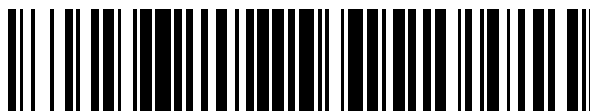


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 648 970**

51 Int. Cl.:

H04N 19/597	(2014.01)
H04N 19/70	(2014.01)
H04N 19/30	(2014.01)
H04N 19/463	(2014.01)
H04N 19/105	(2014.01)
H04N 19/157	(2014.01)
H04N 19/187	(2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **11.06.2013 PCT/SE2013/050670**
- 87 Fecha y número de publicación internacional: **26.06.2014 WO14098703**
- 96 Fecha de presentación y número de la solicitud europea: **11.06.2013 E 13731991 (9)**
- 97 Fecha y número de publicación de la concesión europea: **23.08.2017 EP 2936817**

54 Título: **Codificación y decodificación de flujo de video multicapa**

30 Prioridad:

21.12.2012 US 201261740562 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
09.01.2018

73 Titular/es:

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

72 Inventor/es:

RUSERT, THOMAS

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 648 970 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 flujo de video multicapa

Campo técnico

5 Las realizaciones se relacionan de manera general al procesamiento de un flujo de video multicapa, y en particular a la decodificación y codificación de dicho flujo de video multicapa.

Antecedentes

10 La HECV (Codificación de Vídeo de Alta Eficiencia) es el estándar de codificación de video de la siguiente generación que actualmente está en proceso de estandarización. La HECV mejorará de manera sustancial la eficiencia de codificación comparada con el H-264/Codificación de Video Avanzada (AVC) de última generación. El foco inicial del desarrollo de la HECV estaba en el video mono, esto es una vista de cámara, a una calidad y tasa de bit fija, esto es no escalable. Ahora las extensiones multicapa del estándar HECV están bajo preparación, por ejemplo una extensión escalable, una extensión multivista, y una extensión 3D. Estas extensiones requieren soporte multicapa. Un flujo de bits HEVC sin extensiones se puede considerar un flujo de bits de capa única, esto es representa el video en una única representación, por ejemplo una única vista de video, una única resolución y una
15 única calidad. En las extensiones multicapa, se incluye normalmente un flujo de bits HECV de capa única como una "capa base". En extensiones multivistas o 3D, las capas adicionales pueden representar las vistas de video adicionales capturadas desde diferentes posiciones de cámara o, por ejemplo, información de profundidad. En extensiones de escalabilidad, las capas adicionales pueden representar el video con una resolución de imágenes de video adicional, normalmente mayor, o con una mayor fidelidad de píxel, o con otros espacios de color, o algo similar, proporcionando una calidad de video mejorada en relación a la capa base.

20 Se usan decodificadores específicos para decodificar los flujos de bits de la HECV con múltiples capas, esto es decodificadores escalables o multivista/3D. Para decodificar los flujos de bits multicapa, en necesaria información sobre las dependencias de decodificación entre las capas. Esta información necesita ser señalizada en el flujo de bits. La información también puede ser usada por los elementos de red para identificar las capas que se pueden
25 descartar de la transmisión si es necesaria la adaptación de la tasa de bit, por ejemplo en caso de una congestión de red, o la adaptación del formato, por ejemplo en caso de que un dispositivo de destino sólo pueda decodificar o presentar una determinada resolución máxima, o la adaptación 2D/3D, por ejemplo en caso de que el dispositivo de destino sólo pueda decodificar o presentar un determinado número de vistas.

30 La información de dependencia en HEVC se obtiene normalmente de los así llamados conjuntos de parámetros, tales como el Conjunto de Parámetros de Imagen (PPS), el Conjunto de Parámetros de Secuencia (SPS), o el Conjunto de Parámetros de Video (VPS), junto con otra información. Normalmente, cada conjunto de parámetros se encapsula en una unidad de la Capa de Abstracción de Red (NAL), esto es un paquete en el flujo de bits de video. Ya que los conjuntos de parámetros contienen información que es esencial para la decodificación, pueden ser
35 enviados repetidamente en el flujo de bits, o ser transportados en una transmisión "fuera de banda", esto es transmitidos de manera separada del flujo de datos restante, por ejemplo sobre una conexión de confianza. Dicha transmisión fuera de banda puede ocurrir, por ejemplo durante la configuración de sesión, por ejemplo usando el Protocolo de Descripción de Sesión (SDP).

40 Si los conjuntos de parámetros se envían en el inicio de sesión, la cantidad de datos del conjunto de parámetros tiene un impacto en la duración de la transmisión y así en el tiempo de inicio de sesión. Si los conjuntos de parámetros se envían "en banda", esto es en el flujo de bits, el tamaño de los conjuntos de parámetros se repiten en el flujo de bits por razones de resistencia al error. Por estas razones es importante que la información transportada en los conjuntos de parámetros se exprese de manera compacta.

45 Un documento del Equipo de Colaboración Conjunto sobre la Codificación de Video (JCT-VC) de ITU-T SG 16 WP 3 y ISO/IEC JTC 1/SC 29/WG 11 en la 11ª Reunión: Shangai, CN 10-19 Octubre del 2012 designado JCTVC-K1007 relacionado con la cabecera de la unidad NAL y el conjunto de parámetros diseñados para las extensiones HEVC incluye especificaciones de los diseños de los conjuntos de parámetro para las extensiones de codificación HEVC multivista/3D y escalable. Según este documento, las dependencias de capa se señalizan como parte de la estructura de sintaxis *vps_extension* como se indica a continuación:

50

```

vps_extension() {
    ...
    para(i = 1; i ≤ capas_max_vps_menos1; i++){
        // dependencia de capa
5         num_capas_ref_directa[i]          u(6)
        para(j = 0; j < num_capas_ref_directa[i]; j++){
            id_capa_ref[i][j]              u(6)
        }
    }
}

```

10 num_capas_ref_directa[i] especifica el número de capas de las que la capa i-ésima depende directamente. id_capa_ref[i][j] identifica la capa j-ésima de la que la capa i-ésima depende directamente.

15 La solución anteriormente mencionada requiere muchos bits para señalar las dependencias de capa en el VPS. En concreto, para cada capa sobre la capa base en uso, se usan seis bits para codificar el número de capas de referencia y otros seis bits se usan para identificar cada capa de referencia. Esto permite señalar dependencias para los casos relevantes, sin embargo puede ser ineficiente en términos de uso de bits.

Compendio

Es un objetivo general proporcionar un procesamiento eficiente de un flujo de video multicapa.

Es un objetivo particular señalar las dependencias de capa de una manera eficiente para un flujo de video multicapa.

20 Estos y otros objetivos se cumplen por la invención según lo establecido en las reivindicaciones adjuntas.

Breve descripción de los dibujos

Las realizaciones, junto con los objetivos y ventajas adicionales de las mismas, se pueden entender mejor haciendo referencia a la siguiente descripción tenida en cuenta junto con los dibujos adjuntos, en los cuales:

- La Fig. 1 ilustra de manera esquemática un ejemplo de un flujo de video multicapa;
- 25 La Fig. 2 ilustra de manera esquemática otro ejemplo de un flujo de video multicapa;
- La Fig. 3 ilustra un ejemplo de una parte codificada de una imagen en un flujo de video multicapa;
- La Fig. 4 ilustra un ejemplo de un paquete de datos de un flujo de video multicapa;
- La Fig. 5 es un diagrama de flujo que ilustra un método para decodificar un flujo de video multicapa codificado según una realización;
- 30 La Fig. 6 es un diagrama de flujo que ilustra un método para determinar la relación de decodificación para un flujo de video multicapa según una realización;
- La Fig. 7 es un diagrama de flujo que ilustra un método para determinar la relación de decodificación para un flujo de video multicapa según otra realización;
- La Fig. 8 es un diagrama de flujo que ilustra un paso opcional, adicional del método de la Fig. 7;
- 35 La Fig. 9 es un diagrama de flujo que ilustra un método para codificar un flujo de video multicapa según una realización;
- La Fig. 10 es un diagrama de flujo que ilustra un método para determinar la información de relación de codificación para un flujo de video multicapa según una realización;
- La Fig. 11 es un diagrama de flujo que ilustra un método para determinar la información de relación de codificación para un flujo de video multicapa según otra realización;
- 40 La Fig. 12 es un diagrama de bloques esquemático de un decodificador según una realización;

La Fig. 13 es un diagrama de bloques esquemático de un dispositivo para determinar la relación de decodificación según una realización;

La Fig. 14 es un diagrama de bloques esquemático de un dispositivo para determinar la relación de decodificación según otra realización;

5 La Fig. 15 es un diagrama de bloques esquemático de un codificador según una realización;

La Fig. 16 es un diagrama de bloques esquemático de un dispositivo para determinar la relación de codificación según una realización;

La Fig. 17 es un diagrama de bloques esquemático de un dispositivo para determinar la relación de codificación según otra realización;

10 La Fig. 18 es un diagrama de bloques esquemático que ilustra un ejemplo de una implementación particular de dispositivos según las realizaciones;

La Fig. 19 es un resumen esquemático de la codificación de un flujo de video multicapa, la transmisión de un flujo de video multicapa codificado, la extracción opcional de un subflujo de video multicapa y la decodificación y presentación de un flujo de video multicapa;

15 La Fig. 20 es un diagrama que ilustra el número de bits requeridos para señalar las dependencias de capa cuando cada capa con un índice de capa $i \geq 1$ tiene una capa de referencia;

La Fig. 21 es un diagrama que ilustra el número de bits requeridos para señalar las dependencias de capa cuando cada capa con un índice de capa $i \geq 2$ tiene dos capas de referencia y la capa con índice de capa $i = 1$ tiene una capa de referencia;

20 La Fig. 22 es un diagrama que ilustra el número de bits requeridos para señalar las dependencias de capa cuando cada capa con un índice de capa $i \geq 1$ tiene el número máximo de capas de referencia que es posible; y

La Fig. 23 ilustra de manera esquemática otro ejemplo de un flujo de video multicapa.;

Descripción detallada

En todos los dibujos, los mismos números de referencia se usan para elementos similares o correspondientes.

25 Las presentes realizaciones se refieren de manera general a flujos y secuencias de video multicapa y en particular a la codificación y decodificación de dichos flujos de video multicapa. Las realizaciones proporcionan una manera eficiente de señalización de las dependencias de capas para el flujo de video multicapa, permitiendo de este modo una reducción en el número de bits requeridos para señalar dichas dependencias de capas comparado con las técnicas de la técnica anterior.

30 El video multicapa se refiere de manera general al video multivista, video 3D y/o video escalable. En el video multivista una escena se captura desde diferentes vistas de cámara como se indica en la Fig. 1 que muestra tres de dichas vistas de cámara. El flujo 1 de video multicapa comprende entonces, en este ejemplo, las imágenes 20, 22, 24 de las tres diferentes capas o vistas 10, 12, 14. Las imágenes 20, 22, 24 en el flujo 1 de video multicapa se pueden codificar o decodificar usando imágenes de referencia que pertenecen a la misma capa o vista 10, 12, 14 o que pertenecen a otra capa o vista 10, 12, 14. La Fig. 1 ilustra de manera esquemática las relaciones de codificación y decodificación entre las imágenes mediante flechas. Por ejemplo, la imagen 24 se codifica y decodifica respecto a una imagen anterior 22, según el orden de codificación y decodificación, que pertenece a la misma capa o vista 12. Esta imagen 22, sin embargo, se codifica y decodifica usando una imagen anterior de la misma capa o vista 12 y otra imagen 20 que pertenece a otra capa o vista 10. En la Fig. 1 la capa o vista 10 es la capa o vista base, mientras que las otras capas o vistas 12, 14 son capas o vistas adicionales, también conocidas como capas de mejora o vistas dependientes.

40 En el video 3D la capa o las capas adicionales pueden llevar información de profundidad u otros datos junto con los datos de píxeles de la capa base para generar video 3D.

45 En el video escalable la capa o las capas adicionales pueden representar el video en resoluciones de imágenes de video mayores, mayor fidelidad de píxeles y/u otros espacios de color para proporcionar una calidad de video mejorada en relación a la capa base del video escalable.

La Fig. 23 ilustra de manera esquemática un ejemplo de un flujo 1 de video multicapa para video escalable. La figura indica la organización de las imágenes 20, 22, 24 en varias capas escalables 10, 12, 14, donde la capa 10 indica la capa base.

50 La Fig. 2 ilustra de manera esquemática la escalabilidad temporal en la cual cada capa 12, 14 de mejora perfecciona la tasa de fotogramas.

Una imagen de un flujo de video multicapa puede estar compuesta por una o más fracciones. Una fracción es una parte de una imagen decodificable de manera independiente. En otras palabras, si una fracción se pierde, las otras fracciones de esa imagen son aún decodificables.

La codificación de una fracción genera una representación codificada 30 de la fracción que comprende una cabecera 32 de fracción y datos 34 de fracción como se muestra en la Fig. 3. La presentación codificada 30 se emite desde el proceso de codificación como una así llamada unidad 44 de Capa de Adaptación de Red (NAL) como se muestra en la Fig. 4. A la unidad NAL 44 se le pueden añadir cabeceras 42 para formar un paquete 40 de datos que se puede transmitir como parte de un flujo de video multicapa codificado desde un codificador a un decodificador. Por ejemplo, se podrían añadir las cabeceras 42 del Protocolo de Transporte en Tiempo Real (RTP), el Protocolo de Datagramas de Usuario (UDP) y el Protocolo de Internet (IP) a la unidad NAL 44. Esta forma de empaquetado de unidades NAL 44 constituye simplemente un ejemplo en relación con el transporte de video. Son posibles otros enfoques en el manejo de las unidades NAL 44, tales como el formato de archivo, los flujos de transporte MPEG-2, los flujos de programa MPEG-2, etc. La Fig. 19 ilustra de manera esquemática la generación de un flujo 2 de video multicapa desde un flujo 1 de video multicapa de entrada que comprende múltiples capas 10, 12, 14 de imágenes 20, 22, 24.

Un aspecto de las realizaciones relaciona a un método de decodificación un flujo 2 de video multicapa codificado que define múltiples capas 10, 12, 14 de imágenes 20, 22, 24. Cada capa 10, 12, 14 de las múltiples capas 10, 12, 14 tiene un respectivo identificador de capa. El método comprende la recuperación, para una capa 12 con un índice de capa de las múltiples capas 10, 12, 14, de la información de relación de decodificación basada en el flujo 2 de video multicapa codificado. La información de relación de decodificación define un respectivo índice de capa de cualquier capa 10 de referencia de las múltiples capas 10, 12, 14 sobre la que la capa 12 depende directamente. El método también comprende la correspondencia, para cada capa 10 de referencia y para la capa 12, de su índice de capa a un identificador de capa basado en la información de correspondencia de una relación de correspondencia jerárquica entre los identificadores de capa y los índices de capa. La información de correspondencia es recuperada basada en el flujo 2 de video multicapa codificado. El método además comprende la decodificación de una imagen 22 de la capa 12 basada en al menos una imagen 20 anteriormente decodificada en una capa 10 de las múltiples capas 10, 12, 14 identificada en base a los identificadores de capa hechos corresponder de los índices de capa.

