámetro  $Log2MinCUSize$  como  $Log2MinCUSize = \log_2\_min\_coding\_block\_size\_minus3 + 3$ . Este parámetro  $Log2MinCUSize$  se emplea entonces como representación del tamaño de la SCU y da el tamaño de la SCU  $MinCUSize = (1 \ll Log2MinCUSize)$ , en la que  $\ll$  indica un operador de desplazamiento a la izquierda. Dependiendo del valor del parámetro  $Log2MinCUSize$  y por consiguiente del parámetro  $\log_2\_min\_coding\_block\_size\_minus3$  el tamaño de la SCU podría ser entonces de 8 o 16 como ejemplos ilustrativos.

El tamaño de la LCU se determina preferiblemente en relación con el tamaño de la SCU definiendo el parámetro  $\log_2\_diff\_max\_min\_coding\_block\_size$ . Con mayor detalle, el parámetro  $Log2MaxCUSize$  se calcula como  $Log2MaxCUSize = \log_2\_min\_coding\_block\_size\_minus3 + 3 + \log_2\_diff\_max\_min\_coding\_block\_size$ . Este parámetro  $Log2MaxCUSize$  se emplea entonces como la representación del tamaño de la LCU y da el tamaño de la LCU  $MaxCUSize = (1 \ll Log2MaxCUSize)$ . Dependiendo del valor del parámetro  $Log2MaxCUSize$  y por lo tanto de los



parámetros *log2\_min\_coding\_block\_size\_minus3* y *log2\_diff\_max\_min\_coding\_block\_size* la longitud de la LCU podría ser entonces de 64 o 128 como ejemplos ilustrativos.

Las notificaciones del tamaño de la SCU y del tamaño de la LCU podrían por lo tanto ser los parámetros *log2\_min\_coding\_block\_size\_minus3* y *log2\_diff\_max\_min\_coding\_block\_size*.

- 5 En una realización alternativa, el tamaño de la LCU no se determina en relación con el tamaño de la SCU. Esto significa que no se necesita el parámetro de la SCU para determinar el tamaño de la LCU.

La dirección del fragmento generada para cada fragmento excepto para el primer fragmento en la imagen podría definir la posición del comienzo del fragmento y la primera CU del fragmento como un simple número. Las diferentes posiciones posibles del comienzo del fragmento y las unidades de codificación direccionables se numeran entonces desde cero en adelante. Por ejemplo, una imagen de 1280×960 píxeles tiene 4800 únicas posiciones posibles de comienzo del fragmento si la granularidad jerárquica define que el tamaño de la menor unidad posible de codificación a la cual se puede alinear un comienzo del fragmento en la imagen es de 16×16 píxeles. Estas posiciones se podrían entonces numerar desde cero hasta 4799, necesitando por lo tanto direcciones del fragmento de 13 bits

- 15 Una alternativa es manejar las coordenadas X e Y separadamente. Con el ejemplo anterior, la coordenada X va desde 0 a 79 y la coordenada Y va desde 0 a 59, necesitando por lo tanto 7 más 6 bits para las direcciones del fragmento.

Una alternativa adicional es determinar la dirección del fragmento de modo que las coordenadas de la LCU y las coordenadas de la sub-LCU se puedan recuperar de ella. En tal caso, las coordenadas de la posición de una LCU dentro de la imagen se determinan para un fragmento. El comienzo del fragmento y la primera CU del fragmento se posicionan entonces en la imagen dentro de esta LCU. Las coordenadas se fijan entonces con relación a un origen global o punto de arranque, normalmente la esquina superior izquierda de la imagen. Las coordenadas de la LCU podrían ser entonces las coordenadas de la LCU con relación a este origen global, tal como con respecto a un número de la LCU o con respecto a las coordenadas X e Y como se mencionó anteriormente. También se determinan las coordenadas de la posición de la primera CU y por consiguiente el comienzo del fragmento dentro de la LCU. Estas coordenadas son entonces relativas a un origen local o punto de arranque, normalmente la esquina superior izquierda de la LCU. Estas coordenadas sub LCU podrían también ser de la forma de un número o con respecto a las coordenadas X e Y.

La dirección de fragmento se genera entonces basándose en las coordenadas LCU y en las coordenadas sub-LCU y en la granularidad jerárquica. La granularidad jerárquica se emplea cuando se definen las coordenadas LCU restringiendo y definiendo las posibles posiciones de comienzo para el fragmento y la primera CU del fragmento dentro de la LCU.

En una realización, se genera una primera o representación LCU basándose en las coordenadas determinadas de la LCU y se genera una segunda o representación de la sub-LCU basándose en las coordenadas de la sub-LCU. La dirección del fragmento podría comprender entonces estas dos representaciones. Alternativamente, la dirección del fragmento se genera de tal modo que las coordenadas de la LCU y las coordenadas de la sub-LCU se puedan determinar o calcular a partir de la dirección del fragmento.

La derivación de las coordenadas de la LCU o de la sub-LCU se puede realizar de acuerdo con lo que sigue como un ejemplo ilustrativo pero no limitativo.

40 La granularidad jerárquica determinada en la etapa S10 viene definida por la palabra código *slice\_granularity*. *Slice\_granularity* es normalmente un valor de 2-bits en el intervalo desde  $00_{bin}=0$  hasta un máximo de  $11_{bin}=3$ . Esto posibilita cuatro niveles jerárquicos diferentes. Si se necesitan simplemente dos niveles jerárquicos *slice\_granularity* podría por el contrario ser un valor de 1-bit. En correspondencia, para más de cuatro niveles jerárquicos se necesita un parámetro *slice\_granularity* de 3 bits o más. Alternativamente es posible la codificación de longitud variable para marcar la granularidad jerárquica.

La palabra código *slice\_granularity* se define opcionalmente no mayor que la menor de las otras dos palabras código determinadas durante la codificación de la imagen o de la secuencia de video: *Log2MaxCUSize-4* y *log2\_diff\_max\_min\_coding\_block\_size*. La palabra código *slice\_granularity* se emplea entonces, durante la decodificación, para calcular el parámetro *Slice\_Granularity* como  $SliceGranularity=(slice\_granularity<<1)$ .

50 La dirección del parámetro generada durante la codificación es la palabra código *slice\_address*. Esta palabra código define la dirección del fragmento en la resolución de la granularidad del fragmento en la cual comienza el fragmento. La longitud de la dirección del fragmento, es decir, *slice\_address*, se determina como se mencionó anteriormente basándose en la granularidad jerárquica. En una realización en particular, la longitud de *slice\_address* con respecto al de número de bits es igual a  $ceil(log_2(NumLCUsInPicture) + SliceGranularity)$ .

55 El parámetro *NumLCUsInPicture* define el número total de LCUs en la imagen y se determina basándose en el tamaño de la imagen y basándose en el tamaño de las LCUs, los cuales son fijos o determinados como se mencionó

anteriormente. *NumLCUsInPicture* se puede entonces incluir en la representación codificada de la imagen o asociarse a ella, tal como en un campo PPS o SPS. Alternativamente, un decodificador puede por sí mismo calcular el parámetro *NumLCUsInPicture* basándose en el tamaño de la LCU ( $\log_2\_min\_coding\_block\_size\_minus3$  y  $\log_2\_diff\_max\_min\_coding\_block\_size$ ) y en el tamaño total de la imagen, que se le marca al decodificador en o asociado a la representación codificada de la imagen.

La parte LCU de la dirección del fragmento de acuerdo con un orden de tratamiento, tal como un orden de barrido de tramas, se define entonces como  $LCUAddress=(slice\_address \gg SliceGranularity)$ , en donde  $\gg$  define un operador de desplazamiento a la derecha. La parte sub-LCU de la dirección del fragmento de acuerdo con un orden de tratamiento, tal como el orden Morton, se calcula como  $GranularityAddress=slice\_address - (LCUAddress \ll SliceGranularity)$ .

Y la dirección del fragmento se determina entonces basándose en *LCUAddress* y la *GranularityAddress* como  $SliceAddress=(LCUAddress \ll (\log_2\_diff\_max\_min\_coding\_block\_size \ll 1)) + (GranularityAddress \ll ((\log_2\_diff\_max\_min\_coding\_block\_size \ll 1) - SliceGranularity))$ .

La dirección del fragmento generada para los restantes fragmentos excluyendo el primer fragmento en la imagen podría ser una dirección de longitud fija, en la que la longitud de la dirección queda fijada por la imagen y depende del tamaño de una unidad de codificación lo menor posible en la cual se puede alinear un comienzo de un fragmento y del tamaño total de la imagen. Una alternativa sería utilizar una representación de longitud variable. Como ejemplo de un código de longitud variable que se puede utilizar es la codificación universal de longitud variable (UVLC) como se menciona en Lee y Kuo, Complexity Modeling of H.264/AVC CAVLC/UVLC Entropy Decoders, IEEE International Symposium on Circuits and Systems (ISCAS2008), 2008, pp. 1616-1619. Brevemente, UVLC utiliza el código Exp-Golomb (EG). El código EG para un valor entero sin signo C es  $[P \text{ ceros}][1][info]$ , en donde  $P=floor(\log_2(C+1))$  e  $info=C+1-2^P$ .

La dirección del fragmento no sólo define la posición de la primera CU y por lo tanto el comienzo de un fragmento sino que define adicionalmente el tamaño de la CU mayor posible que pueda ocupar la posición en la imagen definida por la dirección del fragmento. Esto quiere decir que este fragmento depende de la posición determinada por la dirección del fragmento. Aunque, la dirección del fragmento da el tamaño de la CU mayor posible que pueda ocupar la posición, el tamaño de la primera CU no necesita ser igual al tamaño de la mayor CU posible que pueda ocupar la posición. Las figuras 3A y 3B ilustran este concepto. En las figuras, los números de referencia 2, 3 indican dos diferentes fragmentos en una imagen y la línea en negrita define el borde entre los dos fragmentos 2, 3. El borde del fragmento cae en estos ejemplos dentro del área de la imagen ocupada por una LCU 10. El número de referencia 20 indica una CU obtenida para una granularidad de 1 cuando la LCU 10 está dividida jerárquicamente en cuatro CUs 20. Con una granularidad de 2 esta CU 20 puede dividirse jerárquicamente en cuatro CUs 30 menores. En las figuras 3A y 3B se muestra un caso en el que con una granularidad de 3 una CU 30 se puede dividir en 4 SCUs 40.

En la figura 3A la primera CU del fragmento 3 está indicada por 30, mientras que en la figura 3B está indicada por 40B. Los números de referencia 40(figura 3A) y 40A (figura 3B) indican una CU que precede a la LCU 10 de acuerdo con el orden de tratamiento definido, que en este ejemplo es el orden Morton. Tanto en la figura 3A como en la figura 3B el comienzo del fragmento y la posición de las primeras CU 30, 40B son los mismos aunque las primeras CU 30, 40B tengan diferentes tamaños en los dos ejemplos. La dirección del fragmento es, sin embargo, normalmente la misma en estos ambos casos y el tamaño de la CU 30 mayor posible que pueda ocupar la posición pertinente es la misma. Los dos casos se pueden diferenciar complementando la dirección del fragmento con información adicional con respecto a los llamados indicadores de unidades de codificación divididas.

En un ejemplo, supóngase que el tamaño de una LCU es 128x128 píxeles y un tamaño correspondiente de una SCU es de 16x16 píxeles. Supóngase además que las LCUs 10 de las figuras 3A y 3B constan de dos fragmentos 2, 3; entonces la representación codificada se podría definir como:

*Slice\_header\_syntax()* // fragmento 2 en la figura 3A – incluye la información de la dirección

*split\_coding\_unit\_flag=1* // divide la LCU de 128 x128 en cuatro CUs de 64x64

*split\_coding\_unit\_flag=0* // la primera CU de 64x64 no se divide más

*code of first 64x64 CU* // la representación codificada de los valores de píxel de la primera CU de 64x64

*split\_coding\_unit\_flag=1* // divide la segunda CU de 64x64 en 4 CUs de 32x32

*split\_coding\_unit\_flag=0* // la primera CU de 32x32 no se divide más

*code of first 32x32 CU* // la representación codificada de los valores de píxel de la primera CU de 32x32

*split\_coding\_unit\_flag=1* // divide la segunda CU de 32x32 en 4 SCUs, no se necesitan indicadores de unidad de codificación dividida adicionales

- code of first SCU* // la representación codificada de los valores de píxel de la primera SCU
- code of second SCU* // la representación codificada de los valores de píxel de la segunda SCU
- code of third SCU* // la representación codificada de los valores de píxel de la tercera SCU
- code of fourth SCU* // la representación codificada de los valores de píxel de la cuarta SCU
- 5 *split\_coding\_unit\_flag=1* // divide la tercera CU de 32×32 CU en cuatro SCUs, no se necesitan indicadores de unidad de codificación dividida adicionales
- code of first SCU* // la representación codificada de los valores de píxel de la primera SCU
- code of second SCU* // la representación codificada de los valores de píxel de la segunda SCU
- code of third SCU* // la representación codificada de los valores de píxel de la tercera SCU
- 10 *code of fourth SCU* // la representación codificada de los valores de píxel de la cuarta SCU
- split\_coding\_unit\_flag=1* // divide la cuarta CU de 32×32 CU en cuatro SCUs, no se necesitan indicadores de unidad de codificación dividida adicionales
- code of first SCU* // la representación codificada de los valores de píxel de la primera SCU
- code of second SCU* // la representación codificada de los valores de píxel de la segunda SCU
- 15 *code of third SCU* // la representación codificada de los valores de píxel de la tercera SCU
- code of fourth SCU* // la representación codificada de los valores de píxel de la cuarta SCU
- split\_coding\_unit\_flag=1* // divide la tercera CU de 64×64 CU en cuatro CUs de 32×32
- split\_coding\_unit\_flag=1* // divide la primera CU de 32×32 CU en cuatro SCUs, no se necesitan indicadores de unidad de codificación dividida adicionales
- 20 *code of first SCU* // la representación codificada de los datos del píxel de la primera SCU
- code of second SCU* // la representación codificada de los datos del píxel de la segunda SCU
- code of third SCU* // la representación codificada de los datos del píxel de la tercera SCU
- code of fourth SCU* // la representación codificada de los valores de píxel de la precedente CU 40 en la figura. 3A
- Slice\_header\_syntax()* // el fragmento 3 en la figura 3A - incluye la dirección de la información
- 25 *split\_coding\_unitflag=0* // el indicador de la unidad de codificación dividida de la primera CU se pone a cero ya que el tamaño de la primera CU es igual al tamaño de la mayor CU posible CU que puede ocupar la posición de la imagen definida por la información generada de la dirección. – La mayor CU posible en esta dirección es 32 por 32 y no se necesitan dividir los indicadores en sentido descendente hasta 32 por 32. En lugar de ello, el tamaño 32 por 32 se deriva de la dirección y de la granularidad.
- 30 *code of first CU* // la representación codificada de los valores de píxel de la CU 30 en la figura 3A
- split\_coding\_unit\_flag=1* // divide la CU de 32×32 en cuatro SCUs, no se necesita dividir adicionalmente los indicadores de la unidad de codificación
- code of first SCU* // la representación codificada de los valores de píxel de la primera SCU
- code of second SCU* // la representación codificada de los valores de píxel de la segunda SCU
- 35 *code of third SCU* // la representación codificada de los valores de píxel de la tercera SCU
- code of fourth SCU* // la representación codificada de los valores de píxel de la cuarta SCU
- split\_coding\_unit\_flag=1* // divide la CU de 32×32 en cuatro SCUs, no se necesitan adicionalmente los indicadores divididos de la unidad de codificación
- code of first SCU* // la representación codificada de los valores de píxel de la primera SCU
- 40 *code of second SCU* // la representación codificada de los valores de píxel de la segunda SCU
- code of third SCU* // la representación codificada de los valores de píxel de la tercera SCU

*code of fourth SCU* // la representación codificada de los valores de píxel de la cuarta SCU

*split\_coding\_unit\_flag=0* // no hay división adicional de la CU de 64x64

*code of 64x64 CU* // la representación codificada de los valores de píxel de la primera CU de 64x64

5 En la realización ilustrada en la figura 3B el código del primer fragmento 2 sería el mismo que anteriormente, mientras que para el segundo fragmento 3 el código en cambio vendría dado por:

*Slice\_header\_syntax()* // el fragmento 3 en la figura 3B – incluye la información de la dirección

*split\_coding\_unit\_flag=1* // el indicador de la unidad de codificación dividida de la primera CU se pone a uno ya que el tamaño de la primera CU es menor que el tamaño de la mayor CU posible que pueda ocupar la posición en la imagen definida por la información de la dirección generada, no necesitándose indicadores adicionales de la unidad de codificación dividida.

10

*code of first CU* // la representación codificada de los valores de píxel de la primera CU 40B in Fig. 3B

*code of second SCU* // la representación codificada de los valores de píxel de la segunda SCU

*code of third SCU* // la representación codificada de los valores de píxel de la tercera SCU

*code of fourth SCU* // la representación codificada de los valores de píxel de la cuarta SCU

15 *split\_coding\_unit\_flag=1* // divide la CU de 32x32 en cuatro SCUs, no se necesitan adicionalmente los indicadores divididos de la unidad de codificación

*code of first SCU* // la representación codificada de los valores de píxel de la primera SCU

*code of second SCU* // la representación codificada de los valores de píxel de la segunda SCU

*code of third SCU* // la representación codificada de los valores de píxel de la tercera SCU

20 *code of fourth SCU* // la representación codificada de los valores de píxel de la cuarta SCU

*split\_coding\_unit\_flag=1* // divide la CU de 32x32 en cuatro SCUs, no se necesitan adicionalmente los indicadores divididos de la unidad de codificación

*code of first SCU* // la representación codificada de los valores de píxel de la primera SCU

*code of second SCU* // la representación codificada de los valores de píxel de la segunda SCU

25 *code of third SCU* // la representación codificada de los valores de píxel de la tercera SCU

*code of fourth SCU* // la representación codificada de los valores de píxel de la cuarta SCU

*split\_coding\_unit\_flag=0* // no hay división adicional de la CU de 64x64

*code of 64x64 CU* // la representación codificada de los valores del píxel de la primera CU de 64x64

30 La figura 7 es un diagrama de flujo que ilustra un método de descodificar una representación codificada de imagen de una imagen que consta de múltiples fragmentos. El método comienza en la etapa S20, en la que un indicador del fragmento asociado a un fragmento se recupera a partir de la presentación codificada de la imagen, preferiblemente de un encabezamiento del fragmento asignado al fragmento actual en la presentación codificada de la imagen. Una etapa siguiente S21 genera una representación codificada de los valores de píxel de los píxeles en el fragmento basándose en una representación codificada del fragmento asociada al fragmento y recuperada de la representación

35 codificada de la imagen, normalmente de la porción de datos codificados de la misma. Las representaciones descodificadas de los valores de píxel se generan de acuerdo con las técnicas de descodificación habituales, tales como modos de inter o intra descodificación.

El valor de píxel como se utiliza en este documento indica cualquier valor de una propiedad del píxel asignada a un píxel. En realizaciones típicas para HEVC el valor del píxel es un valor del color. Se conocen diferentes formatos de color en la técnica y se pueden utilizar de acuerdo con las realizaciones. Por ejemplo, un valor del color podría comprender tanto el componente de luminancia como el componente de crominancia, normalmente un valor para la luminancia y dos componentes para la crominancia. Un valor de píxel podría por lo tanto ser un valor de luminancia de un píxel, un valor de crominancia de un píxel o de hecho ambos valores de luminancia y crominancia. Otro ejemplo de un formato común del color es el llamado formato RGB, que representa Rojo-Verde-Azul. Un valor de color comprende entonces los valores de rojo, verde y azul. Un valor de píxel podría ser entonces un valor RGB, un valor del rojo, un valor del verde y un valor del rojo. También se conocen variantes del formato RGB, tales como RGBA y se pueden utilizar de acuerdo con las realizaciones.

40

45

En realidad, las realizaciones no tienen que estar necesariamente limitadas al uso de los valores del color como valores de píxel sino que también se pueden aplicar a otros valores conocidos de píxeles incluyendo valores en la escala de grises, valores normales, por ejemplo valores de coordenadas X, Y, Z etcétera.

5 El indicador del fragmento recuperado en la etapa S20 se emplea entonces en la etapa S22 con el fin de determinar si el fragmento actual es el primer fragmento en la imagen y que por ello no tiene que tener asociada ninguna dirección de fragmento o si el fragmento actual no es el primer fragmento y por lo tanto se necesita una dirección de fragmento para el propio fragmento.

10 Si la etapa S22 determina que el fragmento actual es en realidad el primer fragmento, es decir, cuando el indicador del fragmento tiene el valor de uno, el método continúa hasta la etapa S23. La etapa S23 asigna simplemente los valores de píxel generados en la tapa S21 para una primera porción de la imagen que comienza con el comienzo definido del fragmento en la imagen, normalmente la esquina superior izquierda de la imagen. Los valores de píxel se asignan normalmente a los píxeles en un orden definido de tratamiento, tal como el previamente mencionado Morton o un orden de barrido de tramas. En una realización típica aplicada a HEVC, las unidades de codificación menores que la LCU se tratan según el orden Morton mientras que las LCU de la imagen se tratan en el orden de barrido de tramas. Esto implica que la descodificación comienza con la primera LCU del fragmento y entonces si esta LCU se divide en CUs menores, estas menores CUs se descodifican según el orden Morton. Una vez que la LCU ha sido descodificada el proceso continúa con la siguiente LCU de acuerdo con el orden de barrido de tramas y cualesquiera LCU menores de esta siguiente LCU se descodifican según el orden Morton.