Una realización de este aspecto se describirá ahora con más detalle con referencia al diagrama de flujo de la Fig. 5. La Fig. 5 por lo tanto describe un método para decodificar un flujo de video multicapa codificado, tal como un flujo de video multivista, un flujo de video 3D o un flujo de video escalable. El flujo de video multicapa define o comprende múltiples capas de imágenes. Cada una de dichas capas del flujo de video multicapa tiene un respectivo identificador de capa.

El método normalmente comienza en el paso S1 donde se recupera la información de relación de decodificación para una capa (la actual) de las múltiples capas en el flujo de video multivista. La capa actual para la que se recupera la información de relación de decodificación tiene un índice de capa. Así, la capa actual tiene, además del anterior identificador de capa identificado que es un identificador general o "global" para la capa actual, un índice de capa que es un índice empleado para señalar las dependencias de capa. Por lo tanto, el índice de capa se configura preferiblemente para ser usado en lugar del identificador de capa para la señalización de las dependencias de capa.

La información de relación de decodificación recuperada en el paso S1 define un respectivo índice de capa de cualquier capa de referencia de las múltiples capas en el flujo de video multivista, sobre las cuales la capa actual depende directamente. Así, la información de relación de decodificación define un respectivo índice de capa a cada capa de referencia de la que la capa actual depende directamente.

Como se usa en la presente memoria una capa X depende directamente (o es directamente dependiente) de una capa Y si por ejemplo se necesita utilizar una predicción de pixel de la capa Y en la capa X para decodificar la capa X. Así una imagen que pertenezca a la capa X se codifica y por ende se decodifica basada en una imagen de referencia que pertenece a la capa Y.

En consecuencia, si una capa X depende directamente (o es directamente dependiente) de una capa Y pero no depende directamente (no es directamente dependiente) de una capa Z pero la capa Y directamente (o indirectamente) depende (es directa o indirectamente dependiente) de la capa Z entonces la capa X depende indirectamente (es indirectamente dependiente) de la capa Z. Una capa X depende (es dependiente) de una capa Z si bien depende directamente (es directamente dependiente) de la capa Z o indirectamente depende (es indirectamente dependiente) de la capa Z.

Si una capa X depende directamente de una capa Y esta capa Y es una capa de referencia directa para la capa X.

Con referencia a la Fig. 2 o 23, la capa con número de referencia 12 depende directamente de la capa base 10. La capa con número de referencia 14 depende directamente de la capa con número de referencia 12 e indirectamente depende de la capa base 10.

La información de relación de decodificación recuperada en el paso S1 enumera preferiblemente los respectivos índices de capa de las capas de referencia, de los cuales la capa actual depende. Por lo tanto, también las capas de referencia tienen un respectivo índice de capa además de su respectivo identificador de capa.

5 En una realización, la información de relación de decodificación está presente en un conjunto de parámetros incluido como una parte del flujo de video multicapa, por ejemplo en una unidad NAL, o proporcionado fuera de banda con respecto al flujo de video multicapa. El conjunto de parámetros preferiblemente tiene un identificador de conjunto de parámetros asignado que permite la identificación del conjunto de parámetros relevante. Dicho identificador de conjunto de parámetros preferiblemente se incluye después en el flujo de video multicapa, tal como presente en una cabecera de fracción de una representación codificada de una fracción. De manera alternativa, el (primer) conjunto de parámetros se incluye en un segundo conjunto de parámetros con un segundo identificador de conjunto de parámetros que se incluye en el flujo de video multicapa por ejemplo en la cabecera de fracción. En una alternativa adicional, el (primer) identificador de conjunto de parámetros se incluye en un segundo conjunto de parámetros con un segundo identificador de conjunto de parámetros que está incluido en un tercer conjunto de parámetros que tiene un tercer identificador de conjunto de parámetros que está incluido en el flujo de video multicapa por ejemplo en la cabecera de fracción.

En un primer ejemplo, la información de relación de decodificación se podría incluir en un VPS identificado por un identificador de VPS incluido en un SPS que está identificado por un identificador de SPS incluido en un PPS que tiene un identificador de PPS asociado incluido en una cabecera de fracción.

20 En un segundo ejemplo, la información de relación de decodificación se podría incluir en un SPS que está identificado por un identificador SPS incluido en un PPS que tiene un identificador PPS asociado incluido en una cabecera de fracción.

En un tercer ejemplo, la información de relación de decodificación se podría incluir en un PPS que tiene un identificador de PPS asociado incluido en una cabecera de fracción.

25 La información de relación de decodificación se podría, de manera alternativa o adicional, incluir en un mensaje suplementario, por ejemplo en un mensaje de Información de Mejora Suplementaria (SEI), tal como en un mensaje SEI de conjuntos de parámetros activos.

30 En otra realización, la información de relación de decodificación está presente en los datos de la extensión multicapa incluidos como parte del flujo de video multicapa o proporcionados fuera de banda. Dichos datos de la extensión multicapa podrían estar en forma de, por ejemplo, datos de la extensión multivista, datos de la extensión 3D o datos de la extensión escalable. Una alternativa es tener datos de extensión relacionados a un conjunto de parámetros, tales como los datos de la extensión del VPS, los datos de la extensión del PPS, los datos de la extensión del SPS, o los datos de la extensión del PPS, que podrían estar presentes como una extensión opcional de las estructuras de datos del VPS, SPS, o PPS, por ejemplo dependiendo de una bandera. Esto es, un VPS/SPS/PPS podría tener una bandera de extensión que indique si están presentes datos adicionales, que extienden el VPS/SPS/PPS.

35 En una realización adicional, la información de relación de decodificación está presente en las representaciones codificadas del flujo de video multicapa codificado. Por ejemplo, la información de relación de decodificación podría estar presente en una respectiva cabecera de fracción de una o más representaciones codificadas. En tal caso, no son necesarios identificadores a otras estructuras de datos de o asociados con el flujo de video multicapa. Sin embargo, este enfoque aumenta el tamaño total en términos de número de bits de las representaciones codificadas que llevan la información de correspondencia.

40 El siguiente paso S2 de la Fig. 5 comprende la correspondencia del índice de capa con un identificador de capa para cada capa de referencia definida por la información de relación de decodificación recuperada en el paso S1 y para la capa actual. Esta correspondencia de índices a identificadores en el paso S2 se realiza basada en la información de correspondencia de una relación de correspondencia jerárquica entre los identificadores de capa y los índices de capa. Esta información de correspondencia se recupera preferiblemente basada en el flujo de video multicapa codificado. La información de correspondencia se podría recuperar de un conjunto de parámetros, de una extensión de un conjunto de parámetros, de otra estructura de datos de o asociada con el flujo de video multicapa codificado, de una representación codificada de una fracción en el flujo de video multicapa codificado, etc. La información de correspondencia se proporciona preferiblemente en la misma estructura de datos que la información de relación de decodificación pero podría, de manera alternativa, estar presente en otra estructura de datos.

45 La información de correspondencia define una relación de correspondencia jerárquica entre los identificadores de capa y los índices de capa. La correspondencia jerárquica indica tal como se usa en la presente memoria que los índices de capa se asignan a diversas capas de acuerdo con las dependencias de capa. En una realización particular, las capas se indexan de manera jerárquica de manera tal que una capa con un índice i de capa dado puede tener sólo capas de referencia de índices j menores, esto es $j < i$. Así, el respectivo índice j de capa asignado a la capa o a las capas de las múltiples capas en el flujo de video multicapa de la que la capa actual con índice i de capa es dependiente es preferiblemente menor que el índice i de capa. Esto implica además que una capa con índice i de capa puede tener a los sumo i capas de referencia, $j=0, \dots, i-1$.

El uso de los índices de capa en lugar de los identificadores de capa permite una manera más eficiente en bits de señalar las dependencias de capa. Generalmente, los identificadores de capa, como se indica en la sección de compendio, son de seis bits por HEVC. Los identificadores de capa pueden, sin embargo, ser asignados de manera poco dispersa, esto es algunos valores de los identificadores de capa de 6 bits nunca se usan. La correspondencia jerárquica entre los identificadores de capa y los índices de capa según una realización permite el uso de índices de capa que se asignan de manera no dispersa, esto es todos los valores de los índices de capa se usan.

El método como se muestra en la Fig. 5 comprende además un paso S3, que comprende decodificar una imagen de la capa actual basada en al menos una imagen previamente decodificada en una capa de las múltiples capas en el flujo de video multicapa, donde esta capa se identifica en base a los identificadores de capa hechos corresponder a partir de los índices de capa. Por lo tanto, la información de relación de decodificación recuperada en el paso S1 se usa para proporcionar los índices de capa de cualquier capa de referencia. La información de correspondencia se usa después en el paso S2 para hacer corresponder el respectivo índice de capa definido por la información de relación de decodificación a un respectivo identificador de capa y también hacer corresponder el índice de capa de la capa actual a un identificador de capa. Una imagen, o más correctamente al menos una imagen, en la capa actual se decodifica después basada en al menos una imagen previamente decodificada, esto es una imagen de referencia, en una capa identificada en base a al menos uno de los identificadores de capa obtenidos en la correspondencia hecha en el paso S2. La imagen de referencia usada en el paso S3 de decodificación podría ser una imagen de referencia que pertenece a la capa actual o a otra capa del flujo de video multicapa. Si se ha de usar más de una imagen de referencia como base de decodificación para la imagen a decodificar en el paso S3 estas imágenes de referencia podrían pertenecer a la misma o a diferentes capas del flujo de video multicapa.

Los índices de capa empleados por las realizaciones para señalar las dependencias de capa pueden ser diferentes de los identificadores de capa. El número de capas es normalmente menor que el número de diferentes identificadores de capa posibles, tales como 64 posibles identificadores de capa para una representación del identificador de 6 bits. Así, el número de bits empleados para señalar un índice de capa es normalmente menor que el número de bits usado para señalar un identificador de capa. Además, mientras que el identificador de capa está representado por un elemento de sintaxis de un número de bits fijo, tal como un código $u(6)$, que requiere seis bits para representar el identificador de capa, un índice de capa está representado preferiblemente por un elemento de sintaxis de una longitud variable, tal como un código $u(v)$. A este respecto, el número v de bits usado para indicar el índice de capa se puede determinar basado en el máximo número de capas usadas, por ejemplo como $v = \text{ceil}(\log_2(\text{capas_max}))$, donde capas_max indica el número máximo de capas usado, esto es $\text{capas_max} = \text{capas_max_vps_menos1} + 1$, $\log_2(x)$ indica un logaritmo en base 2 de x y $\text{ceil}(y)$ indica el menor número entero mayor o igual que y .

En un ejemplo de implementación, los siguientes elementos de sintaxis se podrían usar para indicar la información de relación de decodificación y la información de correspondencia.

```

35  vps_extension() {                               Descriptor
    ...
    para(i = 1; i ≤ capas_max_vps_menos1; i++){
        nuh_en_id_capa[i]                          u(6)
    }
40  ...
    para(i = 1; i ≤ capas_max_vps_menos1; i++){
        num_capas_ref_directa[i]                   u(v)
        para(j=0; j < num_capas_ref_directa [i];j++)
            índice_capa_ref[i][j]                  u(v)
45  }
}

```

En el primer bucle i representa el índice de capa y $\text{nuh_en_id_capa}[i]$ indica el identificador de capa para el índice de capa i . Por lo tanto, estos elementos de sintaxis constituyen la información de correspondencia. En el segundo bucle $\text{num_capas_ref_directa}[i]$ especifica el número de capas de las que la capa con el índice de capa i directamente depende. $\text{índice_capa_ref}[i][j]$ identifica el índice de capa de la capa j -ésima de la que la capa con índice de capa i depende directamente. En esta implementación de ejemplo se usa una relación jerárquica de capa de manera tal que una capa con índice i pueda sólo tener capas de referencia que tengan índices $j < i$.

La información de correspondencia como se usa en el paso S2 y la información de relación de decodificación recuperada en el paso S1 juntas permiten una señalización eficiente de las dependencias de capa explotando una relación de capa jerárquica y una asignación densa de índices de capa. Como consecuencia, el número de bits requeridos para señalar las dependencias de capa se puede reducir comparado a usar identificadores de capa.

- 5 Así, el uso de índices de capa para señalar las dependencias de capa en lugar de los identificadores de capa y el uso de elementos de sintaxis indexadores de capa jerárquicos o palabras de código de mínima longitud pueden ser usados para definir la información de relación de decodificación.

En ciertas realizaciones los índices de capa pueden ser los mismos que los identificadores de capa, esto es índice_capa = id_capa. Sin embargo, en tales casos los identificadores de capa se organizan preferiblemente de manera jerárquica con respecto a las dependencias de capa como se describe en la presente memoria. En tales casos, los datos de extensión del VPS descritos anteriormente se podrían representar según se muestra a continuación:

```

vps_extension() {
    ...
    bandera_presente_id_capa_nuh_vps
    para(i = 1; i ≤ capas_max_vps_menos1; i++){
        si(bandera_presente_id_capa_nuh_vps)
            nuh_en_id_capa[i]                u(6)
    }
    ...
}

```

Si bandera_presente_id_capa_nuh_vps es igual a cero entonces no hay correspondencia explícita entre los identificadores de capa y los índices de capa. En tal caso, se puede concluir implícitamente que nuh_en_id_capa [i] = i.

- 25 Por lo tanto, en una realización el paso S2 de correspondencia de la Fig. 5 comprende recuperar una bandera bandera_presente_id_capa_nuh_vps basada en el flujo de video multicapa codificado, tal como de una extensión del VPS como se indicó anteriormente. El paso de correspondencia S2 comprende además fijar, para cada capa de referencia y para la capa y si bandera_presente_id_capa_nuh_vps = 0, su identificador de capa igual a su índice de capa. El paso de correspondencia S2 también comprende recuperar, para cada capa de referencia y para la capa y si bandera_presente_id_capa_nuh_vps = 1, su identificador de capa de un vector nuh_en_id_capa [i], i ∈ [1, capas_max_vps_menos1], en donde capas_max_vps_menos1 + 1 indica el número máximo de capas y nuh_en_id_capa[i] indica el identificador de capa para la capa con el índice de capa i.

Otro aspecto de las realizaciones se relaciona a un método para codificar un flujo 1 de video multicapa que define múltiples capas 10, 12, 14 de imágenes 20, 22, 24. Cada capa 10, 12, 14 de las múltiples capas 10, 12, 14 tiene un respectivo identificador de capa. El método comprende hacer corresponder de manera jerárquica, para cada capa 10, 12, 14 de las múltiples capas 10, 12, 14, un identificador de capa de la capa 10, 12, 14 a un índice de capa basado en las dependencias de codificación entre las múltiples capas 10, 12, 14. El método también comprende determinar la información de relación de codificación que define un respectivo índice de capa de cualquier capa 10 de referencia de las múltiples capas 10, 12, 14, de las que la capa 12 de las múltiples capas 10, 12, 14 depende directamente. El método además comprende generar un flujo 2 de video multicapa mediante la codificación de las imágenes 20, 22, 24 de las múltiples capas 10, 12, 14 basado en las múltiples dependencias de codificación y asociando la información de relación de codificación con el flujo 2 de video multicapa codificado.