20 En una realización en particular, la etapa S22 en realidad se realiza antes que la etapa S21 con el fin de determinar que el fragmento presente es verdaderamente el primer fragmento en la imagen. Entonces los datos codificados de la presentación de la imagen codificada pertenecientes al fragmento actual se descodifican y se asignan a la unidad de codificación de píxeles por unidad de codificación. Esto significa que las etapas S21 y S23 forman entonces un bucle que continúa a través de las diferentes CUs del fragmento y se descodifica cada CU una a una y se asigna el valor de píxel a los píxeles CU por CU de acuerdo con el orden de tratamiento mencionado anteriormente.

25 Si la etapa S22 en cambio determina que el fragmento presente no es el primer fragmento de la imagen basándose en el valor del indicador asociado del fragmento, tal como que tiene un valor de cero, el método continúa hasta la etapa S24. La etapa S24 recupera la dirección del fragmento para el fragmento a partir de la presentación de la imagen codificada, normalmente del encabezamiento del fragmento de ese fragmento. La dirección del fragmento se emplea con el fin de identificar el comienzo del fragmento dentro de la imagen y por ello la porción de la imagen que pertenece al fragmento. Una etapa siguiente S25 asigna entonces los valores de píxel a los píxeles en la porción identificada de la imagen para generar por ello un fragmento descodificado.

30 Similarmente a las etapas S21 y S23 anteriores, las etapas S22 y S24 se pueden realizar antes de las etapas S21 y S25 para deducir primero por ello que el fragmento presente no es el primero e identificar entonces y leer la dirección del fragmento a partir de la representación codificada de la imagen. A partir de ahí el comienzo del fragmento se identifica basándose dentro de la imagen y la descodificación de los datos codificados para la imagen puede comenzar. La descodificación puede proceder CU por CU y asignar entonces los valores de píxel descodificados a los píxeles en la CU actual antes de continuar a la siguiente CU de acuerdo con el orden de tratamiento.

40 En un enfoque alternativo, la etapa S22 se realiza antes de la etapa S21. Así, la etapa S22 investiga si el indicador del fragmento está o no fijado y procede entonces a generar la representación descodificada de los valores de píxel y los asigna como se indicó en la tapa S23 o primero los recupera y utiliza la información de la dirección en la etapa S24 con el fin de identificar qué porción de la imagen asignar a los valores de píxel en la etapa S25.

45 Una vez que todos los datos codificados de un fragmento se han descodificado y asignado a la porción del píxel identificada para el fragmento la etapa S23 o S25, el método finaliza o continúa además hasta un siguiente fragmento de la imagen actual a otro fragmento de otra imagen en una secuencia de video. En tal caso, el método de la figura 7 se repite para este u otro fragmento.

50 Sin embargo, en realizaciones preferidas que reducen el tiempo total de descodificación de una imagen, se pueden descodificar en paralelo múltiples fragmentos. En tal caso, el método de la figura 7 se realiza para cada uno de estos fragmentos y se toma por consiguiente una decisión en la etapa S22 para cada uno de los fragmentos basándose en el respectivo indicador del fragmento del propio fragmento a descodificar.

55 La figura 8 es un diagrama de flujo que ilustra etapas adicionales del método de la figura 7. El método continúa desde la etapa S22 de la figura 7, que dedujo que el fragmento presente no es el primer fragmento de la imagen basándose en su indicador del fragmento asociado. Una etapa siguiente S30 recupera la información de la granularidad jerárquica para la representación codificada de la imagen. Como se describió previamente en este documento, la información de la gradualidad se puede incluir en la representación codificada de la imagen y se recupera a continuación de ella en la etapa S30. Alternativamente, la información de la granularidad puede haberse incluido en una representación codificada de la imagen previamente recibida relativa a una misma secuencia de video. En tal caso, la información de la granularidad se recupera de ella y se almacena para un uso posterior cuando

se descodifiquen las siguientes representaciones de la imagen codificada. La información de la granularidad podría también haberse enviado separadamente de cualquier representación de la imagen codificada, tal como un campo separado PPS o SPS. Los identificadores de la sesión, de la imagen o de la secuencia se pueden entonces utilizar para identificar la información pertinente de la granularidad para la presente representación codificada de la imagen.

5 Una etapa siguiente opcional S31 recupera la información del número de LCUs en la imagen presente. Esta información podría simplemente identificar el número de tales LCUs o podría ser utilizada por el descodificador para calcular el número de LCUs. Por ejemplo, la palabra código *NumLCUsInPicture* podría recuperarse de la representación codificada de la imagen o de la información del encabezamiento global, tal como los campos PPS o SPS.

10 Alternativamente, *NumLCUsInPicture* se calcula basándose en la información del tamaño total de la imagen, según se haya recuperado de la representación codificada de la imagen o del encabezamiento global, y de la información del tamaño de la LCU, es decir, las palabras código mencionadas previamente *log2\_min\_coding\_block\_size\_minus3* y *log2\_diff\_max\_min\_coding\_block\_size*.

15 Una etapa siguiente S32 determina la longitud de la dirección del fragmento del fragmento actual basándose en la información de la granularidad jerárquica y basándose preferiblemente en el número de LCUs en la imagen. En una realización en particular, la longitud de la dirección del fragmento viene definida como  $\text{ceil}(\log_2(\text{NumLCUsInPicture}) + \text{SliceGranularity})$ . Así, en una realización en particular la longitud de la dirección del fragmento se determina basándose en la información del número LCUs en la imagen actual. El parámetro *SliceGranularity* se obtiene preferiblemente directamente a partir de la información de la granularidad *slice\_granularity* como  $\text{SliceGranularity} = (\text{slice\_granularity} \ll 1)$ .

20 El método continúa seguidamente en la etapa S24 de la figura 7, en la que la dirección del fragmento del fragmento actual se recupera a partir de la presentación de la imagen codificada basándose en la información de la longitud de la dirección del fragmento como se ha determinado en la etapa S32. Así, esta longitud se emplea con el fin de identificar qué bits o símbolos de la representación codificada de la imagen que define la dirección del fragmento definiendo la longitud de la dirección de fragmento, la cual preferiblemente tiene un punto de comienzo fijo en el encabezamiento del fragmento del propio fragmento pero en la cual el punto de finalización depende de la longitud de la dirección del fragmento.

25 En el caso de que las realizaciones presentes se apliquen por encima de H.2641MPEG-4 AVC no está disponible la información de la granularidad y por consiguiente se puede omitir la etapa S30. La etapa S31 recupera la información de o permite la determinación del número de macrobloques en la imagen y en los que se emplea esta información en la etapa S32 con el fin de determinar la longitud de la dirección del fragmento.

30 La figura 9 es un diagrama de flujo que ilustra una realización en particular para identificar la posición de la primera unidad de codificación del fragmento y por ello el comienzo del fragmento dentro de la imagen. El método continúa desde la etapa S24 de la figura 7. Una etapa siguiente S40 determina una dirección de la LCU que representa una posición de una LCU dentro de la imagen en la cual están presentes la primera CU y el comienzo del fragmento. La etapa S40 utiliza la dirección del fragmento con el fin de determinar la dirección de la LCU. Por ejemplo, el parámetro *LCUAddress* se puede determinar como  $\text{slice\_address} \gg \text{SliceGranularity}$ , en el que *slice\\_address* representa la dirección del fragmento. Correspondientemente una etapa siguiente S41 determina la dirección de una sub-LCU que representa una posición de la primera CU dentro de la LCU identificada en la etapa S40. Esta dirección de la sub-LCU se determina también basándose en la dirección del fragmento. Por ejemplo, el parámetro *GranularityAddress* se determina como  $\text{slice\_address} - (\text{LCUAddress} \ll \text{SliceGranularity})$ .

35 Las partes LCU y sub-LCU se pueden emplear entonces para calcular la dirección final del fragmento como  $(\text{LGUIAddress} \ll (\log_2\_diff\_max\_min\_coding\_block\_size \ll 1)) + (\text{GranularityAddress} \ll ((\log_2\_diff\_max\_min\_coding\_block\_size \ll 1) - \text{SliceGranularity})$  que se emplea en la etapa S42 con el fin de identificar la porción de la imagen que pertenece al fragmento presente. Así, esta porción comienza con el inicio del fragmento y la primera CU identificada basándose en la dirección del fragmento y continúa a continuación de acuerdo con el orden de tratamiento a través de la imagen hasta que todos los datos codificados se hayan descodificado y asignados a las CUs de la imagen.

40 En realizaciones alternativas, la dirección del fragmento recuperada a partir de la representación de la imagen codificada se emplea directamente para identificar el comienzo del fragmento y la primera unidad de codificación. La dirección de fragmento podría entonces corresponder con el número de posiciones de comienzo del fragmento o las CUs direccionables en las cuales comienza el fragmento. La dirección del fragmento puede entonces ser un índice en una lista de todas las posibles direcciones en el orden de las codificación/descodificación. Una variante adicional es recuperar o calcular a partir de la dirección del fragmento las coordenadas de la LCU y las coordenadas de la sub-LCU como se describió anteriormente en este documento.

50 La dirección del fragmento no sólo define la posición de la primera CU del fragmento y el comienzo del fragmento sino que define preferiblemente también el tamaño de la primera CU. Así, el tamaño de la primera CU se determina basándose al menos parcialmente en la dirección del fragmento. Con más detalle, la dirección del fragmento

dictamina el mayor tamaño posible en términos de número de píxeles que puede tener la primera CU. Esto significa que la primera CU puede tener un tamaño igual a este posible mayor tamaño o un tamaño menor que el mayor tamaño posible. En el último caso, se emplea adicionalmente un indicador de la unidad de codificación dividida añadido a la dirección de fragmento con el fin de definir el tamaño correcto de la primera CU, lo cual se describirá adicionalmente en este documento.

Por ejemplo, la primera CU puede estar asociada con un indicador de la unidad de codificación dividida incluido en la representación codificada de la imagen, normalmente en la porción de los datos codificados. El valor del indicador de la unidad de codificación dividida se emplea entonces junto con la dirección del fragmento con el fin de definir el tamaño correcto de la primera CU. Así, si el indicador de la unidad de codificación dividida se fija a un valor definido, preferiblemente a uno, el tamaño de la primera CU es menor que el tamaño de la mayor CU posible que pueda ocupar la posición dentro de la imagen definida basándose en la dirección de fragmento, véase la figura 3B. Sin embargo, si el indicador de la unidad de codificación dividida se fija a otro valor definido, preferiblemente a cero, el tamaño de la primera CU es igual al tamaño de la mayor CU posible que pueda ocupar la posición en la imagen definida por la dirección del fragmento, véase la figura 3A.

Es posible que la primera CU esté asociada a múltiples indicadores de la unidad de codificación dividida. Por ejemplo, si el tamaño de la mayor CU posible es de 32x32 píxeles, en la que el tamaño de la primera CU es de 8x8 píxeles con un tamaño de una LCU y de una SCU de 64x64 píxeles y de 8x8 píxeles, el código podría ser:

```
split_coding_unit_flag=1 // la CU de 32x32 se divide en CUs de 16x16
```

```
split_coding_unit_flag=1 // la primera CU de 16x16 se divide en SCUs, no necesitándose un indicador de la unidad de codificación dividida adicional ya que se ha alcanzado ahora el mayor tamaño de la primera CU y esto que es también el tamaño de la SCU implica que ahora es posible una división adicional.
```

En algunas realizaciones, el tamaño de la primera CU se puede determinar únicamente basándose en la dirección del fragmento sin utilizar ningún indicador de la unidad de codificación dividida como información adicional. Esto es posible cuando el tamaño de la mayor CU posible que pueda ocupar la posición dentro de la imagen definida basándose en la dirección del fragmento es igual al tamaño de la SCU. En tal caso, no es posible dividir esta mayor CU posible adicionalmente ya que es en realidad una SCU.

La figura 10 es un diagrama esquemático de bloques de un codificador o dispositivo 100 para codificar una imagen que consta de múltiples fragmentos. El dispositivo 100 comprende un generador de la representación 110 configurado para generar una representación del fragmento codificado respectivo para cada fragmento en la imagen basándose en los valores de píxel de los píxeles en el fragmento. El generador de la representación 110 realiza la codificación del píxel de acuerdo con los esquemas de codificación conocidos, tales como inter o intra codificación. Un generador de indicadores 120 del dispositivo 100 está configurado para fijar un indicador del fragmento asociado con el propio fragmento. Si el fragmento presente es el primer fragmento en la imagen, el fijador del indicador 120 fija el indicador del fragmento en un primer valor definido, tal como uno, mientras que para el(los) restante(s) fragmento(s) en la imagen el indicador del fragmento respectivo se fija a un segundo valor definido, tal como cero.

Un generador de la dirección 130 genera una dirección del fragmento respectivo para cada fragmento excepto para el primer fragmento en la imagen, es decir, para cada fragmento con un indicador del fragmento puesto a cero. La dirección del fragmento generada por el generador de la dirección 130 permite identificar una posición de una primera CU del fragmento dentro de la imagen y por ello la posición de comienzo del fragmento dentro de la imagen.

El dispositivo 100 también comprende un gestor de la representación 140 configurado para incluir las representaciones del fragmento codificado respectivas a partir del generador de la representación 110, de los indicadores del fragmento del generador de indicadores 120 y de la(s) dirección(es) del fragmento del generador de la dirección 130 de los fragmentos en una representación codificada de la imagen de la imagen. En una realización en particular, el indicador del fragmento se proporciona en la representación codificada antes de la(s) dirección(es) del fragmento. En tal caso, es posible el barrido ya que el indicador del fragmento decide si hay un campo o no de dirección del fragmento en la representación codificada del fragmento.

En una realización, el generador de la dirección 130 genera la dirección de fragmento basándose en la granularidad jerárquica determinada para la imagen por el dispositivo 100. En tal caso, se puede realizar un determinador de la longitud 150 opcional en el dispositivo 100 para utilizar la granularidad jerárquica con el fin de determinar la longitud de la dirección del fragmento y por ello el número de bits que la dirección del fragmento debe contener. El determinador de la longitud 150 utiliza también adicionalmente con preferencia la información del número total de LCUs en la imagen cuando determina la longitud de la dirección del fragmento, en la que este número total de LCUs se puede calcular como se describió previamente en este documento. En otra realización, el determinador de la longitud 150 se omite y el generador de la dirección 130 determina por sí mismo la longitud de la dirección del fragmento.

En el caso de la codificación H.264/MPEG-4 AVC, el determinador de la longitud 150 determina preferiblemente la longitud de la dirección del fragmento basándose en el número de macrobloques en la imagen, que se puede calcular basándose en la información del tamaño total de la imagen.

El generador de la dirección 130 utiliza entonces esta información de la longitud cuando genera la dirección del fragmento. En una realización en particular, el generador de la dirección 130 determina las coordenadas de la posición de una LCU dentro de la imagen y las coordenadas de la posición de una sub-LCU dentro de la LCU como se ha descrito previamente electrodo. La dirección del fragmento podría entonces comprender representaciones de las representaciones de estas posiciones de la LCU y de la sub-LCU o determinarse a partir de ella.

El dispositivo 100 se emplea ventajosamente con el fin de codificar múltiples fragmentos en paralelo con el fin de reducir el tiempo total de codificación de una imagen o de una secuencia de video.

El dispositivo 100 podría realizarse al menos parcialmente en software. En tal realización, el dispositivo 100 se realiza como un producto de programa de ordenador almacenado en la memoria y cargado y ejecutado en un ordenador, un procesador o un micro procesador de uso general o adaptado especialmente tal como una unidad central de proceso (CPU). El software incluye elementos de códigos de programas de ordenador o porciones de código de software que realiza la operación de al menos el generador de representación 110, el generador de indicadores 120, el generador de la dirección 130, el gestor de la representación 140 y el determinador opcional de la longitud 150. El programa se puede almacenar total o en parte, en o en uno o más medios de almacenamiento de datos o medios interpretables por ordenador volátiles adecuados, tales como RAM, o en uno o más medios de almacenamiento de datos o medios interpretables por ordenador no volátiles, tales como discos magnéticos, CD ROMs, discos DVD, disco duro, ROM o memoria flash. Los medios de almacenamiento de datos puede ser medios de almacenamiento de datos locales proporcionados a distancia, tales como un servidor de datos. El software puede por consiguiente ser cargado en la memoria operativa de un ordenador o de un sistema de tratamiento equivalente para la ejecución por un procesador. El ordenador/procesador no tiene que estar dedicado sólo a ejecutar las funciones descritas anteriormente sino que pueden también ejecutar otras tareas de software. Un ejemplo no limitativo del código de programa utilizado para definir el dispositivo 100 incluye código de datos múltiples en una única instrucción (SIMD).

Alternativamente el dispositivo 100 se puede realizar mediante hardware. Existen numerosas variantes de elementos de circuitería que se pueden utilizar y combinar para lograr las funciones de las unidades del dispositivo 100. Tales variantes quedan abarcadas por las realizaciones. Ejemplos particulares de realizar mediante hardware el dispositivo 100 es la realización en tecnología de circuitos integrados y hardware de procesador digital de la señal (DSP), incluyendo tanto la circuitería electrónica de uso general como de la aplicación específica.

La figura 11 es un diagrama esquemático de bloques de una realización de un descodificador o dispositivo 200 para descodificar una representación codificada de una imagen que consta de múltiples fragmentos. El dispositivo 200 comprende un recuperador de la representación 210 configurado para recuperar un indicador del fragmento asociado con un fragmento a ser descodificado a partir de la representación de la imagen codificada, normalmente a partir de un encabezamiento del fragmento en la representación de la imagen codificada. Un generador de la representación 220 se proporciona en el dispositivo 200 para generar una representación descodificada de los valores de píxel de los píxeles en el fragmento basándose en la representación de la imagen codificada. El generador de la representación 220 genera los valores de píxel de acuerdo con las técnicas conocidas, tales como los esquemas en modo intra o inter descodificación.

Un recuperador de la dirección 230 se hace operativo si el indicador del fragmento recuperado para un fragmento actual por el recuperador de la representación 210 tiene un segundo valor definido, tal como cero, indicando que el fragmento no es el primer fragmento en la imagen. El recuperador de la dirección 230 lee entonces y recupera una dirección del fragmento asociada con el fragmento a partir de la representación de la imagen codificada, tal como a partir de un encabezamiento del fragmento en la representación de la imagen codificada. Un asignador de valores 240 asigna entonces los valores del píxel obtenidos del generador de la representación 220 lo mismo que el fragmento se descodifica en píxeles en una porción del fragmento identificado basándose en la dirección del fragmento recuperada por el recuperador de la dirección 230.

Si el indicador del fragmento es recuperado por el de representaciones para un fragmento actual que tiene un primer valor definido, tal como uno, el asignador de valores 240 puede identificar directamente la porción del fragmento a la cual se le deben asignar los valores de píxel procedentes del generador de la representación 220. Esto generalmente es la primera parte de la imagen en el orden de codificación/descodificación, tal como la porción superior izquierda. Así, en tal caso no se necesita dirección del fragmento con el fin de identificar esta porción de la imagen.

En el caso de una realización HEVC y si la información de la granularidad jerárquica se le asigna a los datos de la imagen codificada, se puede proporcionar un recuperador opcional 250 de la información de la granularidad en el dispositivo 200 para recuperar la información de una granularidad jerárquica aplicable al fragmento presente para ser descodificado. El recuperador 250 de la información de la granularidad podría recuperar la información de la granularidad a partir de la representación de la imagen codificada o del campo global del encabezamiento, tal como el campo PTS o SPS, asociado con la representación codificada de la imagen. La información de la granularidad recuperada por el recuperador de información 250 de la granularidad la emplea un determinador opcional 260 de la longitud para determinar la longitud de la dirección de fragmento y determinar por ello el número de bits que el recuperador de la dirección 230 debe leer con el fin de recuperar la dirección del fragmento. Alternativamente, este



determinador de la longitud 260 se puede omitir y el recuperador de la dirección 230 determina por sí mismo la longitud de la dirección basándose en la información de la granularidad.

Un recuperador opcional 270 de la información de la unidad de codificación se ejecuta ventajosamente en el dispositivo 200 con objeto de recuperar la información del número total de LCUs en la imagen a partir de la representación codificada de la imagen, tal como de un campo global de encabezamientos, campo PPS o SPS. Esa información podría ser la mencionada anteriormente *log2\_min\_coding\_block\_size\_minus3* y *log2\_diff\_max\_min\_coding\_block\_size*, que permite que el recuperador 270 de la información de la unidad de codificación calcule el número de LCUs en la información de la imagen dada del tamaño total de la imagen, lo que preferiblemente está también disponible a partir de la representación de la imagen codificada o a partir de un campo global de encabezamiento, campo PPS o SPS.

El determinador 260 de la longitud determina entonces ventajosamente la longitud de la dirección del fragmento basándose en la información del recuperador 250 de la información de la granularidad y el número total de LCUs como el determinado por el recuperador 270 de información de la unidad de codificación.

En una realización, el recuperador 230 de la dirección se configura para determinar una primera representación o dirección de la LCU de las coordenadas de una posición de una LCU dentro de la imagen basándose en la dirección del fragmento si el fragmento actual no es el primer fragmento en la imagen. El recuperador 230 de la dirección determina también preferiblemente una segunda representación o dirección de la sub-LCU de las coordenadas de una posición de una primera unidad de codificación del fragmento y por ello el comienzo del fragmento dentro de la LCU. La dirección de la LCU y de la sub-LCU las utiliza entonces el recuperador 230 de la dirección para identificar la porción de la imagen que pertenece al fragmento actual basándose en la dirección de la LCU y en la dirección de la sub-LCU como se describió anteriormente.