La Fig. 9 es un diagrama de flujo que ilustra una realización del método de codificación de un flujo de video multicapa. En método comienza en el paso S40, que comprende hacer corresponder de manera jerárquica un respectivo identificador de capa de cada capa en el flujo de video multicapa a un respectivo índice de capa. Esta correspondencia entre identificadores de capa e índices de capa se realiza basada en las dependencias de codificación entre las múltiples capas del flujo de video multicapa. Así, la asignación de índices de capa a las capas se realiza basada en como las diversas capas dependen las unas de las otras. En concreto, la asignación de los índices de capa durante la correspondencia del paso S40 se realiza tal que una capa con un índice de capa asignado i sólo pueda tener capas de referencia, esto es capas de las que dependa, que tengan índices j de capa, donde j es más pequeño que i. Así, la información dada de como las capas del flujo de video multivista dependen de los otros identificadores de capa de la capas se hace corresponder a los índices de capa para formar la relación de capa jerárquica.

El paso S41 del método comprende determinar la información de relación de codificación que define un respectivo índice de capa de cualquier capa de referencia de la que una capa actual del flujo de video multicapa depende directamente. Así, la información de relación de codificación especifica el índice de capa o los índices de todas las capas de referencia para la capa actual.

5 Un flujo de video multivista es generado en el paso S42 mediante la codificación de las imágenes de las múltiples capas basadas en y según a las dependencias de codificación entre las múltiples capas. Así, P o B fracciones en las diversas capas se codifican usando una o más imágenes de referencia que pertenecen a la misma capa que la fracción P o B o que pertenecen a una de las capas de referencia para la capa de la fracción P o B. La imagen podría, de manera alternativa, ser una imagen I. En tal caso, las imágenes de referencia no se manejan de la misma manera que las imágenes de referencia “temporales” normales. Aún una imagen I, por ejemplo en una extensión de HEVC escalable podría tener una imagen de referencia, para “predicción entre capas”.

10 El flujo de video multivista comprende preferiblemente las respectivas representaciones codificadas 30 de fracciones como se muestra en la Fig. 3 para cada fracción del flujo de video multicapa. Las representaciones codificadas 30 se empaquetan de manera ventajosa en unidades NAL 44 como se muestra en la Fig. 4. El flujo de video multicapa codificado puede comprender también unidades NAL 44 que comprenden otros datos, tales como conjuntos de parámetros, que las representaciones codificadas 30.

15 La información de relación de codificación determinada en el paso S41 se asocia, en el paso S43 con el flujo de video multicapa codificado generado en el paso S42. Este paso S43 se podría realizar antes del paso S42 o al menos parcialmente en paralelo con el paso S42. La asociación de la información de relación de codificación con el flujo de video multicapa se podría realizar según diversas realizaciones como se ha descrito anteriormente en la presente memoria. Por ejemplo, la información de relación de codificación podría estar presente en un conjunto de parámetros en cuyo caso se incluye un identificador de conjunto de parámetros en una cabecera de fracción de al menos una representación codificada de una fracción en el flujo de video multicapa o en otro conjunto de parámetros que es identificable basado en los datos incluidos en la cabecera de fracción. De manera alternativa, la información de relación de codificación podría estar presente en una extensión del conjunto de parámetros o de hecho en las representaciones codificadas de fracciones.

20 En una realización particular la información de correspondencia empleada en el paso S40 para hacer corresponder de manera jerárquica un respectivo identificador de capa a un respectivo índice de capa se genera y se asocia preferiblemente con el flujo de video multicapa. En tal caso, esta información de correspondencia se proporciona preferiblemente en la misma estructura de datos que la información de relación de codificación según se discutió anteriormente en la presente memoria.

25 Los aspectos de las realizaciones también se relacionan a los métodos de determinación de las relaciones de decodificación para un flujo de video multicapa codificado y de determinación de las relaciones de codificación para un flujo de video multicapa. Estas realizaciones se podrían aplicar de manera ventajosa al método de decodificación como se muestra en la Fig. 5 o al método de codificación como se muestra en la Fig. 9. De manera alternativa, estos métodos de determinación de las relaciones de codificación/decodificación se podrían usar de manera separada a partir del método de la Fig. 5 y del método de la Fig. 9.

30 Si estos métodos de determinación de las relaciones de codificación/decodificación se usan separados del método de la Fig. 5 y del método de la Fig.9, los métodos podrían bien usar índices de capa para las múltiples capas en el flujo de video multicapa o identificadores de capa en el flujo de video multicapa. Así, en el último caso no es necesaria la correspondencia explícita de los identificadores de capa en índices de capa. Para tales caso, el índice de capa/índices de capa como se usan en la presente memoria con referencia a los métodos de determinación de las relaciones de codificación/decodificación se pueden reemplazar por un identificador de capa/identificadores de capa. Se prefiere después que los identificadores de capa estén organizados jerárquicamente.

35 Un aspecto de las realizaciones se relaciona a un método de determinación de la relación de decodificación para un flujo 2 de video multicapa codificado que define múltiples capas 10, 12, 14 de las imágenes 20, 22, 24. El método comprende la recuperación basada en el flujo 2 de video multicapa codificado, de al menos una bandera de decodificación directa que indica una relación de codificación directa entre una capa 12 con índice i de capa de las múltiples capas 10, 12, 14 y una capa 10 con índice j de capa de las múltiples capas 10, 12, 14 $i \neq j$. El método también comprende determinar la información que define cualquier relación de decodificación entre las múltiples capas 10, 12, 14 basada en al menos una bandera de decodificación directa.

40 La Fig. 6 es un diagrama de flujo que ilustra una realización de este método para determinar la relación de decodificación para un flujo de video multicapa que define múltiples capas de imágenes. El método comienza en el paso S10, que comprende la recuperación de al menos una bandera de decodificación directa basada en el flujo de video multicapa. La al menos una bandera de decodificación directa podría estar presente en un conjunto de parámetros, tales como el PPS, SPS o VPS, una extensión del conjunto de parámetros, tal como una extensión del PPS, una extensión del SPS o una extensión del VPS, estar presente en una representación codificada de una fracción, tal como en una cabecera de fracción, del flujo de video multicapa codificado, o de otra forma ser proporcionada en una estructura de datos de o asociada con el flujo de video multicapa codificado. Así, el flujo de

video multicapa codificado bien comprende la al menos una bandera de decodificación directa o comprende información, tal como un identificador, que permite la identificación de la estructura de datos que comprende la al menos una bandera de decodificación directa. La recuperación en el paso S10 podría por lo tanto ser realizada mediante el análisis y la decodificación de un conjunto de parámetros o de una extensión del conjunto de parámetros para obtener la bandera de decodificación directa. De manera alternativa y si el conjunto de parámetros o la extensión del conjunto de parámetros ya se ha analizado y decodificado y los datos del conjunto de parámetros o de la extensión del conjunto de parámetros se han almacenado de forma decodificada en una memoria, el paso S10 podría comprender la lectura o la recuperación de al menos una bandera de decodificación directa desde dicha memoria. Si la al menos una bandera de decodificación directa está presente en una representación codificada de un paso S10 de fracción preferiblemente se recupera la al menos una bandera de decodificación directa en relación con el análisis y la decodificación de la cabecera de fracción en la representación codificada.

La al menos una bandera de decodificación directa recuperada en el paso S10 indica una relación de codificación directa entre una capa con un índice i de capa (capa actual) de las múltiples capas y una capa con un índice j de capa de las múltiples capas, donde $i \neq j$. Así, la al menos una bandera de decodificación directa indica si hay alguna codificación directa y por tanto una relación de decodificación entre las capas con índices i y j de capa. En una realización particular, la bandera de decodificación directa indica si la capa con índice j de capa es una capa de referencia directa para la capa con índice i de capa.

La al menos una bandera de decodificación directa es preferiblemente una bandeja de 1 bit, esto es puede asumir bien el valor 0_{bin} o el valor 1_{bin} . En tal caso, una bandera de decodificación directa fijada a 1_{bin} podría indicar que la capa con índice j de capa es una capa de referencia directa para la capa con índice i de capa, mientras que en cambio una bandera de decodificación directa fijada a 0_{bin} indica que la capa con índice j de capa no es una capa de referencia directa para la capa con índice i de capa.

El método también comprende determinar, en el paso S11, la información que define cualquier relación de codificación entre las múltiples capas en el flujo de video multicapa basada en la al menos una bandera de decodificación directa recuperada en el paso S10. Así, la al menos una bandera de decodificación directa recuperada en el paso S10 se usa para determinar la información que define la relación de decodificación entre las diversas capas en el flujo de video multicapa.

El método termina entonces en una realización.

En una realización alternativa la información determinada en el paso S11 se usa como la información de relación de decodificación en el método de decodificación de un flujo de video multicapa codificado descrito en relación con la Fig. 5. En tal caso, el método continúa a partir del paso S11 de la Fig. 6 hasta el paso S2 de la Fig. 5.

Esto significa que en esta realización alternativa los índices de capa se asignan jerárquicamente a las diversas capas para que una capa con índice de capa i tenga cualquier capa de referencia que tenga índice $j < i$. En tal caso, las banderas de decodificación directa permiten una manera muy compacta y eficiente de señalar las dependencias de capa para el flujo de video multicapa.

La determinación de la información en el paso S11 podría, lo cual se discute en más detalle más adelante, involucrar la generación o el cálculo de la información usando al menos una bandera de decodificación directa recuperada en el paso S10. En una realización alternativa, la determinación de la información que define cualquier relación de decodificación entre las múltiples capas podría simplemente ser la determinación del valor o de los valores de la al menos una bandera de decodificación directa. Así, en este caso los respectivos valores de la al menos una bandera de decodificación directa según lo decodificado podrían constituir la información determinada o proporcionada en el paso S11.

En una realización del paso S10 una respectiva bandera de dependencia directa se recupera para la capa con índice i de capa para cada índice j de capa basada en el flujo de video multicapa codificado. En esta realización $j < i$ y la bandera de dependencia directa indica si la capa con índice j de capa es una capa de referencia directa para la capa con índice i de capa.

El paso S11 comprende preferiblemente, en esta realización, determinar la información que define cualquier capa con índice $j < i$ de capa de las cuales la capa con índice i de capa depende basada en la bandera o banderas de dependencia directa recuperadas en el paso S10. Así, la bandera o banderas de dependencia directa obtenidas en el paso S10 se usan para determinar la información que define la capa o las capas con los respectivos índices de capa menores que el índice de capa de la capa actual (índice i de capa) de las que la capa actual depende de manera directa o indirecta.

En una realización el paso S10 comprende la recuperación, de un conjunto de parámetros de video o una extensión del conjunto de parámetros de video asociada con el flujo de video multicapa codificado, de la al menos una bandera de decodificación directa que indica la relación de codificación directa entre la capa con índice i de capa y la capa con índice j de capa.

En una realización particular el paso S10 preferiblemente comprende la recuperación, para la capa con índice i de capa, de una respectiva bandera de dependencia directa $bandera_dependencia_directa[i][j]$ para cada índice j de capa basada en el flujo de video multicapa codificado. En esta realización particular el paso S11 preferiblemente comprende la determinación de un conjunto o colección de banderas de dependencia $bandera_dependencia[i]$ para la capa con índice i de capa. Este conjunto de banderas de dependencia se determina preferiblemente como $bandera_dependencia[i] = bandera_dependencia_directa[i] + \sum_{j=0..j-1} (bandera_dependencia_directa[i][j] \times bandera_dependencia[j])$.

- 5
- $bandera_dependencia_directa[i]$ representa un vector, esto es una concatenación de elementos en $bandera_dependencia_directa[i][j]$;
- 10
- “+” representa una operación O lógica bit a bit;
 - $\sum_{j=0..j-1}(x(j))$ representa una operación O lógica bit a bit sobre todo $x(j)$ con $j = 0..i-1$; y
 - “x” representa una operación Y lógica bit a bit entre cada elemento en $bandera_dependencia[j]$ y $bandera_dependencia_directa[i][j]$

En tal caso, los elementos de sintaxis para señalar las banderas de dependencia directa podrían ser como se define a continuación.

```

15  para(i = 1; i ≤ capas_max_vps_menos1; i++){           Descriptor
        //dependencia de capa
        para(j = 0; j < i; j++){
                bandera_dependencia_directa[i][j]       u(1)
20    }

```

$bandera_dependencia_directa[i][j]$ igual a cero indica preferiblemente que la capa con el índice j de capa no es una capa de referencia directa para la capa con el índice i de capa. $bandera_dependencia_directa[i][j]$ igual a uno indica preferiblemente que la capa con el índice j de capa es una capa de referencia directa para la capa con el índice i de capa. Si no está presente la $bandera_dependencia_directa[i][j]$ se concluye que es igual a cero. $capas_max_vps_menos1 + 1$ indica el máximo número posible de capas en el flujo de video multicapa.

25 Si esta realización se usa en relación con el método de decodificación como se muestra en la Fig. 5 las banderas de dependencia directa y la información de correspondencia se podría señalar junta, tal como en una extensión del VPS como se muestra más adelante.

```

vps_extension() {           Descriptor
30    ...
        para(i = 1; i ≤ capas_max_vps_menos1; i++){
                nuh_en_id_capa[i]                       u(6)
        }
        ...
35    para(i = 1; i ≤ capas_max_vps_menos1; i++){
            para(j=0; j < i; j++)
                    bandera_dependencia_directa[i][j]   u(1)
        }
}

```

40 En otra realización el paso S10 de la Fig. 6 comprende la recuperación, para una capa con índice i de capa, de una bandera de referencia directa para cada índice j de capa basada en el flujo de video multicapa codificado. En esta realización $j > i$ y la bandera de referencia directa indica si la capa con índice i de capa es una capa de referencia directa para la capa con índice j de capa. El paso S11 comprende preferiblemente determinar la información que define cualquier capa con índice $j > i$ de capa para el cual la capa con índice i de capa es una capa de referencia (directa o indirecta) basada en la bandera o banderas de referencia directa recuperadas en el paso S10.

45

En una realización particular el paso S10 comprende preferiblemente la recuperación, para la capa con índice i de capa, de una respectiva bandera de referencia directa $bandera_referencia_directa[i][j]$ para cada índice j de capa basada en el flujo de video de multicapa. El paso S11 comprende preferiblemente determinar un conjunto o colección de banderas de referencia $bandera_referencia[i]$ para la capa con índice i de capa. Este conjunto de banderas de referencia se determina preferiblemente como $bandera_referencia[i] = bandera_referencia_directa[i] + \sum_{j=i+1 \dots imax}(bandera_referencia_directa[i][j] \times bandera_referencia[j])$.