Por ejemplo, el recuperador 230 de la dirección puede determinar el parámetro *LCUAddress=slice\_address>>SliceGranularity* basándose en la dirección del fragmento (*slice-address*) y basándose en la información de la granularidad jerárquica (*SliceGranularity*). La dirección de la sub-LCU se determina preferiblemente como *GranularityAddress=slice\_address - (LCUAddress<<SliceGranularity)* basándose en la dirección del fragmento (*slice\_address*), en la información de la granularidad jerárquica (*SliceGranularity*) y en la dirección de la LCU.

El generador de la representación 220 determina preferiblemente el tamaño de la primera CU en el fragmento de acuerdo con el número de píxeles basándose al menos parcialmente en la dirección del fragmento. La dirección del fragmento define entonces el tamaño de la mayor CU posible que pueda ocupar la posición definida por la dirección del fragmento en la imagen. En una realización, el tamaño de la primera CU viene determinado por el generador 220 de representaciones basándose únicamente en la dirección del fragmento. Esto es posible cuando el tamaño de la primera CU es igual al tamaño de la SCU y no es posible dividir más la CU. En otras realizaciones, el generador 220 de representaciones utiliza adicionalmente uno o más indicadores de la unidad de codificación dividida incluidos en la representación de la imagen codificada junto con la dirección del fragmento para determinar el tamaño de la primera CU. Si un único indicador de la unidad de codificación dividida es igual a cero o a algún otro valor definido, el tamaño de la primera CU es igual al tamaño de la mayor CU que pueda ocupar la posición dentro de la imagen definida por la dirección del fragmento. Si el indicador de la unidad de codificación dividida es por el contrario igual a uno o a algún otro valor definido, el tamaño de la primera CU es menor que, preferiblemente la cuarta parte de la mayor CU posible que pueda ocupar la posición dentro de la imagen definida por la dirección del fragmento.

Por ejemplo, si el tamaño de la mayor CU posible en las coordenadas de comienzo es 32x32 píxeles (con un tamaño de la LCU de 64x64 píxeles y el tamaño de una SCU es de 8x8 píxeles) el(los) indicador(es) de la unidad de codificación dividida sería(n):

*split\_coding\_unit\_flag=0*  
para un tamaño de 32x32 píxeles de la primera CU

*split\_coding\_unit\_flag=1*  
*split\_coding\_unit\_flag=0*  
para un tamaño de 16x16 píxeles de la primera CU

*split\_coding\_unit\_flag=1*  
*split\_coding\_unit\_flag=1*  
para un tamaño de 8x8 píxeles de la primera CU

El dispositivo 200 se podría realizar al menos parcialmente mediante software. En tal realización, el dispositivo 200 se realiza como un producto de programa de ordenador almacenado en una memoria y cargado y ejecutado en un ordenador, procesador o microprocesador, de uso general o adaptado especialmente, tal como una unidad central de proceso (CPU). El software incluye elementos de código de programa de ordenador o porciones de código de software que efectúan la operación de por lo menos el recuperador 210 de la representación, el generador 220 de representaciones, el recuperador 230 de la dirección, el asignador 240 de valores, el recuperador 250 de la información de la granularidad, el determinador opcional 260 de la longitud y el recuperador opcional 270 de la

información de la unidad de codificación. El programa se puede almacenar completo o parcialmente, en uno o en más medios volátiles apropiados interpretables por ordenador o de almacenamiento de datos, tales como ram, o en uno o en más medios no volátiles interpretables por ordenador o medios de almacenamiento de datos, tales como discos magnéticos, CD-ROMs, discos DVD, discos duros, en rom o en memoria flash. Los medios de almacenamiento de datos puede ser medios locales de almacenamiento de datos o proporcionados a distancia, tales como en un servidor de datos. El software se puede así cargar en la memoria de trabajo de un ordenador o en un sistema equivalente de tratamiento para su ejecución por medio de un procesador. El ordenador/procesador no tiene que estar dedicado sólo a ejecutar las funciones descritas anteriormente sino que puede también ejecutar otras tareas de software. Un ejemplo no limitativo de un código de programa utilizado para definir el dispositivo 200 incluye códigos de múltiples datos de instrucción única (SIMD).

Alternativamente, el dispositivo 200 se puede realizar en hardware. Existen numerosas variantes de elementos de circuitería que se pueden utilizar y combinar para conseguir las funciones de las unidades del dispositivo 200. Tales variantes quedan abarcadas en la realizaciones. Ejemplos particulares de realización mediante hardware del dispositivo 200 es la realización en hardware en un procesador digital de señal (DSP) y tecnología de circuitos integrados, incluyendo tanto la circuitería electrónica de uso general como la circuitería de aplicaciones específicas.

La figura 12 es un diagrama esquemático de bloques de un terminal de medios 300 que contiene un dispositivo 200 para descodificar una representación codificada de una imagen. El terminal de medios 300 puede ser cualquier dispositivo que disponga de funciones de descodificación de medios que opera sobre una secuencia codificada de bits, tal como un secuencia de video de tramas codificadas de video para descodificar por lo tanto las tramas de video y hacer disponibles los datos de video. Ejemplos no limitativos de tales dispositivos, incluyen teléfonos móviles y otros medios de reproducción portátiles, ordenadores, descodificadores, consolas de videojuegos, etcétera. El terminal de medios 300 consta de una memoria 320 configurada para la representación codificada de una imagen, tal como tramas de video codificadas. Las representaciones codificadas pueden haber sido generadas por el propio terminal de medios 300. En tal caso, el terminal de medios 300 comprende preferiblemente un motor de medios o registrador junto con un codificador conectado, tal como el dispositivo para codificar una imagen de la figura 10. Alternativamente, las representaciones codificadas se generan por medio de algún otro dispositivo o se transmiten inalámbricamente o mediante cable al terminal de medios 300. El terminal de medios 300 comprende entonces un transceptor 310 (transmisor y receptor) o un puerto de entrada y salida para lograr la transferencia de datos.

La representación codificada se trae desde la memoria 320 al dispositivo 200 para la descodificación, tal como el dispositivo ilustrado en la figura 11. El dispositivo 200 descodifica entonces la representación codificada en una imagen descodificada o como tramas de video descodificadas. Los datos descodificados se proporcionan a un reproductor de medios 330 que está configurado para interpretar los datos de la imagen descodificada o de las tramas de video en datos que se presentan en una pantalla 340 de o conectada al terminal de medios 300.

En la figura 12, el terminal de medios 300 se ha ilustrado como que comprende tanto el dispositivo 200 para la descodificación como el reproductor de medios 330. Esto debe, sin embargo, ser visto simplemente como un ejemplo ilustrativo pero no limitativo de una realización de la ejecución para el terminal de medios 300. También son posibles realizaciones distribuidas en las que el dispositivo 200 y el reproductor de medios 330 se proporcionan en dos dispositivos físicamente separados y dentro del alcance del terminal de medios 300 como se ha usado en ese documento. La pantalla 340 se podría también proporcionar como un dispositivo separado conectado al terminal de medios 300, en el que tiene lugar el tratamiento real de los datos.

Las realizaciones descritas anteriormente tienen que entenderse como unos pocos ejemplos ilustrativos de la presente invención. Los expertos en la técnica comprenderán que se pueden hacer diversas modificaciones, combinaciones y cambios a las realizaciones sin apartarse del alcance de la presente invención. En particular, diferentes soluciones parciales en las diferentes realizaciones se pueden combinar en otras configuraciones que sean técnicamente posibles. El alcance de la presente invención está, sin embargo, definido por las reivindicaciones adjuntas.

## REIVINDICACIONES

1. Un método de descodificar una imagen (1) que comprende  $N \geq 2$  fragmentos (2-5), que comprende:
- 5 generar (S1) para cada fragmento (2-5) de dichos  $N$  fragmentos (2-5), una representación codificada del fragmento basándose en los valores de píxel de los píxeles en dicho fragmento (2-5);
- fijar (S2) un indicador del fragmento asociado al primer fragmento (2) de dicha imagen (1) a un primer valor definido para indicar que dicho primer fragmento (2) tiene su comienzo del fragmento en una posición definida en dicha imagen (1);
- 10 fijar (S3), para cada fragmento (3-5) de los  $N - 1$  fragmentos restantes (3-5) de dichos  $N$  fragmentos (2-5), un indicador del fragmento asociado a dicho fragmento (3-5) a un segundo valor definido;
- generar (S4), para cada fragmento (3-5) de dichos  $N - 1$  fragmentos restantes (3-5), una dirección del fragmento que permite la identificación de una posición de una primera unidad de codificación (10B, 20B, 30B, 40B) de dicho fragmento (3-5) dentro de dicha imagen (1); y
- 15 generar (S5) una representación codificada de la imagen (50) de dicha imagen (1) comprendiendo dichas  $N$  representaciones codificadas del fragmento, dichas  $N - 1$  direcciones del fragmento y dichos  $N$  indicadores del fragmento, caracterizado por:
- determinar (S11) una longitud de dichas  $N - 1$  direcciones del fragmento con respecto al número de bits como uno de  $\text{ceil}(\log_2(P))$  y  $\text{ceil}(\log_2(P - 1))$ , en donde  $\text{ceil}()$  indica la función techo definida como  $\text{ceil}(x) = \lceil x \rceil$  y extrae el menor entero no menor de  $x$  y  $P$  indica un número total de posibles direcciones del fragmento en dicha imagen (1), en el
- 20 que la generación (S4) de dicha dirección del fragmento comprende generar (S4), para cada fragmento (3-5) de dichos  $N - 1$  fragmentos restantes (3-5), dichas direcciones del fragmento codificando una palabra código que tiene dicha longitud.
2. El método de acuerdo con la reivindicación 1, en el que dicha imagen (1) consta de múltiples unidades de codificación mayores (10), que tienen un tamaño seleccionado con respecto al número de píxeles y que puede ser
- 25 dividida jerárquicamente en múltiples unidades de codificación (20, 30, 40) que tienen tamaño respectivos que son menores que dicho tamaño seleccionado, comprendiendo además dicho método determinar (S11) una longitud de dichas  $N - 1$  direcciones del fragmento con respecto al número de bits basándose en una granularidad jerárquica seleccionada para dicha imagen (1), definiendo dicha granularidad jerárquica un nivel jerárquico para el alineamiento del borde del fragmento dentro de dicha imagen (1), en el que dicho nivel jerárquico define un tamaño de una unidad
- 30 de codificación direccional lo menor posible (30) en la cual se puede alinear el comienzo de un fragmento (3) en dicha imagen (1).
3. El método de acuerdo con la reivindicación 2, en el la determinación (S11) de dicha longitud de dichas  $N - 1$  direcciones del fragmento comprende determinar (S11) dicha longitud de dichas  $N - 1$  direcciones del fragmentos basándose en dicha granularidad jerárquica y basándose en un número total de las mayores posibles unidades de
- 35 codificación (10) en dicha imagen (1).
4. Un método de descodificar la representación codificada (50) de una imagen (1) que comprende  $N \geq 2$  fragmentos (2-5), comprendiendo dicho método:
- recuperar (S20), de dicha representación codificada de la imagen (50), un indicador del fragmento asociado a un
- 40 fragmento (2-5) de dichos  $N$  fragmentos (2-5);
- generar (S21) una representación descodificada de los valores de píxel de los píxeles en dicho fragmento (2-5) basándose en la representación codificada de un fragmento asociada a dicho fragmento (2-5) y recuperada de dicha representación codificada de la imagen (50);
- 45 asignar (S23) dichos valores de píxel a los píxeles en una primera porción definida de dicha imagen (1) si un indicador del fragmento asociado a dicho fragmento (2) y recuperado de dicha representación codificada de la imagen (50) tiene un primer valor definido que indica que es el primer fragmento (2) en la imagen (1);
- recuperar (S24), si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene un segundo valor definido, una dirección del fragmento de dicha representación codificada de la imagen (50), y
- 50 asignar (S25), si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, dichos valores de píxel a los píxeles en una porción de dicha imagen (1) identificada basándose en dicha dirección del fragmento, caracterizado por
- determinar (S32), si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, una longitud de dicha dirección del fragmento con respecto al número de bits como uno de  $\text{ceil}(\log_2(P))$  y

$\text{ceil}(\log_2(P - 1))$ , en el que  $\text{ceil}()$  indica la función techo definida como  $\text{ceil}(x) = \lceil x \rceil$  y extrae el menor entero no menor de  $x$  y  $P$  indica un número total de posibles direcciones del fragmento en dicha imagen (1), en el que la recuperación (S24) de dicha imagen del fragmento comprende recuperar (S24) si dicho indicador del fragmento tiene dicho segundo valor definido, dicha dirección del fragmento como una palabra código codificada que tiene dicha longitud de dicha representación codificada de la imagen (50).

5 El método de acuerdo con la reivindicación 4, en el que dicha imagen (1) consta de múltiples unidades de codificación mayores (10), que tienen un tamaño seleccionado con respecto al número de píxeles y que se pueden dividir jerárquicamente en múltiples unidades de codificación (20, 30, 40) que tienen tamaños respectivos que son menores que dicho tamaño seleccionado, comprendiendo además dicho método:

10 recuperar (S30), si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, información de una granularidad jerárquica asociada a dicha presentación codificada de la imagen (50), definiendo dicha granularidad jerárquica un nivel jerárquico para el alineamiento del borde del fragmento dentro de dicha imagen (1), definiendo dicha nivel jerárquico un tamaño de una unidad de codificación direccionable lo más pequeña posible (30) en la cual se puede alinear un comienzo de un fragmento (3) en dicha imagen (1), y

15 determinar (S32) la longitud de una dirección de dicho fragmento con respecto al número de bits basándose en dicha información de dicha granularidad jerárquica.

20 6. El método de acuerdo con la reivindicación 5, que comprende además recuperar (S31) la información de un número total de las mayores unidades de codificación (10) en dicha imagen (1) de dicha representación codificada de la imagen (50), en el que la determinación (S32) de dicha longitud de dicha dirección del fragmento comprende determinar (S32) dicha longitud de dicha dirección del fragmento basándose en dicha información de dicha granularidad jerárquica y basándose en dicho número total de las mayores unidades posibles de codificación (10).

7. El método de acuerdo con cualquiera de las reivindicaciones 4 a 6, que comprende además:

25 determinar (S40), si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, una primera representación de las coordenadas de una posición de la unidad mayor de codificación (10) dentro de dicha imagen (1) basándose en dicha dirección del fragmento;

determinar (S41), si dicho fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, una segunda representación de las coordenadas de una posición de una primera unidad de codificación (30) de dicho fragmento (3) dentro de dicha mayor unidad posible de codificación (10), e

30 identificar (S42) dicha porción de dicha imagen (1) basándose en dicha primera representación y en dicha segunda representación.

8. El método de acuerdo con la reivindicación 7, en el que dicha imagen (1) comprende múltiples unidades mayores de codificación (10), que tienen un tamaño seleccionado con respecto al número de píxeles y que se puede dividir jerárquicamente en múltiples unidades de codificación (20, 30, 40) teniendo tamaño respectivos que son menores que dicho tamaño seleccionado, comprendiendo además dicho método:

35 recuperar (S30), si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, la información de una granularidad jerárquica asociada a dicha presentación codificada de la imagen (50), definiendo dicha granularidad jerárquica un nivel jerárquico para el alineamiento del borde del fragmento dentro de dicha imagen (1), definiendo dicho nivel jerárquico un tamaño de una unidad de codificación direccionable lo menor posible (30) en la cual se puede alinear un comienzo de un fragmento (3) en dicha imagen (1); y

40 determinar (S32) una longitud de dicha dirección del fragmento con respecto al número de bits basándose en dicha información de dicha granularidad jerárquica, en el que la determinación (S40) de dicha primera representación comprende determinar (S40), si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, basándose en dicha primera representación en dicha dirección del fragmento y en dicha información de dicha granularidad jerárquica.

45 9. El método de acuerdo con la reivindicación 7 o con la reivindicación 8, en el que dicha imagen (1) comprende las múltiples mayores unidades de codificación (10), que tienen un tamaño seleccionado con respecto al número de píxeles y que se pueden dividir jerárquicamente en múltiples unidades de codificación (20, 30, 40) que tienen respectivos tamaños que son menores que dicho tamaño seleccionado, comprendiendo dicho método además:

50 recuperar (S30), si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, la información de una granularidad jerárquica asociada a dicha presentación codificada de la imagen (50), definiendo dicha granularidad jerárquica un nivel jerárquico para el alineamiento del borde del fragmento dentro de dicha imagen (1), definiendo dicho nivel jerárquico un tamaño de una unidad de codificación direccionable lo menor posible (30) en la cual se puede alinear un comienzo de un fragmento (3) en dicha imagen (1); y

- determinar (S32) una longitud de dicha dirección del fragmento con respecto al número de bits basándose en dicha información de dicha granularidad jerárquica, en el que la determinación (S41) dicha segunda representación comprende determinar (S41), si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, dicha segunda representación basada en dicha dirección del fragmento, dicha información de dicha granularidad jerárquica y dicha primera representación.
- 5
10. Un dispositivo (100) para codificar una imagen (1) que comprende  $N \geq 2$  fragmentos (2-5), comprendiendo dicho dispositivo (100):
- un generador de la representación (110) configurado para generar, para cada fragmento (2-5) de dichos  $N$  fragmentos (2-5), una representación codificada del fragmento basándose en los valores de píxel de los píxeles en dicho fragmento (2-5);
- 10 un fijador del indicador (120) configurado para fijar un indicador del fragmento asociado a dicho fragmento (2) de dicha imagen (1) a un primer valor definido y fijar, para cada fragmento (3-5) de los  $N - 1$  fragmentos restantes (3-5) de dichos  $N$  fragmentos (2-5), un indicador del fragmento asociado a dicho fragmento (3-5) a un segundo valor definido;
- 15 un generador de la dirección (130) configurado para generar, para cada fragmento (3-5) de dichos  $N - 1$  restantes fragmentos (3-5), una dirección del fragmento que permite identificar una posición de una primera unidad de codificación (10B, 20B, 30B, 40B) de dicho fragmento (3-5) dentro de dicha imagen (1); y
- un gestor de la representación (140) configurado para generar una representación codificada de la imagen (50) de dicha imagen (1) comprendiendo dichas  $N$  representaciones codificadas del fragmento, dichas  $N - 1$  direcciones del fragmento y dichos  $N$  indicadores del fragmento, caracterizado por:
- 20 un determinador de la longitud (150) configurado para determinar una longitud de dichas  $N - 1$  direcciones del fragmento con respecto al número de bits como uno de  $\text{ceil}(\log_2(P))$  y  $\text{ceil}(\log_2(P - 1))$ , en el que  $\text{ceil}()$  indica la función techo definida como  $\text{ceil}(x) = \lceil x \rceil$  y extrae el menor entero no menor que  $x$  y  $P$  indica un número total de las posibles direcciones del fragmento en dicha imagen (1), en el que dicho generador de la dirección (130) está configurado para generar, para cada fragmento (3-5) de dichos  $N - 1$  fragmentos restantes (3-5), dicha dirección del fragmento para codificar una palabra código que tiene dicha longitud.
- 25
11. El dispositivo de acuerdo con la reivindicación 10, en el que dicha imagen (1) comprende las múltiples mayores unidades de codificación (10), que tienen un tamaño seleccionado con respecto al número de píxeles y que se pueden dividir jerárquicamente en múltiples unidades de codificación (20, 30, 40) que tienen tamaños respectivos que son menores que dicho tamaño seleccionado, comprendiendo además dicho dispositivo (100) un determinador de la longitud (150) configurado para determinar una longitud de dichas  $N - 1$  direcciones del fragmento con respecto al número de bits basándose en una granularidad jerárquica seleccionada para dicha imagen (1), definiendo dicha granularidad jerárquica un nivel jerárquico para el alineamiento del borde del fragmento dentro de dicha imagen (1), en el que dicho nivel jerárquico define un tamaño de una unidad de codificación direccionable lo menor posible (30) en la cual se puede alinear un comienzo de un fragmento (3) en dicha imagen (1).
- 30
12. El dispositivo de acuerdo con la reivindicación 11 en el que dicho determinador de la longitud (150) está configurado para determinar dicha longitud de dichas  $N - 1$  direcciones del fragmento basándose en dicha granularidad jerárquica y basándose en un número total de las mayores unidades de codificación (10) en dicha imagen (1).
- 35
13. Un dispositivo (200) para descodificar la representación codificada (50) de una imagen (1) que comprende  $N \geq 2$  fragmentos (2-5), comprendiendo dicho dispositivo (200):
- 40 un recuperador de la representación (210) configurado para recuperar, de dicha representación codificada de la imagen (50), un indicador del fragmento asociado con un fragmento (2-5) de dichos  $N$  fragmentos (2-5);
- un generador de la representación (220) configurado para generar una representación descodificada de los valores de píxel de los píxeles en dicho fragmento (2-5) basándose en una representación codificada del fragmento asociada a dicho fragmento (2-5) y recuperada de dicha representación codificada de la imagen (50);
- 45 un recuperador de la dirección (230) configurado para recuperar, si un indicador del fragmento asociado a dicho fragmento (3-5) y recuperado de dicha representación codificada de la imagen (50) tiene un segundo valor definido, una dirección del fragmento asociada a dicho fragmento (3-5) de dicha representación codificada de la imagen (50);
- 50 y
- un asignador de valores (240) configurado para asignar dichos valores de píxel a los píxeles en una primera porción definida de dicha imagen (1) si dicho indicador del fragmento asociado a dicho fragmento (2) tiene un primer valor definido que indica el primer fragmento (2) en la imagen (1) y para asignar, si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, dichos valores de píxel a los píxeles en una porción de dicha imagen (1) identificada basándose en dicha dirección del fragmento, caracterizado por:
- 55