- 5 • $bandera_referencia[i]$ representa un vector, esto es una concatenación de elementos en la $bandera_referencia_directa[i][j]$;
- 10 • $bandera_referencia_directa[i]$ representa un vector, esto es una concatenación de elementos en la $bandera_referencia_directa[i][j]$;
- “+” representa una operación O lógica bit a bit;
- $\sum_{j=i+1 \dots imax}(x(j))$ representa una operación O lógica bit a bit sobre todo $x(j)$ con $j = i+1 \dots imax$; donde $imax$ representa el índice de capa máximo; y
- 15 • “x” representa una operación Y lógica bit a bit entre cada elemento en la $bandera_referencia[j]$ y $bandera_referencia_directa[i][j]$

En dicho caso, los elementos de sintaxis para señalar las banderas de dependencia directa podrían ser como se definen a continuación.

```

para(i = 1; i ≤ capas_max_vps_menos1; i++){           Descriptor
    //dependencia de capa
20    para(j = i+1; j ≤ capas_max_vps_menos1; j++)
        bandera_referencia_directa[i][j]           u(1)
    }

```

Una $bandera_referencia_directa[i][j]$ igual a cero preferiblemente indica que la capa con índice i de capa no es una capa de referencia directa para la capa con índice j de capa. Una $bandera_referencia_directa[i][j]$ igual a uno preferiblemente indica que la capa con índice i de capa es una capa de referencia directa para la capa con índice j de capa. Si no hay presente una $bandera_referencia_directa[i][j]$ preferiblemente se concluye que es igual a cero.

Si esta realización se usa en relación con el método de decodificación que se muestra en la Fig. 5 las banderas de referencia directa y la información de correspondencia se podrían señalar juntas, tal como en una extensión del VPS como se muestra más adelante. Esta realización entonces tiene preferiblemente una relación de capas jerárquica ya que una capa con índice i de capa preferiblemente sólo puede ser una capa de referencia para capas con índice $j > i$ de capa.

```

vps_extension() {           Descriptor
    ...
    para(i = 1; i ≤ capas_max_vps_menos1; i++){
35        nuh_en_id_capa[i]           u(6)
    }
    ...
    para(i = 1; i ≤ capas_max_vps_menos1; i++){
        para(j=i+1; j ≤ capas_max_vps_menos1; j++)
40        bandera_referencia_directa[i][j]           u(1)
    }
}

```

En una realización adicional el paso S10 de la Fig. 6 comprende la recuperación, para una capa con índice i de capa, de una respectiva bandera de dependencia directa $bandera_dependencia_directa[i][j]$ para cada índice i de capa basada en dicho flujo de video multicapa codificado.

En una realización particular la bandera_dependencia_directa[i][j] igual a cero indica que la capa con índice j de capa no es una capa de referencia directa para la capa con índice i de capa. En consecuencia una bandera_dependencia_directa[i][j] igual a uno indica que la capa con índice j de capa puede ser una capa de referencia directa para la capa con índice i de capa.

5 El paso S11 de la Fig. 6 comprende en esta realización determinar las variables NumCapasRefDirect[i] e IdCapaRef[i][j], respectivamente, en base a las banderas de dependencia directa bandera_dependencia_directa[i][j]. NumCapasRefDirect[i] es una variable que representa el número de capas de referencia directa para la capa con índice i de capa y la variable IdCapaRef[i][j] representa el identificador de capa de la j-ésima capa de referencia directa para la capa con índice i de capa.

10 En una realización particular, las variables NumCapasRefDirect[i] e IdCapaRef[i][j] se derivan en el paso S11 según:

```
para(i = 1; i ≤ capas_max_vps_menos1; i++)
    para(j=0; NumCapasRefDirect[i] = 0; j < i; j++)
        si(bandera_dependencia_directa[i][j] == 1)
            IdCapaRef[i][ NumCapasRefDirect[i]++] = nuh_en_id_capa[j]
```

15 Si esta realización se usa en relación con el método de decodificación que se muestra en la Fig. 5 las banderas de referencia directa y la información de correspondencia se podría señalar junta, tal como en una extensión del VPS como se muestra a continuación. Esta realización entonces tiene preferiblemente una relación de capa jerárquica ya que una capa con índice i de capa preferiblemente puede ser sólo una capa de referencia para las capas con índice j > i de capa.

```
20 vps_extension() {
    ...
    para(i = 1; i ≤ capas_max_vps_menos1; i++){
        nuh_en_id_capa[i]
        Descriptor
        u(6)
    }
25 ...
    para(i = 1; i ≤ capas_max_vps_menos1; i++){
        para(j=i+1; j ≤ capas_max_vps_menos1; j++)
            bandera_referencia_directa[i][j]
            u(1)
    }
30 }
```

Un aspecto de las realizaciones relacionado con un método de determinación de la relación de decodificación para un flujo 2 de video multicapa codificado que define múltiples capas 10, 12, 14 de las imágenes 20, 22, 24. El método comprende de manera opcional calcular una longitud de un elemento de sintaxis numérico. El método también comprende recuperar, basado en el flujo 2 de video multicapa codificado, el número de elementos de sintaxis que define un número de capas de referencia de las múltiples capas 10, 12, 14 para una capa 12 con un índice i de capa de las múltiples capas 10, 12, 14, basado opcionalmente en la longitud del elemento de sintaxis numérico. El método además comprende recuperar, basado en el flujo 2 de video multicapa codificado y para cada número de capas de referencia, un respectivo elemento de sintaxis de referencia que define un índice j de capa de una capa 10 de las múltiples capas 10, 12, 14 del que la capa 12 con el índice i de capa depende, j < i.

40 La Fig. 7 es un diagrama de flujo que ilustra una realización de este método de determinación de la relación de decodificación para un flujo de video multicapa codificado que define múltiples capas de imágenes. El método comienza en un paso S20 opcional, que comprende el cálculo de un elemento de sintaxis numérico. Este elemento de sintaxis numérico define preferiblemente un número de capas de referencia para una capa actual con índice i de capa.

45 En una realización particular la longitud del elemento de sintaxis numérico en términos de número de bits es variable y depende de al menos un parámetro de entrada o un elemento de sintaxis como se discute en detalle más adelante. En tal caso, la longitud del elemento de sintaxis numérico se podría calcular en el paso S20 para determinar cuántos bits abarca el elemento de sintaxis numérico. Es posible que el elemento de sintaxis numérico tenga una longitud común para múltiples capas, así como para todas las capas, en el flujo de video multicapa. En tal

caso, la longitud del elemento de sintaxis numérico se podría calcular una vez en el paso S20 para una primera capa. Este valor de longitud calculado se podría entonces almacenar y reutilizar para las otras capas en el flujo de video multicapa, relajando de este modo la necesidad de realizar el paso S20 para cada capa en el flujo de video multicapa.

5 El paso S21 de la Fig. 7 comprende la recuperación del elemento de sintaxis numérico basado en el flujo de video multicapa codificado. El elemento de sintaxis numérico se podría recuperar de un conjunto de parámetros, de una extensión del conjunto de parámetros, de cualquier otra estructura de datos de o asociada con el flujo de video multicapa codificado o a partir de una representación codificada del fracción en el flujo de video multicapa codificado.

10 En una realización, el paso S21 comprende la recuperación del elemento de sintaxis numérico basado en la longitud del elemento de sintaxis numérico según se calculó en el paso opcional S20. Así, la información de la longitud del elemento de sintaxis numérico se emplea en el paso S21 para identificar la parte, esto es el número de bits, en la estructura relevante que corresponde con el elemento de sintaxis numérico.

15 En un enfoque alternativo, la recuperación del elemento de sintaxis numérico se realiza en el paso S21 sin usar ninguna longitud calculada del elemento de sintaxis numérico. Por ejemplo, el elemento de sintaxis numérico se podría identificar basado en el análisis y la decodificación de la estructura de datos, así como comenzando desde el inicio de la estructura de datos.

20 El método entonces continúa al paso S22 donde se recupera un respectivo elemento de sintaxis de referencia basado en el flujo de video multicapa codificado y para cada una del número de capas de referencia que se define por el elemento de sintaxis numérica recuperado en el paso S21. El elemento o los elementos de sintaxis de referencia recuperados en el paso S22 se podrían recuperar de un conjunto de parámetros, de una extensión del conjunto de parámetros, de cualquier otra estructura de datos de o asociada con el flujo de video multicapa codificado o a partir de una representación codificada de una fracción en el flujo de video multicapa codificado. En una realización particular, el elemento o los elementos de sintaxis de referencia están presentes preferiblemente en la misma estructura de datos que el elemento de sintaxis numérico.

25 Un elemento de sintaxis de referencia define un índice j de capa de una capa en el flujo de video multicapa de la que la capa actual con índice i de capa depende directamente. Además, en una realización preferida se induce una relación de capa jerárquica tal que $j < i$.

30 El elemento de sintaxis numérico, así, señala de cuantas capas de referencia depende directamente la capa actual con índice i de capa. Este elemento de sintaxis numérico se emplea por lo tanto en el paso S22 para saber cuántos elementos de sintaxis de referencia recuperar para la capa con índice i de capa. Cada uno de dichos elementos de sintaxis de referencia recuperados en el paso S22 define un respectivo índice j de capa para una capa de la que la capa actual con índice i de capa depende directamente.

El método entonces termina en la realización.

35 En una realización alternativa los elementos de sintaxis de referencia recuperados en el paso S22 se usan como la información de relación de decodificación en el método de decodificación de un flujo de video multicapa codificado descrito en relación con la Fig. 5. En dicho caso, el método continúa desde el paso S22 de la Fig. 7 al paso S2 de la Fig. 5.

40 En una primera realización particular el paso opcional S20 comprende el cálculo de la longitud del elemento de sintaxis numérico basado en el índice i de capa. Así, en esta realización la longitud del elemento de sintaxis numérico se calcula basada en el valor particular del índice i de capa.

En una primera implementación de ejemplo el paso S20 comprende el cálculo de la longitud del elemento de sintaxis numérico como $\text{ceil}(\log_2(i+1))$, en donde la i representa el índice de capa, $\text{ceil}(x)$ representa el menor número entero mayor o igual que x y $\log_2(x)$ representa un logaritmo en base 2 de x .

45 En consecuencia, en esta implementación de ejemplo el número de bits usados para representar el número de capas de referencia, esto es el elemento de sintaxis numérico, está representado por un código $u(v)$ con $v = \text{ceil}(\log_2(i+1))$.

En una segunda realización particular el paso S20 opcional comprende el cálculo de la longitud del elemento de sintaxis numérico basado en la información del número máximo posible de capas en el flujo de video multicapa codificado.

50 En una segunda implementación de ejemplo el paso S20 comprende el cálculo de la longitud del elemento de sintaxis numérica como $\text{ceil}(\log_2(\text{capas_max_vps_menos1} + 1))$, en donde $\text{capas_max_vps_menos1} + 1$ indica, como se indicó anteriormente en la presente memoria, el número máximo de capas en el flujo de video multicapa.

Esta segunda implementación de ejemplo sería de eficiencia ligeramente menor que la primera implementación de ejemplo. Sin embargo, una ventaja es que la longitud del código del elemento de sintaxis numérico es independiente

de la capa concreta (índice de capa), lo que puede llevar a una decodificación menos compleja de la estructura de sintaxis.

5 En una tercera realización particular el paso S20 opcional comprende el cálculo de la longitud del elemento de sintaxis numérico basado en la información del valor máximo de identificador de capa en el flujo de video multicapa codificado.

En una tercera implementación de ejemplo el paso S20 comprende el cálculo de la longitud del elemento de sintaxis numérico como $\text{ceil}(\log_2(\text{id_capa_max_vps} - 1))$, en donde id_capa_max_vps indica el máximo valor posible para los identificadores de capa (id_capa) en el flujo de video multicapa codificado.

10 En una realización opcional el método como se muestra en la Fig. 7 comprende un paso adicional S30, véase la Fig. 8. En tal caso el método continúa desde el paso S21 de la Fig. 7. El paso S30 comprende el cálculo de la longitud respectiva de los respectivos elementos de sintaxis de referencia para ser recuperados en el paso S22 de la Fig. 7.

En una realización, el paso S30 comprende el cálculo de la longitud respectiva del respectivo elemento de sintaxis de referencia basado en el índice i de capa, tal como basado en el valor concreto de este índice i de capa.

15 En un primer ejemplo de implementación el número de bits usados para representar el elemento o los elementos de sintaxis de referencia es igual a $\text{ceil}(\log_2(i))$.

En tal caso, los elementos de sintaxis para señalar el elemento de sintaxis numérico y los elementos de sintaxis de referencia se podrían definir como a continuación.

```

para(i = 1; i ≤ capas_max_vps_menos1; i++){           Descriptor
    num_capas_ref_directa[i]                          u(v)
20     para(j=0; j < num_capas_ref_directa[i];j++)
        indice_capa_ref[i][j]                        u(v)
}

```

25 $\text{num_capas_ref_directa}[i]$ especifica el número de capas de referencia de las que la capa con índice i de capa depende. El número de bits usados para representar $\text{num_capas_ref_directa}[i]$ podría ser $v = \text{ceil}(\log_2(i + 1))$ o $v = \text{ceil}(\log_2(\text{capas_max_vps_menos1} + 1))$ o $v = \text{ceil}(\log_2(\text{cps_max_capa_id} - 1))$. $\text{indice_capa_ref}[i][j]$ identifica el índice de capa de la capa j -ésima de la que la capa con índice i de capa depende. El número de bits usados para representar el $\text{indice_capa_ref}[i][j]$ es preferiblemente igual a $v = \text{ceil}(\log_2(i))$.

30 Si esta realización se usa en relación con el método de decodificación que se muestra en la Fig. 5 los elementos de sintaxis numéricos, los elementos de sintaxis de referencia y la información de correspondencia se podrían señalar juntos, tal como en una extensión del VPS como se muestra más adelante. Entonces esta realización tiene preferiblemente una relación de capa jerárquica ya que una capa con índice i de capa preferiblemente sólo puede ser una capa de referencia para capas con índice $j > i$ de capa.

```

vps_extension() {                                     Descriptor
...
35     para(i = 1; i ≤ capas_max_vps_menos1; i++){
        nuh_en_id_capa[i]                             u(6)
    }
...
    para(i = 1; i ≤ capas_max_vps_menos1; i++){
40     num_capas_ref_directa[i]                       u(v)
        para(j=0; j < num_capas_ref_directa[i]; j++)
            indice_capa_ref[i][j]                     u(v)
    }
}

```

En un segundo ejemplo de implementación se hace la suposición de que los índices de referencia, esto es los elementos de sintaxis de referencia, se indican en un orden jerárquico de índices de capa para las capas de referencia. Por ejemplo, se puede suponer que $\text{indice_capa_ref}[i][j] > \text{indice_capa_ref}[i][j-1]$ para $j > 0$, en donde $\text{indice_capa_ref}[i][j]$ representa el índice de capa de la capa j -ésima de la que la capa con índice i de capa depende. Entonces el índice j -ésimo de capa de referencia para la capa con índice i de capa se puede representar como un índice de capa delta, $\text{indice_capa_ref_delta}[i][j]$, tal que $\text{indice_capa_ref}[i][j] = \text{indice_capa_ref}[i][j-1] + 1 + \text{indice_capa_ref_delta}[i][j]$. En tal caso, el $\text{indice_capa_ref}[i][-1] = -1$ y el $\text{indice_capa_ref_delta}[i][j] \geq 0$. Con una relación de capa jerárquica $\text{indice_capa_ref}[i][j] < i$ y, así, $0 \leq \text{indice_capa_ref_delta}[i][j] < i - \text{indice_capa_ref}[i][j-1] - 1$. Esto a veces implica que el $\text{indice_capa_ref_delta}[i][j]$ puede ser representado por un código $u(v)$ con $v = \text{ceil}(\log_2(i - \text{indice_capa_ref}[i][j-1] - 1))$.