- un determinador de la longitud (260) configurado para determinar, si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, una longitud de dichas  $N - 1$  direcciones del fragmento con respecto al número de bits como uno de  $\text{ceil}(\log_2(P))$  y  $\text{ceil}(\log_2(P - 1))$  en el que  $\text{ceil}(\ )$  indica la función techo definida como  $\text{ceil}(x) = \lceil x \rceil$  y extrae el menor entero no menor que  $X$  y  $P$  indica un número total de posibles direcciones del fragmento en dicha imagen (1), en el que dicho recuperador de la dirección (230) está configurado para recuperar, si dicho indicador del fragmento tiene dicho segundo valor, dicha dirección del fragmento como una palabra de código codificada que tiene dicha longitud de dicha representación codificada de la imagen (50).
- 5
14. El dispositivo de acuerdo con la reivindicación 13, en el que dicha imagen (1) comprende las múltiples mayores unidades de codificación (10), que tienen un tamaño seleccionado con respecto al número de píxeles y que se pueden dividir jerárquicamente en múltiples unidades de codificación (20, 30, 40) que tienen tamaño respectivos que son menores que dicho tamaño seleccionado, comprendiendo además dicho dispositivo (200):
- 10
- un recuperador de información de la granularidad (250) configurado para recuperar, si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, la información de una granularidad jerárquica asociada a dicha presentación codificada de la imagen (50), definiendo dicha granularidad jerárquica un nivel jerárquico para el alineamiento del borde del fragmento dentro de dicha imagen (1), definiendo dicho nivel jerárquico un tamaño de una unidad de codificación direccionable lo menor posible (30) en la cual se puede alinear un comienzo de un fragmento (3) en dicha imagen (1); y
- 15
- un determinador de la longitud (260) configurado para determinar una longitud de dicha dirección del fragmento con respecto al número de bits basándose en dicha información de dicha granularidad jerárquica.
- 20
15. El dispositivo de acuerdo con la reivindicación 14, que comprende además un recuperador de información de la unidad de codificación (270) configurado para recuperar la información de un número total de las mayores unidades de codificación (10) en dicha imagen (1) de dicha representación codificada de la imagen (50), en el que dicho determinador de la longitud (260) está configurado para determinar dicha longitud de dicha dirección del fragmento basándose en dicha información de dicha granularidad jerárquica y basándose en dicho número total de las mayores unidades de codificación (10).
- 25
16. El dispositivo de acuerdo con cualquiera de las reivindicaciones 13 a 15, en el que dicho recuperador de la dirección (230) está configurado para i) determinar, si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, una primera representación de las coordenadas de una posición de la mayor unidad de codificación (10) dentro de dicha imagen (1) basándose en dicha dirección del fragmento, ii) determinar, si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, una segunda representación de las coordenadas de una posición de una primera unidad de codificación (30) de dicho fragmento (3) dentro de dicha mayor unidad de codificación (10) y iii) identificar, dicha porción de dicha imagen (1) basándose en dicha primera representación y en dicha segunda representación.
- 30
17. El dispositivo de acuerdo con la reivindicación 16, en el que dicha imagen (1) comprende las múltiples mayores unidades de codificación (10), que tienen un valor seleccionado con respecto al número de píxeles y que pueden dividirse jerárquicamente en múltiples unidades de codificación (20, 30, 40) que tienen respectivos tamaños que son menores que dicho valor seleccionado, comprendiendo además dicho dispositivo (200):
- 35
- un recuperador de la información de la granularidad (250) configurado para recuperar, si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, la información de una granularidad jerárquica asociada a dicha presentación codificada de la imagen (50), definiendo dicha granularidad jerárquica un nivel jerárquico para el alineamiento del borde del fragmento dentro de dicha imagen (1), definiendo dicho nivel jerárquico un tamaño de una unidad de codificación direccionable lo menor posible (30) en la cual se puede alinear un comienzo de un fragmento (3) en dicha imagen (1); y
- 40
- un determinador de la longitud (260) configurado para determinar una longitud de dicha dirección del fragmento con respecto al número de bits basándose en dicha información de dicha granularidad jerárquica, en el que dicho recuperador de la longitud (230) está configurado para determinar, si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, dicha primera representación basada en dicha dirección del fragmento y dicha información de dicha granularidad jerárquica.
- 45
18. El dispositivo al acuerdo con la reivindicación 16 o con la reivindicación 17, en el que dicha imagen (1) comprende las múltiples mayores unidades de codificación (10), que tienen un tamaño seleccionado con respecto al número de píxeles y que se pueden dividir jerárquicamente en múltiples unidades de codificación (20, 30, 40) teniendo respectivos tamaños que son menores que dicho valor seleccionado, comprendiendo además dicho dispositivo (200):
- 50
- un recuperador de la información de la granularidad (250) configurado para recuperar, si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, la información de una granularidad jerárquica asociada a dicha presentación codificada de la imagen (50), definiendo dicha granularidad jerárquica un nivel jerárquico para el alineamiento del borde del fragmento dentro de dicha imagen (1), definiendo dicho nivel
- 55

jerárquico un tamaño de una unidad de codificación direccionable lo menor posible (30) en la cual se puede alinear un comienzo de un fragmento (3) en dicha imagen(1), y

5 un determinador de la longitud (260) configurado para determinar una longitud de dicha dirección del fragmento con respecto al número de bits basándose en dicha información de dicha granularidad jerárquica, en el que dicho recuperador de la longitud (230) está configurado para determinar, si dicho indicador del fragmento asociado a dicho fragmento (3-5) tiene dicho segundo valor definido, dicha segunda representación basada en dicha longitud del fragmento, dicha información de dicha granularidad jerárquica y dicha primera representación.

19. Un terminal de medios (300) que comprende:

10 una memoria (320) configurada para almacenar una representación codificada (50) de una imagen (1) que comprende múltiples fragmentos (2-5) y que se compone de las múltiples mayores unidades de codificación (10) que tienen un tamaño seleccionado con respecto al número de píxeles, y

un dispositivo (200) para descodificar dicha representación codificada (50) de acuerdo con cualquiera de las reivindicaciones 13 a 18.



Fig. 1

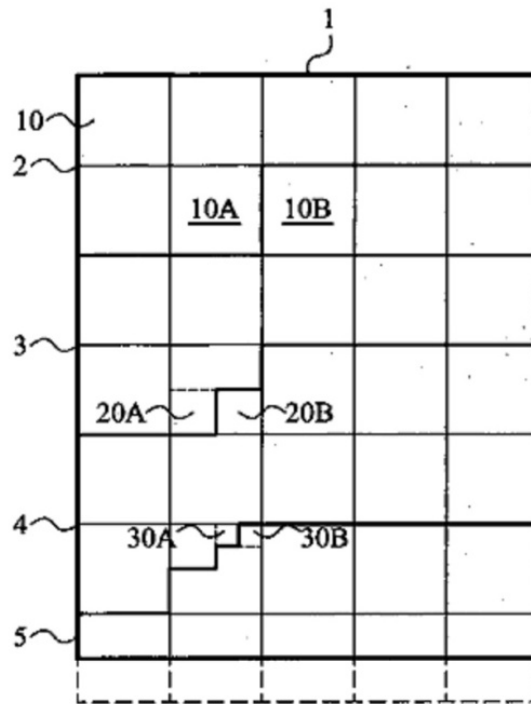


Fig. 2

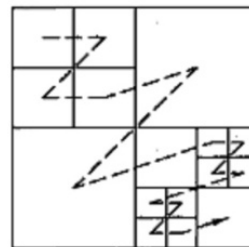


Fig. 4



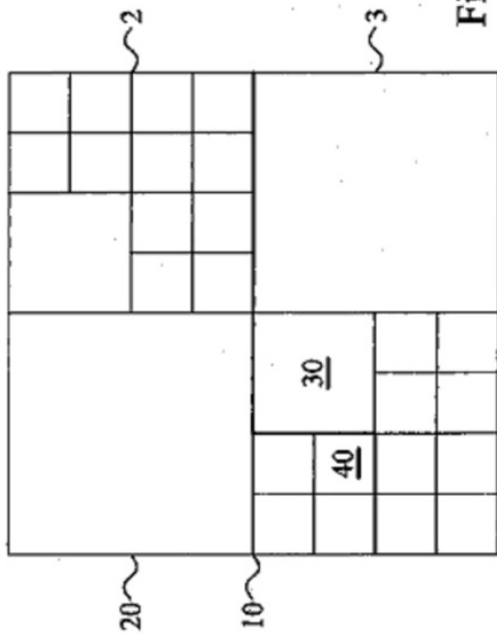


Fig. 3A

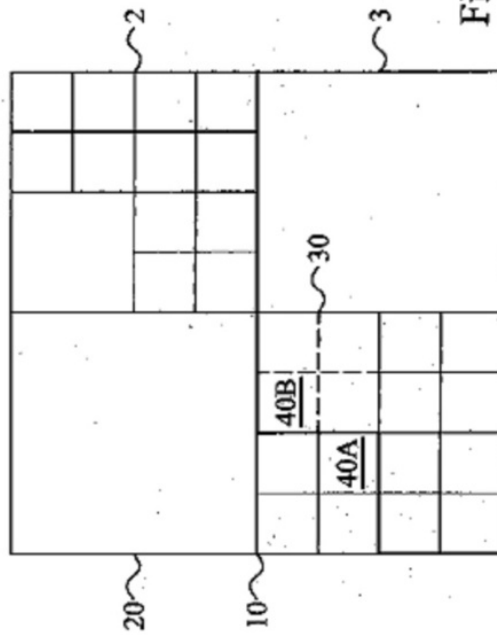


Fig. 3B

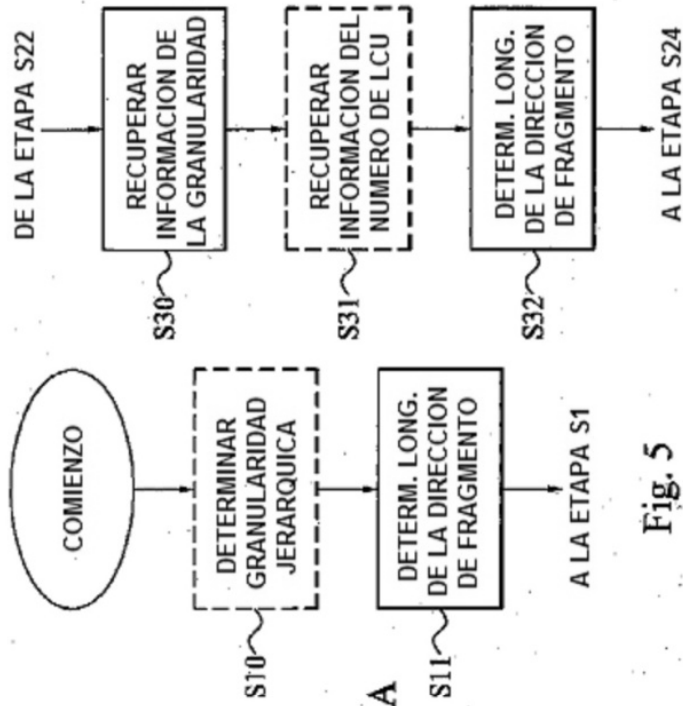


Fig. 5

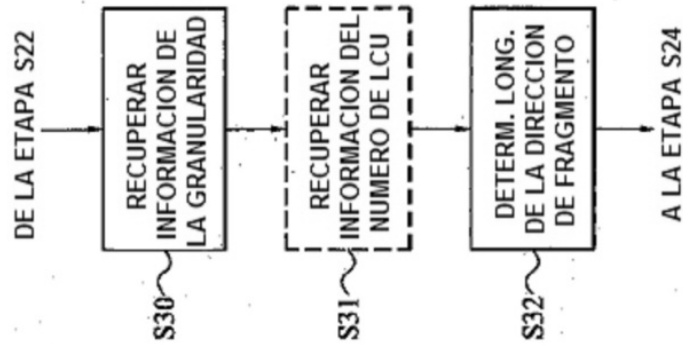
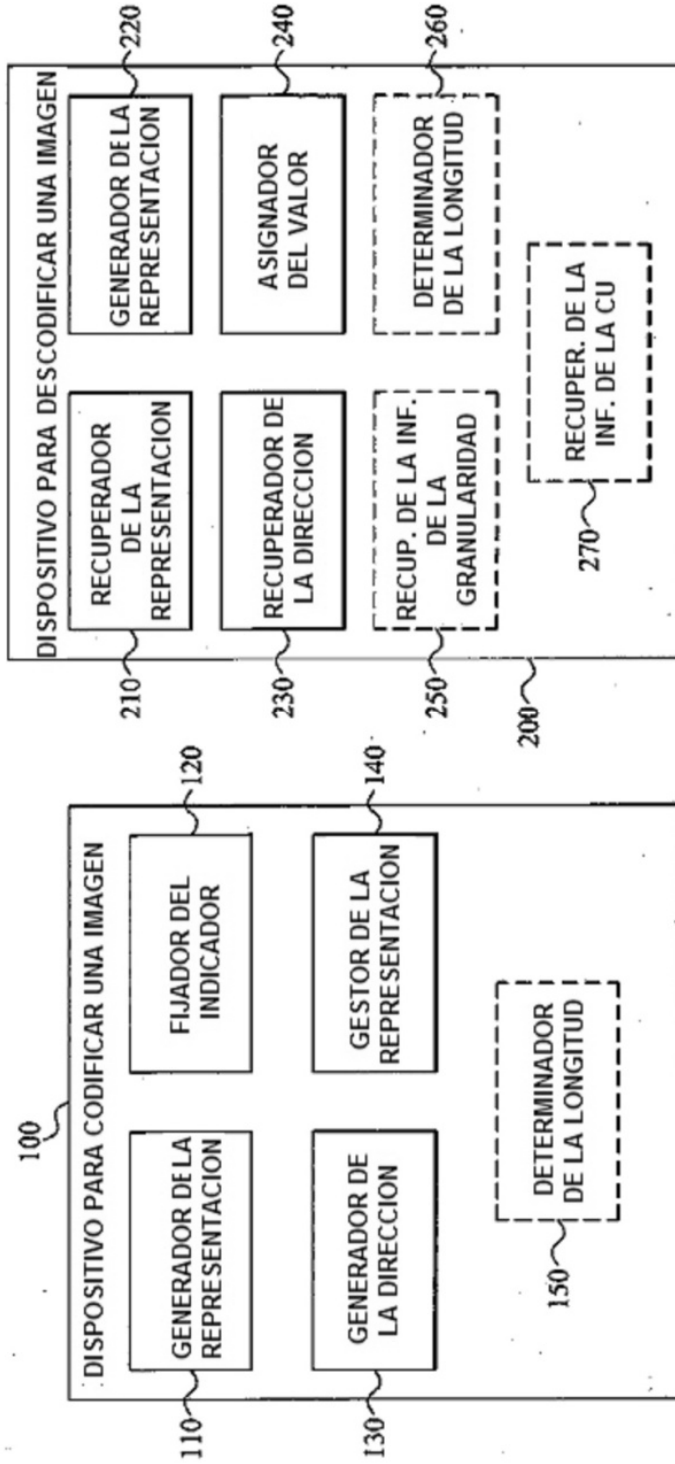


Fig. 8



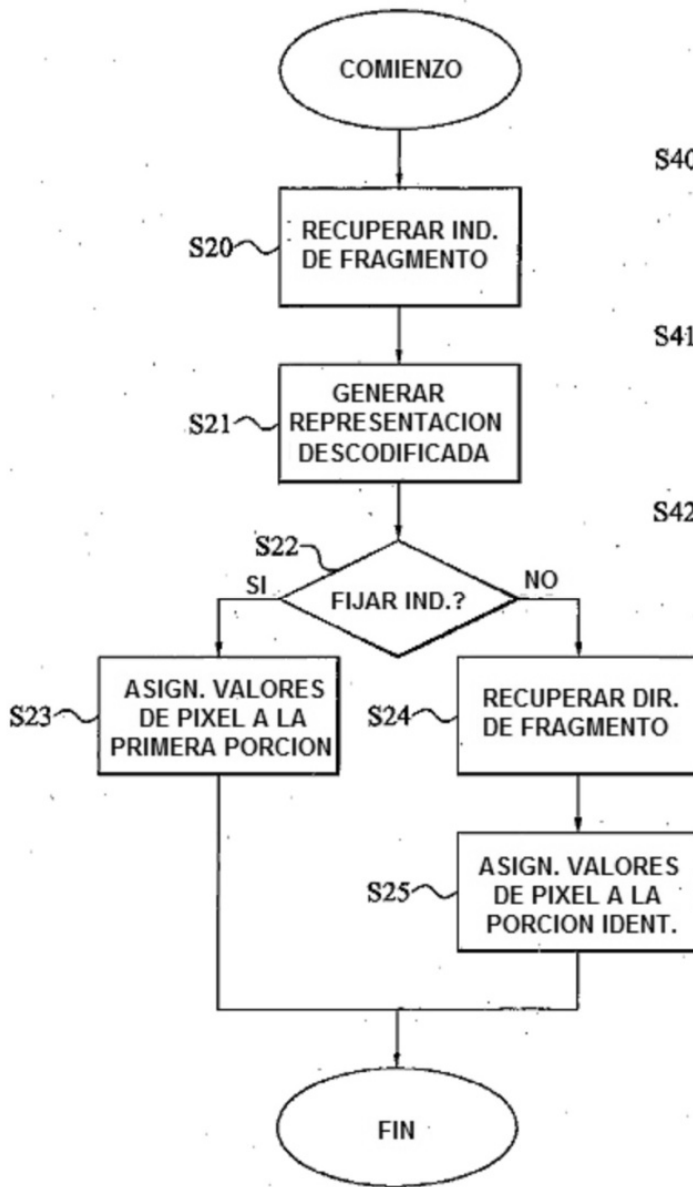


Fig. 7

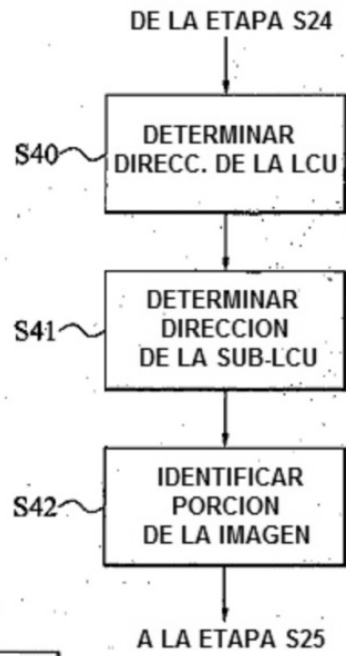


Fig. 9

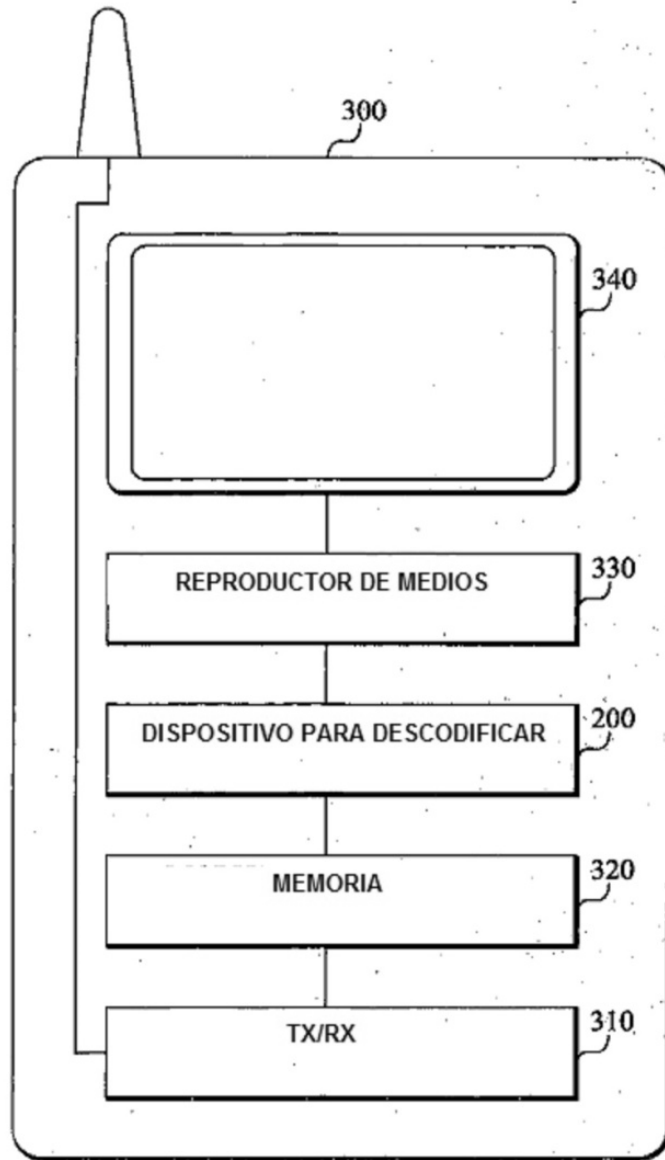


Fig. 12