En tal caso, los elementos de sintaxis para señalar el elemento de sintaxis numérico y los elementos de sintaxis de referencia podrían ser como se definen a continuación.

```

para(i = 1; i ≤ capas_max_vps_menos1; i++){           Descriptor
    num_capas_ref_directa[i]                          u(v)
15    para(j=0; j < num_capas_ref_directa[i]; j++)
        indice_capa_ref_delta[i][j]                  u(v)
}

```

Un aspecto de las realizaciones se relaciona con un método de determinación de la relación de codificación para un flujo 1 de video multicapa que define múltiples capas 10, 12, 14 de imágenes 20, 22, 24. El método comprende determinar cualquier relación de codificación entre las múltiples capas 10, 12, 14. El método también comprende determinar, para una capa 12 con índice i de capa de las múltiples capas 10, 12, 14 y basada en la relación de codificación, al menos una bandera de decodificación directa que indique una relación de codificación directa entre la capa 12 con índice i de capa y una capa 10 con índice j de capa de las múltiples capas 10, 12, 14, $i \neq j$. El método comprende además asociar la al menos una bandera de decodificación directa con una representación 2 codificada del flujo 1 de video multicapa.

La Fig. 10 es un diagrama de flujo que ilustra una realización de este método de determinación de la relación de codificación para un flujo de video multicapa que define múltiples capas de imágenes. El método generalmente comienza en el paso S50, que comprende determinar cualquier relación de codificación entre las múltiples capas en el flujo de video multicapa. La relación de codificación se determina preferiblemente basada en la información de a qué capa pertenecen las imágenes de referencia usadas para codificar las imágenes en una capa actual. Así, mediante la determinación de en qué capa o capas se usan las imágenes de referencia como base de codificación para las imágenes en la capa actual se puede determinar cualquier relación de codificación entre las diversas capas en el flujo de video multicapa en el paso S50.

La relación de codificación determinada para el flujo de video multicapa en el paso S50 se usa después en el paso S51 para determinar, para una capa con índice i de capa, al menos una bandera de decodificación directa que indique una relación de codificación directa entre la capa con índice i de capa y una capa con índice j de capa, donde i es diferente de j . Esto significa que la relación de codificación se emplea para identificar cualquier relación de codificación directa, esto es dependencia directa, entre la capa actual con índice i de capa y cualquier otra capa en el flujo de video multicapa. Dicha dependencia directa entre capas es indicada después por la bandera o banderas de decodificación directa determinadas en el paso S51.

El método también comprende asociar, en el paso S52, la al menos una bandera de decodificación directa con una representación codificada del flujo de video multicapa. Esta asociación en el paso S52 preferiblemente, como se ha descrito anteriormente en la presente memoria, implica incluir la bandera o banderas de decodificación directas en un conjunto de parámetros, en una extensión del conjunto de parámetros, en otras estructuras de datos de o asociadas con el flujo de video multicapa codificado o en al menos una representación codificada de una fracción en el flujo de video multicapa codificado.

El método entonces termina.

El método como se describe en la Fig. 10 se podría implementar como un método independiente de determinación de la relación de codificación para un flujo de video multicapa. En un enfoque alternativo, el método de la Fig. 10 se usa junto con el método de codificación de un flujo de video multicapa como se muestra en la Fig. 9. En tal caso, los pasos S50 y S51 de la Fig. 10 se realizan como una realización particular de la determinación de la información de relación de codificación de la Fig. 9. En tal caso, el método preferiblemente comienza en el paso S40 de la Fig. 9 y continúa a los pasos S50 y S51 de la Fig. 10 y después vuelve al paso S42 de la Fig. 9. El paso S43 de la Fig. 9 se realiza preferiblemente después como se muestra en el paso S52 de la Fig. 10.

Un aspecto de las realizaciones se relaciona a un método de determinación de la relación de codificación para un flujo 1 de video multicapa que define múltiples capas 10, 12, 14 de imágenes 20, 22, 24. El método comprende determinar un elemento de sintaxis numérico que define un número de capas de referencia de las múltiples capas 10, 12, 14 para una capa 12 con índice i de capa de las múltiples capas 10, 12, 14. El método también comprende
 5 determinar, para cada uno de los números de capas de referencia, un respectivo elemento de sintaxis de referencia que define un índice j de capa de una capa 10 de las múltiples capas 10, 12, 14 de la que la capa 12 con índice i de capa depende directamente, $j < i$. El método además comprende asociar el elemento de sintaxis numérico y los respectivos elementos de sintaxis de referencia con una representación codificada 2 del flujo 1 de video multicapa.

La Fig. 11 es un diagrama de flujo que ilustra una realización de este método de determinación de la relación de codificación para un flujo de video multicapa. El método comienza en el paso S60 donde un elemento de sintaxis numérico se determina para una capa con índice i de capa en el flujo de video multicapa. Este elemento de sintaxis numérico representa el número de capas de referencia para la capa con índice i de capa.

El elemento de sintaxis numérico se determina preferiblemente basado en la información de qué imágenes de referencia de capas usadas para codificar las imágenes pertenecen a la capa actual con índice i de capa. Así, mediante la determinación de en qué capa o capas las imágenes de referencia usadas como base de codificación para las imágenes en la capa actual el número de dichas capas de referencia se puede definir y usar para determinar el elemento de sintaxis numérico en el paso S60.

El método de la Fig. 11 también comprende determinar, en el paso S61, un respectivo elemento de sintaxis de referencia para cada uno de los números de las capas de referencia. Así, si el elemento de sintaxis numérico define X capas de referencia entonces se determinan preferiblemente X elementos de sintaxis de referencia en el paso S61. Un elemento de sintaxis de referencia como se determina en el paso S61 define un índice j de capa de una capa en el flujo de video multicapa de la que la capa con índice i de capa depende directamente. En una realización particular, se usa una dependencia de capa jerárquica tal que $j < i$.

El elemento de sintaxis numérico determinador en el paso S60 y los elementos de sintaxis de referencia determinados en el paso S61 son entonces asociados con una representación codificada del flujo de video multicapa en el paso S62. Esta asociación en el paso S62 preferiblemente, como se ha descrito anteriormente en la presente memoria, involucra incluir los elementos de sintaxis numérico y de referencia en un conjunto de parámetros, en una extensión del conjunto de parámetros, en otra estructura de datos de o asociada con el flujo de video multicapa o en al menos una representación codificada de una fracción en el flujo de video multicapa codificado.

El método entonces termina.

El método como se describe en la Fig. 11 se podría implementar como un método independiente de determinación de la relación de codificación para un flujo de video multicapa. En un enfoque alternativo, el método de la Fig. 11 se usa junto con el método de codificación de un flujo de video multicapa como se muestra en la Fig. 9. En tal caso, los pasos S60 y S61 de la Fig. 11 se realizan como una realización particular de determinación de la información de la relación de codificación en la Fig. 9. En tal caso, el método comienza preferiblemente en el paso S40 de la Fig. 9 y continúa en los pasos S60 y S61 de la Fig. 11 y después vuelve al paso S42 de la Fig. 9. El paso S43 de la Fig. 9 se realiza preferiblemente después como se muestra en el paso S62 de la Fig. 11.

Se han realizado experimentos de simulación para determinar el número de bits requeridos para señalar las dependencias de capas. Se han comparado tres diferentes métodos de señalización de dichas dependencias de capa: la señalización de dependencia de capa de la técnica anterior descrita en el documento anteriormente mencionado JCTVC-K1007 (referido como K1007 en las Fig. 20-22), la señalización de dependencia de capa como se describe en la presente memoria en la cual `num_capas_ref_directa[i]` e `id_capa_red[i][j]` están en código $u(v)$ en lugar de código $u(6)$ (referido como $u(v)$ en las Fig. 20-22) y la señalización de dependencia de capa que usa la bandera `referencia_directa[i][j]` como se describe en la presente memoria (referido como `bandera_dependencia` en las Fig. 20-22).

La Fig. 20 ilustra los resultados de la simulación cuando cada capa con índice $i \geq 1$ de capa tiene una capa de referencia. La Fig. 21 ilustra los resultados de la simulación cuando cada capa con índice $i \geq 2$ de capa tiene dos capas de referencia y la capa con índice $i = 1$ de capa tiene una capa de referencia, esto es el número máximo de capas de referencia que es posible para esta capa con índice $i = 1$ de capa. La Fig. 22 ilustra los resultados de simulación cuando cada capa con índice $i \geq 1$ tiene el número máximo de capas de referencia que es posible, esto es una capa con índice $i = N$ de capa tiene N capas de referencia. Los resultados en las Fig. 20-22 se representan para 1... 64 capas, con 64 correspondiendo al máximo número de capas permitido.

Los resultados en las Fig. 20-22 muestran que el método $u(v)$ es siempre inferior que el método K1007. Hasta un cierto umbral para el número de capas (8 en la Fig. 20, 21 en la Fig. 21 y 64 en la Fig. 22) el método de la bandera `dependencia` tiene el menor número de bits. El número de bits para el método de la bandera `dependencia` es independiente de la configuración de dependencia de la capa.

Un aspecto de las realizaciones define un decodificador configurado para decodificar un flujo 2 de video multicapa codificado que define múltiples capas 10, 12, 14 de imágenes 20, 22, 24, teniendo cada capa 10, 12, 14 de las

múltiples capas 10, 12, 14 un respectivo identificador de capa. El decodificador comprende un recuperador de información de relación de decodificación configurado para recuperar, para una capa 12 con un índice de capa de las múltiples capas 10, 12, 14, la información de relación de decodificación basada en el flujo 2 de video multicapa. La información de relación de decodificación que define un respectivo índice de capa de cualquier capa 10 de referencia de las múltiples capas 10, 12, 14, de las que la capa 12 depende directamente. El decodificador también comprende una unidad de correspondencia de índice a identificador configurada para hacer corresponder, para cada capa 10 de referencia y para la capa 12, su índice de capa a un identificador de capa basado en la información de correspondencia de la relación de correspondencia jerárquica entre los identificadores de capa y los índices de capa. La información de correspondencia se recupera basada en el flujo 2 de video multicapa codificado. El decodificador además comprende una unidad de decodificación configurada para decodificar una imagen 22 de la capa 12 basada en al menos una imagen 20 anteriormente decodificada en una capa 10 de las múltiples capas 10, 12, 14 identificada en base a los identificadores de capa hechos corresponder de los índices de capa.

La Fig. 12 es un diagrama de bloques esquemático que ilustra una realización de implementación de un decodificador 100 configurado para decodificar un flujo de video multicapa en el que cada capa tiene un respectivo identificador de capa. El decodificador 100 comprende un recuperador 110 de información de relación de decodificación, también referido como unidad o medio de recuperación de información de relación de decodificación. El recuperador 110 de información de relación de decodificación se configura para recuperar la información de relación de decodificación basada en el flujo de video multicapa codificado para una capa actual que tiene un índice de capa. El recuperador 110 de información de relación de decodificación se configura preferiblemente para recuperar la información como anteriormente se describió en la presente memoria desde una estructura de datos de o asociada con el flujo de video multicapa codificado, tal como un conjunto de parámetros, una extensión del conjunto de parámetros, otras estructuras de datos o una representación codificada de un fracción.

La información de relación de decodificación según es recuperada por el recuperador 110 de información de relación de decodificación define un respectivo índice de capa de cualquier capa de referencia en el flujo de video multicapa, de la que la capa actual depende directamente.

Una unidad 120 de correspondencia de índice a identificador, también designada mapeador o medio de correspondencia de índice a identificador, se configura para hacer corresponder un índice de capa a un identificador de capa para cada capa de referencia y para la capa actual basado en la información de correspondencia de la relación de correspondencia jerárquica entre los identificadores de capa y los índices. La unidad 120 de correspondencia de índice a identificador se configura para recuperar la información de correspondencia basada en el flujo de video multicapa codificado, preferiblemente de la misma estructura de datos de la que el recuperador 110 de información de relación de decodificación recupera la información de relación de decodificación.

Los identificadores de capa según se obtienen de la unidad 120 de correspondencia de índice a identificador son usados por una unidad 130 de decodificación, también referida como decodificador o medio de decodificación de imagen, cuando decodifica una imagen de la capa actual basada en al menos una imagen previamente decodificada en una capa identificada en base a los identificadores de capa.

En una realización, la información de relación de decodificación es una entrada a la unidad 120 de correspondencia de índice a identificador, que luego hace corresponder el índice/índices de capa de la capa/capas de las que la capa actual depende directamente. Así, en esta realización sólo dicho índice/índices se hacen corresponder junto con el índice de capa de la capa actual en los respectivos identificadores de capa.

En una realización alternativa, la unidad 120 de correspondencia índice a identificador se configura para hacer corresponder todos los índices de capa que se señalan en o asociados con el flujo de video multicapa codificado en los respectivos identificadores de capa. En tal caso, la unidad 130 de decodificación identifica preferiblemente aquellos identificadores de capa de entre los identificadores de capa hechos corresponder que son de relevancia para decodificar las imágenes en la capa actual en base a la información de relación de decodificación del recuperador 110 de información de relación de decodificación.

Así, de este modo la información de relación de decodificación y la información de correspondencia juntas permiten la identificación y determinación de los identificadores de capa de cualquier capa de referencia que comprenda las imágenes basadas en las que las imágenes de la capa actual deberían ser decodificadas.

En una realización la unidad 120 de correspondencia de índice a identificador se configura para recuperar una bandera `bandera_presente_id_capa_nuh_vps` basada en el flujo de video multicapa codificado. La unidad 120 de correspondencia de índice a identificador se configura también para fijar, para cada capa de referencia y para la capa y si `bandera_presente_id_capa_nuh_vps = 0`, su identificador de capa igual a su índice de capa. La unidad 120 de correspondencia de índice a identificador se configura además para recuperar, para cada capa de referencia y para la capa y si `bandera_presente_id_capa_nuh_vps = 1`, su identificador de capa de un vector `nuh_en_id_capa[i]`, $i \in [1, \text{capas_max_vps_menos1}]$, en donde `capas_max_vps_menos1 + 1` indica el número máximo de capas y `nuh_en_id_capa[i]` indica un identificador de capa para una capa con índice i de capa.

Un aspecto de las realizaciones define un codificador configurado para codificar un flujo 1 de video multicapa que define múltiples capas 10, 12, 14 de imágenes 20, 22, 24, teniendo cada capa 10, 12, 14 de las múltiples capas 10, 12, 14 un respectivo identificador de capa. El codificador comprende una unidad de correspondencia configurada para hacer corresponder de manera jerárquica, para cada capa 10, 12, 14 de las múltiples capas 10, 12, 14, un identificador de capa de la capa 10, 12, 14 a un índice de capa basada en las dependencias de codificación entre las múltiples capas 10, 12, 14. El codificador también comprende un determinador de información configurado para determinar la información de relación de codificación que define un respectivo índice de capa de cualquier capa 10 de referencia de las múltiples capas 10, 12, 14, de las que una capa 12 de las múltiples capas 10, 12, 14 depende directamente. El codificador comprende además una unidad de codificación configurada para generar un flujo 2 de video multicapa codificado mediante la codificación de las imágenes 20, 22, 24 de las múltiples capas 10, 12, 14 basadas en las dependencias de codificación y una unidad de asociación configurada para asociar la información de relación de codificación con el flujo 2 de video multicapa codificado.

La Fig. 15 es un diagrama de bloques esquemático de una realización de una codificador 400 configurado para codificar un flujo de video multicapa en el que cada capa tiene un respectivo identificador de capa. El codificador 400 comprende una unidad 410 de correspondencia, también referida como un mapeador o medio de correspondencia. La unidad 410 de correspondencia se configura para hacer corresponder de manera jerárquica un identificador de capa para cada capa del flujo de video multicapa a un índice de capa basado en las dependencias de codificación entre las múltiples capas en el flujo de video multicapa. Esto significa que la información sobre qué capa o capas que son capas de referencia para una u otras capas es usada por la unidad 410 de correspondencia para asignar índices de capa a las capas.

El codificador 400 también comprende un determinador 420 de información, también designado como unidad o medio de determinación de información. El determinador 420 de información se configura para determinar la información de relación de codificación que define un respectivo índice de capa de cualquier capa de referencia basada en la cual la capa actual depende directamente.

Una unidad 430 de codificación, también referida como codificador de imagen o medio de codificación, se configura para generar un flujo de video multicapa mediante la codificación de imágenes en las múltiples capas en base a las dependencias de codificación. Una unidad 440 de asociación, también referida como asociador o medio de asociación, se configura para asociar la información de relación de codificación con el flujo de video multicapa codificado, como se describió anteriormente en la presente memoria, tal como en un conjunto de parámetros, en una extensión del conjunto de parámetros, en otras estructuras de datos o en una representación codificada de un fracción.

La unidad 440 de asociación se configura también preferiblemente para asociar la información de correspondencia que define una relación de correspondencia jerárquica entre los identificadores de capa y los índices de capa con el flujo de video multicapa codificado.

Otros aspectos de las realizaciones se relacionan con los dispositivos para determinar las relaciones de decodificación o codificación para un flujo de video multicapa (codificado). Dichos dispositivos se podrían proporcionar como entidades separadas o se podrían implementar como parte del decodificador 100 de la Fig. 12 o el codificador 400 de la Fig. 15.

Un aspecto de las realizaciones define un dispositivo para determinar la relación de decodificación para un flujo 2 de video multicapa codificado que define múltiples capas 10, 12, 14, de imágenes 20, 22, 24. El dispositivo comprende un recuperador de bandera configurado para recuperar, basado en el flujo 2 de video multicapa, al menos una bandera de decodificación directa que indica una relación de codificación directa entre una capa 12 con índice i de capa de las múltiples capas 10, 12, 14 y una capa 10 con índice j de capa de las múltiples capas 10, 12, 14, $i \neq j$. El dispositivo también comprende un determinador de información configurado para determinar la información que define cualquier relación de decodificación entre las múltiples capas 10, 12, 14 basada en al menos una bandera de decodificación directa.

La Fig. 13 es una ilustración de una realización de dicho dispositivo 200 para determinar la relación de decodificación. El dispositivo 200 comprende un recuperador 210 de bandera, también referido como unidad o medio de recuperación de bandera. El recuperador 210 de bandera se configura para recuperar al menos una bandera de decodificación directa basada en el flujo de video multicapa codificado, tal como de un conjunto de parámetros, de una extensión del conjunto de parámetros, de otras estructuras de datos de una representación codificada de una fracción en un flujo de video multicapa codificado. La al menos una bandera de decodificación directa recuperada indica una relación de codificación directa entre una capa actual con índice i de capa y una capa con índice j de capa del flujo de video multicapa, donde el índice i es diferente del índice j .

El dispositivo 200 también comprende un determinador 220 de información, también referido como unidad o medio de determinación. El determinador 220 de información se configura para determinar la información que define cualquier relación de decodificación entre las múltiples capas en el flujo de video multicapa basado en la al menos una bandera de decodificación directa recuperada por el recuperador 210 de bandera.

5 En una realización el recuperador 210 de bandera se configura para recuperar, para la capa con índice i de capa, una respectiva bandera de dependencia directa para cada índice j de capa basada en el flujo de video multicapa codificado. En esta realización, $j < i$ y la bandera de dependencia directa indica si la capa con índice j de capa es una capa de referencia directa para la capa con índice i de capa. El determinador 220 de información se configura preferiblemente para determinar la información que define cualquier capa con índice $j < i$ de capa, de la que la capa con índice i de capa depende, basado en las banderas de dependencia directa.

10 En un ejemplo de implementación el recuperador 210 de bandera se configura para recuperar, para la capa con índice i de capa, una respectiva bandera de dependencia directa $\text{bandera_dependencia_directa}[i][j]$ para cada índice j de capa basado en el flujo de video multicapa codificado. El determinador 220 de información se configura preferiblemente para determinar un conjunto de banderas de dependencia $\text{bandera_dependencia}[i]$ como $\text{bandera_dependencia}[i] = \text{bandera_dependencia_directa}[i] + \sum_{j=0..j-1} (\text{bandera_dependencia_directa}[i][j] \times \text{bandera_dependencia}[j])$. El conjunto de banderas de dependencia o las banderas de dependencia $\text{bandera_dependencia}[i][j]$ entonces constituyen en este ejemplo de implementación la información que define cualquier relación de decodificación entre las múltiples capas.

15 En otra realización el recuperador 210 de bandera se configura para recuperar, para la capa con índice i de capa, una respectiva bandera de referencia directa para cada índice j de capa basada en el flujo de video multicapa codificado. En esta realización, $j > i$ y la bandera de referencia directa indica si la capa con índice i de capa es una capa de referencia directa para la capa con índice j de capa. El determinador 220 de información, en esta realización, se configura preferiblemente para determinar la información que define cualquier capa con índice $j > i$ de capa para la que la capa con índice i de capa es una capa de referencia basada en las banderas de referencia directa.

20 En otro ejemplo de implementación el recuperador 210 de bandera se configura para recuperar, para la capa con índice i de capa, una respectiva bandera de referencia $\text{bandera_referencia_directa}[i][j]$ para cada índice j de capa basado en el flujo de video multicapa codificado. El determinador 220 de información, en este ejemplo, se configura para determinar un conjunto de banderas de referencia $\text{bandera_referencia}[i]$ como $\text{bandera_referencia}[i] = \text{bandera_referencia_directa}[i] + \sum_{j=1..i_{\max}} (\text{bandera_referencia_directa}[i][j] \times \text{bandera_referencia}[j])$. El conjunto de banderas de referencia o las banderas de referencia $\text{bandera_referencia}[i][j]$ entonces constituyen en este ejemplo de implementación la información que define cualquier relación de decodificación entre las múltiples capas.

25 El dispositivo para determinar la relación 200 de decodificación en la Fig. 13 está en una realización de un ejemplo de implementación particular del recuperador 110 de información de relación de decodificación de la Fig. 12.

30 Otro aspecto de las realizaciones define un dispositivo para determinar la relación de decodificación para un flujo 2 de video multicapa codificado que define múltiples capas 10, 12, 14 de imágenes 20, 22, 24. El dispositivo comprende de manera opcional una calculadora de longitud configurada para calcular la longitud de un elemento de sintaxis numérico. El dispositivo comprende un elemento recuperador configurado para recuperar, basado en el flujo
35 2 de video multicapa codificado, el elemento de sintaxis numérico que define un número de capas 10 de referencia de las múltiples capas 10, 12, 14 para una capa 12 con índice i de capa de las múltiples capas 10, 12, 14, basada opcionalmente en la longitud del elemento de sintaxis numérico. El recuperador de elemento se configura además para recuperar, basado en el flujo 2 de video multicapa codificado y para cada uno de los números de las capas de referencia, un respectivo elemento de sintaxis de referencia que define un índice j de capa de una capa 10 de las
40 múltiples capas 10, 12, 14 de la que la capa 12 con índice i de capa depende directamente, $j < i$.

45 La Fig. 14 es un diagrama de bloques esquemático de dicho dispositivo 300 para determinar la relación de decodificación. El dispositivo 300 comprende de manera opcional una calculadora 310 de longitud, también designada como unidad o medio de cálculo de longitud. La calculadora 310 de longitud se configura preferiblemente para calcular la longitud de un elemento de sintaxis numérico que define un número de capa de referencia en el flujo de video multicapa para una capa actual con índice i de capa. Un recuperador 320 de elemento, también designado como unidad o medio de recuperación de elemento, del dispositivo 300 se configura para recuperar el elemento de sintaxis numérico basado en el flujo de video multicapa codificado, tal como de un conjunto de parámetros, de una extensión del conjunto de parámetros, de otras estructuras de datos o de una representación codificada de una fracción. En una realización particular, el recuperador 320 de elemento usa la información de la longitud del
50 elemento de sintaxis numérico, preferiblemente según es determinado por la calculadora 310 de longitud, para identificar la parte relevante de la estructura de datos que lleva el elemento de sintaxis numérico que corresponde al elemento de sintaxis numérico.

55 El valor de elemento de sintaxis numérico recuperado es también usado por el recuperador 320 de elemento para determinar cuántos elementos de sintaxis de referencia recuperar para la capa actual con índice i de capa. Así, el recuperador 320 de elemento recupera un respectivo elemento de sintaxis de referencia para cada uno de los números de las capas de referencia según se define por el elemento de sintaxis numérico. El recuperador 320 de elemento recupera preferiblemente el elemento de sintaxis de referencia de un conjunto de parámetros, de una extensión de un conjunto de parámetros, de otras estructuras de datos o de la representación codificada de una fracción. En una realización particular, el recuperador 320 de elemento se configura para recuperar el elemento de
60 sintaxis de referencia de la misma estructura de datos que también comprende el elemento de sintaxis numérico.

Cada elemento de sintaxis de referencia recuperado por el recuperador 320 de elemento define un índice j de capa de una capa de referencia de la que la capa actual con índice i de capa depende directamente. En una realización particular, se usa una dependencia de capa jerárquica para que $j < i$.

5 En una realización la calculadora 310 de longitud se configura para calcular la longitud del elemento de sintaxis numérico basado en el índice i de capa, esto es basado en el valor del índice i de capa.

En una realización particular la calculadora 310 de longitud se configura para calcular la longitud del elemento de sintaxis numérico como $\text{ceil}(\log_2(i+1))$.

En otra realización la calculadora 310 de longitud se configura para calcular la longitud del elemento de sintaxis numérico basada en la información de un número máximo posible de capas en el flujo de video multicapa codificado.

10 En otra realización la calculadora 310 de longitud se configura para calcular la longitud del elemento de sintaxis numérico como $\text{ceil}(\log_2(\text{capas_max_vps_menos1} + 1))$.

En una realización adicional la calculadora 310 de longitud se configura para calcular la longitud del elemento de sintaxis numérico basada en la información de un valor máximo para el identificador de capa en el flujo de video multicapa codificado.

15 En una realización adicional particular la calculadora 310 de longitud se configura para calcular la longitud del elemento de sintaxis numérico como $\text{ceil}(\log_2(\text{id_capa_max_vps} - 1))$.

En una realización la calculadora 310 de longitud opcional se configura para calcular la respectiva longitud del respectivo elemento de sintaxis de referencia.

20 En una realización particular la calculadora 310 de longitud se configura para calcular la respectiva longitud del respectivo elemento de sintaxis de referencia basado en el índice i de capa.

En otra realización particular la calculadora 310 de longitud se configura para calcular la respectiva longitud del elemento de sintaxis de referencia como $\text{ceil}(\log_2(i - \text{indice_capa_ref}[i][j-1] - 1))$. En tal caso, el recuperador 320 de elemento se configura preferiblemente para recuperar, basado en el flujo de video multicapa codificado y para cada capa j del número de capas de referencia, un respectivo elemento de sintaxis de referencia $\text{indice_capa_ref_delta}[i][j]$, $j < i$. El dispositivo 300 comprende preferiblemente un determinador 330 de índice opcional configurado para determinar, para cada capa j del número de capas de referencia, el índice de capa de la capa j -ésima de la que la capa con índice i de capa depende directamente como $\text{indice_capa_ref}[i][j] = [i][j-1] + 1 + \text{indice_capa_ref_delta}[i][j]$ con $\text{indice_capa_ref}[i][j-1] = -1, 0 < j < i$.

30 El dispositivo para determinar la relación 300 de decodificación en la Fig. 14 está en una realización de un ejemplo de implementación particular del recuperador 110 de información de relación de decodificación de la Fig. 12.

Un aspecto de las realizaciones define un dispositivo para determinar la relación de codificación para un flujo 1 de video multicapa que define múltiples capas 10, 12, 14 de las imágenes 20, 22, 24. El dispositivo comprende un determinador de relación configurado para determinar cualquier relación de codificación entre las múltiples capas 10, 12, 14. El dispositivo también comprende un determinador de bandera configurado para determinar, para una capa 12 con índice i de capa de las múltiples capas 10, 12, 14 y basado en la relación de codificación, al menos una bandera de decodificación directa que indica una relación de codificación directa entre la capa 12 con índice i de capa y una capa 10 con índice j de capa de las múltiples capas 10, 12, 14, $i \neq j$. Una unidad de asociación se configura para asociar la al menos una bandera de decodificación directa con una representación codificada 2 del flujo 1 de video multicapa.

40 La Fig. 16 muestra un diagrama de bloques de una realización de este dispositivo 500 para determinar la relación de codificación para un flujo de video multicapa. El dispositivo 500 comprende un determinador 510 de relación, también referido como unidad o medio de determinación de relación. El determinador 510 de relación se configura para determinar cualquier relación de codificación entre las múltiples capas en el flujo de video multicapa. El determinador 510 de relación realiza preferiblemente esta determinación como se describió anteriormente en la presente memoria en base a la información de las capas que se usan como capas de referencia para otras capas en el flujo de video multicapa. El dispositivo 500 también comprende un determinador 520 de bandera, también referido como unidad o medio de determinación de bandera. El determinador 520 de bandera se configura para determinar al menos una bandera de decodificación directa para una capa actual con índice i de capa y basado en la relación de codificación que se determinó por el determinador 510 de relación. Una bandera de decodificación directa indica una relación de codificación directa entre la capa actual con índice i de capa y una capa con índice $j \neq i$ de capa.

El dispositivo 500 comprende además una unidad 530 de asociación, también referida como asociador o medio de asociación, configurada para asociar la al menos una bandera de decodificación con una representación codificada del flujo de video multicapa, así como la inserción de al menos una bandera de decodificación en un conjunto de parámetros, en una extensión del conjunto de parámetros, en otras estructuras de datos o en una representación codificada de una fracción.

55

El dispositivo para determinar la relación 500 de codificación en la Fig. 16 está en una realización de un ejemplo de implementación particular del determinador 420 de información de la Fig. 15.

Un aspecto de las realizaciones define un dispositivo para determinar la relación de codificación para un flujo 1 de video multicapa que define múltiples capas 10, 12, 14 de las imágenes 20, 22, 24. El dispositivo comprende un
 5 determinador de elemento de sintaxis numérico configurado para determinar un elemento de sintaxis numérico que define un número de capas 10 de referencia de las múltiples capas 10, 12, 14 para una capa 12 con índice i de capa de las múltiples capas 10, 12, 14. El dispositivo también comprende un determinador de elemento de sintaxis de referencia configurado para determinar, para cada uno de los números de las capas 10 de referencia, un respectivo
 10 elemento de sintaxis de referencia que define un índice de capa j de una capa 10 de las múltiples capas 10, 12, 14 de la que la capa 12 con índice i de capa depende directamente, $j < i$. El dispositivo además comprende una unidad de asociación configurada para asociar el elemento de sintaxis numérico y los respectivos elementos de sintaxis de referencia con una representación codificada 2 del flujo 1 de video multicapa.

La Fig. 17 muestra un diagrama de bloques de una realización de este dispositivo 600 para determinar la relación de codificación para un flujo de video multicapa. El dispositivo 600 comprende un determinador 610 de elemento de sintaxis numérico, también designado unidad o medio de determinación de elemento de sintaxis numérico. El
 15 determinador 610 de elemento de sintaxis numérico se configura para determinar un elemento de sintaxis numérico que define un número de capa de referencia para la capa con índice i de capa. El determinador 610 de elemento de sintaxis numérico determina preferiblemente elementos de sintaxis numéricos basados en la información de a qué capas de referencia pertenecen las imágenes de referencia, si pertenecen a alguna, en base a las imágenes en la
 20 capa actual con índice i de capa.

El dispositivo 600 también comprende un determinador 620 de elemento de sintaxis de referencia, a veces referido como unidad o medio de determinación de elemento de sintaxis de referencia. El determinador 620 de elemento de sintaxis de referencia se configura para determinar un respectivo elemento de sintaxis de referencia para cada uno de los números de las capas de referencia según son definidos por el elemento de sintaxis numérico. Un elemento
 25 de sintaxis de referencia según es determinado por el determinador 620 de elemento de sintaxis de referencia define un índice j de capa de una capa de referencia del flujo de video multicapa de la que la capa actual con índice i de capa depende directamente. En una realización particular, se usa una dependencia de capa jerárquica para que $j < i$.

Una unidad 630 de asociación, también designada asociador o medio de asociación, del dispositivo 600 se configura para asociar el elemento de sintaxis numérico y los respectivos elementos de sintaxis de referencia con una
 30 representación codificada del flujo de video multicapa. Los elementos de sintaxis numérico y de referencia se podrían insertar en el mismo o en diferentes conjuntos de parámetros, extensiones de conjuntos de parámetros, otras estructuras de datos o representaciones codificadas de fracciones.

El dispositivo para determinar la relación 600 de codificación en la Fig. 17 está en una realización de un ejemplo de implementación particular del determinador 420 de información de la Fig. 15.

El decodificador 100 en la Fig. 12, los dispositivo 200, 300 para determinar la relación de codificación en las Fig. 13 y
 35 14, el codificador 400 de la Fig. 15 y los dispositivos 500, 600 para determinar la relación de codificación en las Fig. 16 y 17 se pueden implementar en hardware. Existen numerosas variantes de elementos de circuito que se pueden usar y combinar para lograr las funciones de las unidades del decodificador, codificador o dispositivos en las Fig. 12-17. Dichas variantes son abarcadas por las realizaciones. Los ejemplos particulares de implementación en hardware
 40 incluyen la implementación en hardware de un procesador digital de señales (DSP) y de tecnología de circuitos integrados, que incluye tanto los circuitos electrónicos de propósito general como los circuitos de aplicación específica.

De manera alternativa, el decodificador 100 en la Fig. 12, los dispositivos 200, 300 para determinar la relación de decodificación en las Fig. 13 y 14, el codificador 400 de la Fig. 15 y los dispositivos 500, 600 para determinar la
 45 relación de codificación en las Fig. 16 y 17 se pueden implementar al menos parcialmente en software. Dicha implementación se muestra en la Fig. 18. El decodificador, el codificador o el dispositivo 700 de las Fig. 12-17 comprende un procesador 710 configurado para procesar los medios 740 de código de un programa informático 730 almacenado en un medio legible por ordenador, representado por una memoria 720 en la Fig. 18. Los medios 740 de código provocan, cuando se ejecutan en el procesador 710, que el procesador 710 realice las funciones de las
 50 unidades del decodificador, codificador o dispositivos de las Fig. 12-17.

El procesador 710 podría ser un ordenador de propósito general o especialmente adaptado, un procesador o un microprocesador, tal como una unidad central de procesamiento (CPU). El software incluye elementos de código de programas informáticos o partes de código de software que efectúan la operación de al menos el recuperador 110 de información de relación de decodificación, la unidad 120 de correspondencia de índice a identificador y la unidad
 55 130 de decodificación de la Fig. 12, y/o el recuperador 210 de bandera y el determinador 220 de información de la Fig. 13, y/o la calculadora 310 de longitud opcional, el recuperador 320 de elemento y el determinador 330 de índices opcional de la Fig. 14 y/o la unidad 410 de correspondencia, el determinador 420 de información, la unidad 430 de codificación y la unidad 440 de asociación de la Fig. 15, y/o el determinador 510 de relación, el determinador

520 de bandera y la unidad 530 de asociación de la Fig. 16, y/o el determinador 610 del elemento de sintaxis numérico, el determinador 620 del elemento de sintaxis de referencia y la unidad 630 de asociación de la Fig. 17.

El programa informático 730 se puede almacenar por completo o en parte, en uno o más medios 720 legibles por ordenador volátiles adecuados o medios de almacenamiento de datos, tales como una RAM, o en uno o más medios legibles por ordenador no volátiles adecuados o medios de almacenamiento de datos, tales como en discos magnéticos, CD-ROM, discos DVD, discos duros, ROM o memoria flash. Los medios 720 de almacenamiento de datos pueden ser medios de almacenamiento de datos locales o proporcionados de manera remota, tales como en un servidor de datos. El software se puede cargar así en la memoria operativa de un ordenador o sistema de procesamiento equivalente para su ejecución por un procesador. El ordenador/procesador no tiene que ser dedicado a sólo ejecutar las funciones anteriormente descritas sino que puede ejecutar también otras tareas de software. Un ejemplo no limitante de código de programa usado para definir el nodo de red incluye un código de una instrucción múltiples datos (SIMD).

El número 700 de referencia en la Fig. 18, así, indica el decodificador, codificador o dispositivo de las Fig. 12-17 implementado al menos parcialmente en software.

El decodificador 100 de la Fig. 12, el dispositivo 200 para determinar la relación de decodificación en la Fig. 13 y/o el dispositivo 300 para determinar la relación de decodificación en la Fig. 14 se pueden disponer de manera ventajosa en un dispositivo de usuario o terminal 53 como se muestra en la Fig. 19. Este dispositivo 53 de usuario podría representar entonces un decodificador, un ordenador, un dispositivo móvil, tal como un teléfono móvil, un portátil o un ordenador de tableta, etc. El dispositivo 200 de la Fig. 13 o el dispositivo 300 de la fig. 14 se implementa de manera ventajosa en el decodificador 100 de la Fig. 12.

El dispositivo 53 de usuario, puede, además de un decodificador 54, tal como un decodificador 100 mostrado en la Fig. 12 u otro decodificador que comprenda un dispositivo 200 de la Fig. 13 o un dispositivo 300 de la Fig. 14, comprender una memoria 55 de imágenes decodificadas (DPB), en la que el dispositivo 53 de usuario almacena las imágenes decodificadas cuando son generadas por el decodificador 54. Estas imágenes decodificadas se almacenan temporalmente en la DPB 55 i) para ser usadas como imágenes de referencia al decodificar las imágenes posteriores, según el orden de decodificación, del flujo de video multicapa y/o ii) hasta que son emitidas, tal como emitidas para ser presentadas en un elemento 56 de presentación o pantalla de o conectadas al dispositivo 53 de usuario.

El decodificador 100 de la Fig. 12, el dispositivo 200 para determinar la relación de decodificación en la Fig. 13 y/o el dispositivo 300 para determinar la relación de decodificación en la Fig. 14 pueden, también o de manera alternativa, ser implementados en un nodo 52 de red o elemento de red. En tal caso, el nodo 52 de red o elemento de red podría realizar, por ejemplo el envío selectivo de capas del flujo de video multicapa codificado, en base a y de este modo considerando las dependencias de capa.

El codificador 400 de la Fig. 15, el dispositivo 500 para determinar la relación de codificación en la Fig. 16 y/o el dispositivo 600 para determinar la relación de codificación en la Fig. 17 se pueden disponer de manera ventajosa en un dispositivo de usuario o terminal 50 como se muestra en la Fig. 19. Este dispositivo 50 de usuario podría entonces representar una video cámara, un ordenador, un dispositivo móvil, tal como un teléfono móvil, un portátil o un ordenador de tableta, etc. El dispositivo 500 de la Fig. 16 o el dispositivo 600 de la Fig. 17 se implementan de manera ventajosa en el codificador 400 de la Fig. 15.

El codificador 51 implementado en el dispositivo 50 de usuario podría ser el codificador 400 mostrado en le Fig. 15 u otro codificador que comprende el dispositivo 500 de la Fig. 16 o el dispositivo 600 de la Fig. 17.

Las imágenes 20, 22, 24 de las múltiples capas 10, 12, 14 en un flujo 1 de video multicapa son introducidas en el codificador 51 del dispositivo 50 de usuario para generar y emitir un flujo 2 de video multicapa codificado, que normalmente comprende unidades NAL que contienen las respectivas representaciones codificadas de las fracciones en las imágenes o los conjuntos de parámetros generados para el flujo 1 de video multicapa.

El flujo 2 de video multicapa codificado se transmite preferiblemente, de manera inalámbrica o usando una conexión por cable desde el dispositivo 50 de usuario hacia el dispositivo 53 de usuario, donde el decodificador 54 decodifica el flujo 2 de video multicapa para generar imágenes decodificadas que puedan ser mostradas en el elemento de presentación 56. Ejemplos no limitativos de protocolos de transporte que se podrían usar en una red para transmitir un flujo 2 de video multicapa codificado y/o paquetes de datos que lleven unidades NAL que son enviadas fuera de banda incluyen el RTP, la Transmisión Dinámica Adaptativa obre el Protocolo de Transferencia de Hipertexto (DASH), el Flujo de Transporte MPEG2 (TS). La red es preferiblemente una red inalámbrica, tal como una red móvil (de banda ancha).

El flujo 2 de video multicapa codificado se puede transmitir desde el dispositivo 50 de usuario codificador al dispositivo 53 de usuario decodificador a través de uno o más nodos 52 de red. Dichos nodos 52 de red podrían de manera opcional procesar el flujo 2 de video multicapa codificado, así como transcodificar el flujo de video multicapa, o eliminar una o más capas del flujo 2 de video multicapa codificado para formar un subflujo 3 de video multicapa codificado. Dicho descarte de capa podría ser debido a una adaptación de la tasa de bit, por ejemplo, en el caso de

una congestión de red; una adaptación de formato, por ejemplo en el caso de un dispositivo 53 de usuario decodificador que pueda sólo decodificar o presentar una cierta resolución máxima; o una adaptación 2D/3D, por ejemplo en el caso que el dispositivo 53 de usuario sólo pueda decodificar o presentar un cierto número de vistas.

5 Las presentes realizaciones son aplicables en concreto a la HEVC y a otros estándares de codificación de video capaces de codificar y decodificar video multicapa, así como video multivista, video 3D o video escalable.

10 Modificaciones y otras variantes de las realizaciones descritas vendrán a la mente de alguien experto en la técnica que tiene el beneficio de las enseñanzas presentadas en las descripciones anteriores y los dibujos asociados. Por lo tanto, se ha de entender que las realizaciones no se limitan a los ejemplos específicos descritos y que las modificaciones y otras variantes están destinadas a ser incluidas dentro del alcance de la descripción. En concreto, las realizaciones descritas en la presente memoria se pueden combinar allá donde sea técnicamente factible.

REIVINDICACIONES

1. Un método de decodificación de un flujo (2) de video multicapa codificado que define múltiples capas (10, 12, 14) de imágenes (20, 22, 24), teniendo cada capa (10, 12, 14) de dichas múltiples capas (10, 12, 14) un respectivo identificador de capa, dicho método comprende:

5 recuperar (S1), para una capa (12) con índice i de capa de dichas múltiples capas (10, 12, 14), decodificar la información de relación basada en dicho flujo (2) de video multicapa codificado, definiendo dicha información de relación de decodificación una respectiva bandera de dependencia directa, bandera_dependencia_directa[i][j], para cada capa (10) de dichas múltiples capas (10, 12, 14) que tiene un índice j de capa para el que $0 \leq j < i$, en donde una bandera de dependencia directa, bandera_dependencia_directa[i][j], igual a 1 indica que dicha capa (10) con índice j de capa es una capa de referencia directa para dicha capa (12) con índice i de capa y una bandera de dependencia directa, bandera_dependencia_directa[i][j], igual a 0 indica que dicha capa (10) con índice j de capa no es una capa de referencia directa para dicha capa (12) con índice i de capa;

recuperar una bandera bandera_presente_id_capa_nuh_vps basada en dicho flujo (2) de video multicapa codificado;

15 fijar, para cada capa (10) de referencia y para dicha capa (12) con índice i de capa y si bandera_presente_id_capa_nuh_vps = 0, su identificador de capa igual a su índice de capa.

recuperar, para cada capa (10) y para dicha capa (12) con índice i de capa y si bandera_presente_id_capa_nuh_vps = 1, su identificador de capa de un vector nuh_en_id_capa[i], $i \in [1, \text{capas_max_vps_menos1}]$, en donde capas_max_vps_menos1 + 1 indica el número máximo de capas y nuh_en_id_capa[i] indica un identificador de capa para una capa con índice i de capa; y

20 decodificar (S3) una imagen (22) de dicha capa (12) con índice i de capa basada en al menos una imagen (20) previamente decodificada en una capa (10) de dichas múltiples capas (10, 12, 14) identificada en base a dichos identificadores de capa.

2. El método según la reivindicación 1, en donde la recuperación (S1) de dicha información de relación de decodificación comprende recuperar (S10), de un conjunto de parámetros de video o de una extensión del conjunto de parámetros de video asociada con dicho flujo (2) de video multicapa codificado, dicha respectiva bandera de dependencia directa.

3. Un decodificador (700) configurado para decodificar un flujo (2) de video multicapa codificado que define múltiples capas (10, 12, 14) de imágenes (20, 22, 24), cada capa (10, 12, 14) de dichas múltiples capas (10, 12, 14) teniendo un respectivo identificador de capa, dicho decodificador (700) comprende un procesador (710) configurado para procesar medios (740) de código de un programa informático (730) almacenado en un medio (720) legible por ordenador, dichos medios de código (740) provocan, cuando se ejecutan en dicho procesador (710), a dicho procesador a:

35 recuperar, para una capa (12) con un índice i de capa de múltiples capas (10, 12, 14), la información de relación de decodificación basada en dicho flujo (2) de video multicapa codificado, definiendo dicha información de relación de decodificación una respectiva bandera de dependencia directa, bandera_dependencia_directa[i][j], para cada capa (10) de dichas múltiples capas (10, 12, 14) que tiene un índice j de capa para el que $0 \leq j < i$, en donde una bandera de dependencia directa, bandera_dependencia_directa[i][j], igual a 1 indica que dicha capa (10) con índice j de capa es una capa de referencia directa para dicha capa (12) con índice i de capa y una bandera de dependencia directa, bandera_dependencia_directa[i][j], igual a 0 indica que dicha capa (10) con índice j de capa no es una capa de referencia directa para dicha capa (12) con índice i de capa;

recuperar una bandera bandera_presente_id_capa_nuh_vps basada en dicho flujo (2) de video multicapa codificado;

fijar, para cada capa (10) de referencia y para dicha capa (12) con índice i de capa y si bandera_presente_id_capa_nuh_vps = 0, su identificador de capa igual a su índice de capa;

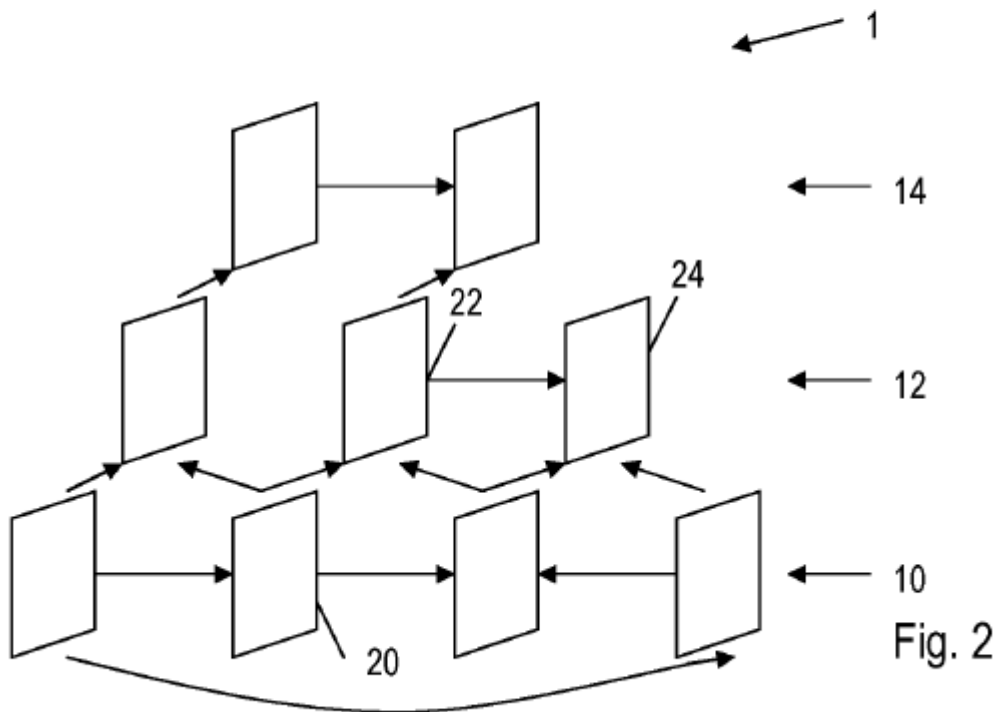
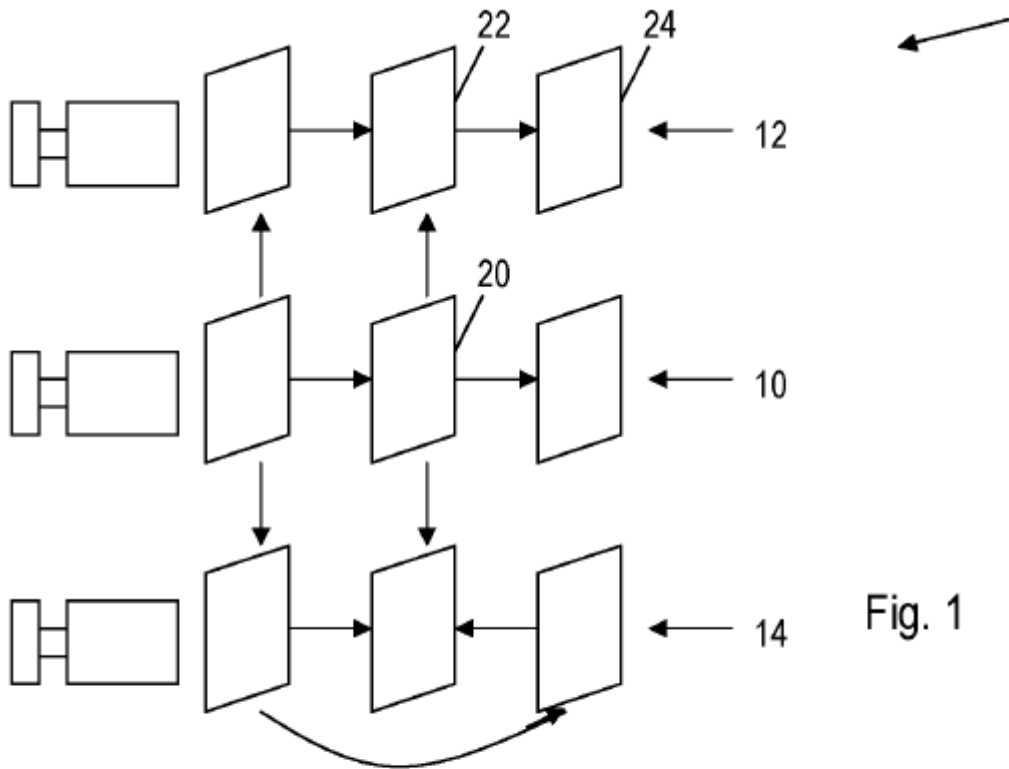
45 recuperar, para cada capa (10) de referencia y para dicha capa (12) con índice i de capa y si bandera_presente_id_capa_nuh_vps = 1, su identificador de capa de un vector nuh_en_id_capa[i], $i \in [1, \text{capas_max_vps_menos1}]$, en donde, capas_max_vps_menos1 + 1 indica el número máximo de capas y nuh_en_id_capa[i] indica el identificador de capa de una capa con índice i de capa; y

decodificar una imagen (22) de dicha capa (12) basada en al menos una imagen (20) previamente decodificada en una capa (10) de dichas múltiple capas (10, 12, 14) identificada en base a dichos identificadores de capa.

50 4. El decodificador según la reivindicación 3, configurado para recuperar dicha información de relación de decodificación de un conjunto de parámetros de video o extensión del conjunto de parámetros de video asociado con dicho flujo (2) de video multicapa codificado.

5. Un dispositivo (53) de usuario que comprende dicho decodificador (700) según la reivindicación 3.

6. Un nodo de red (52) que comprende dicho decodificador (700) según la reivindicación 3.



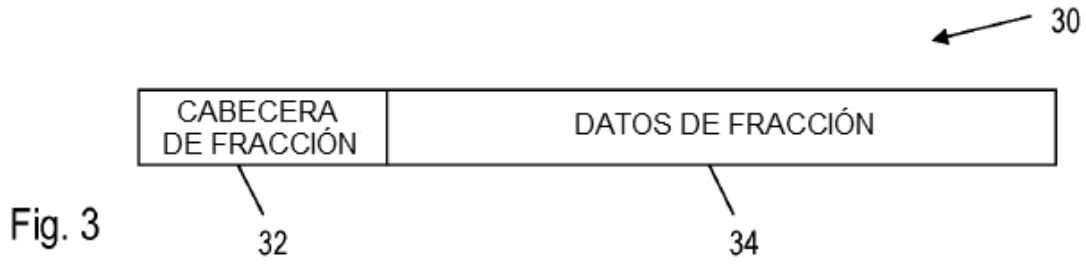


Fig. 3

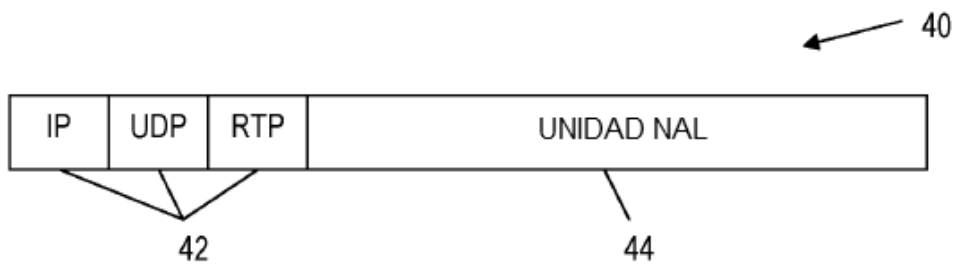


Fig. 4

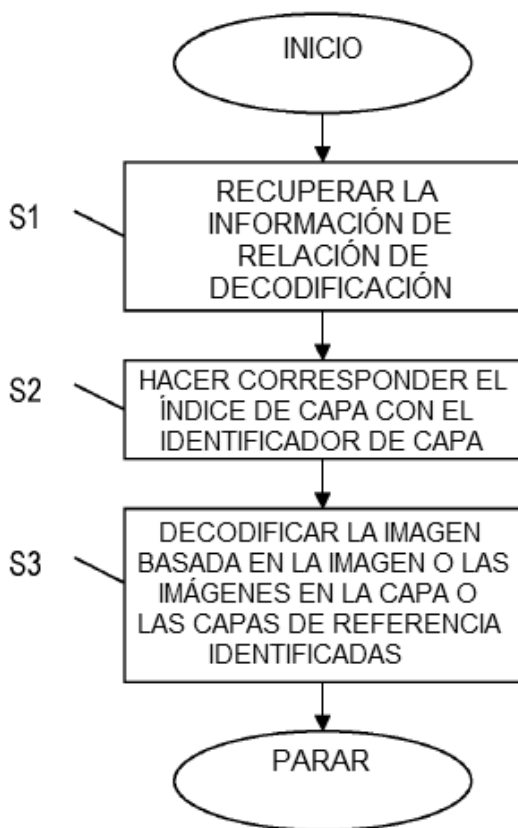


Fig. 5

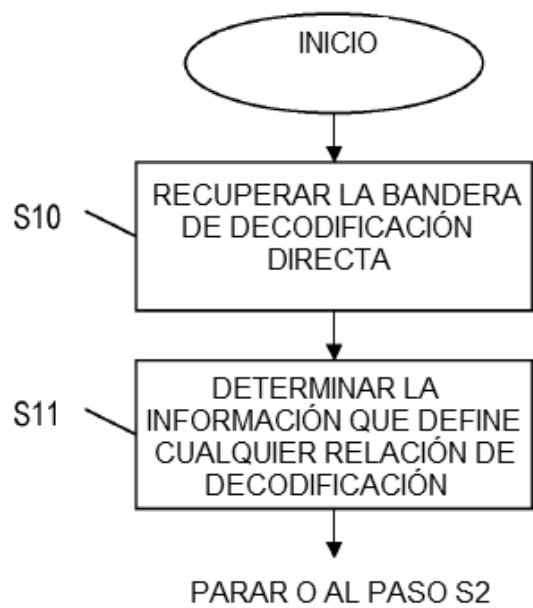


Fig. 6

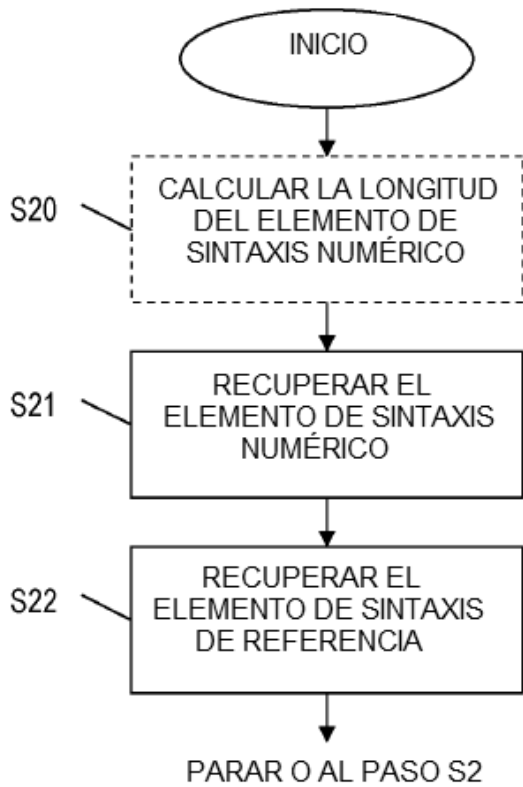


Fig. 7

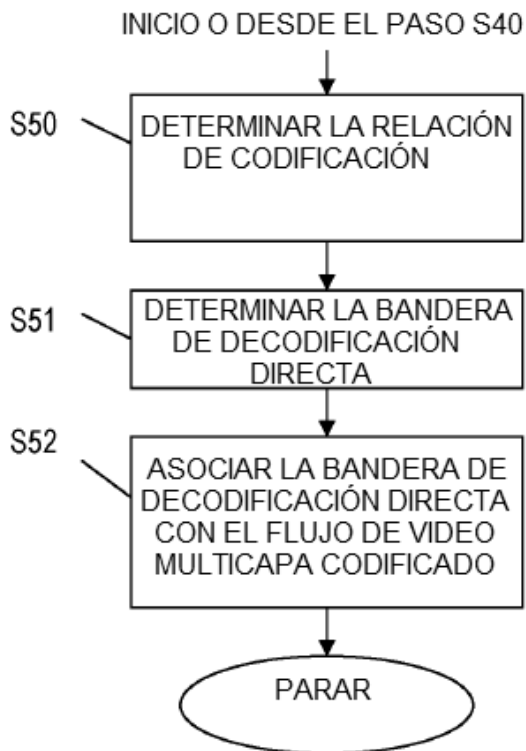


Fig. 10

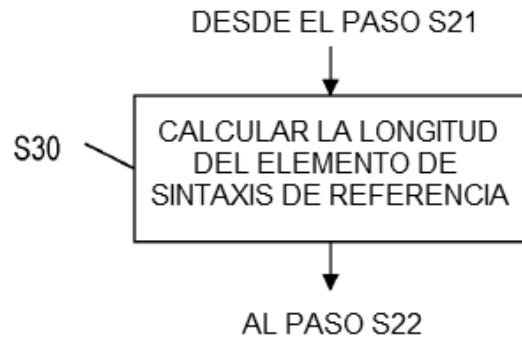


Fig. 8

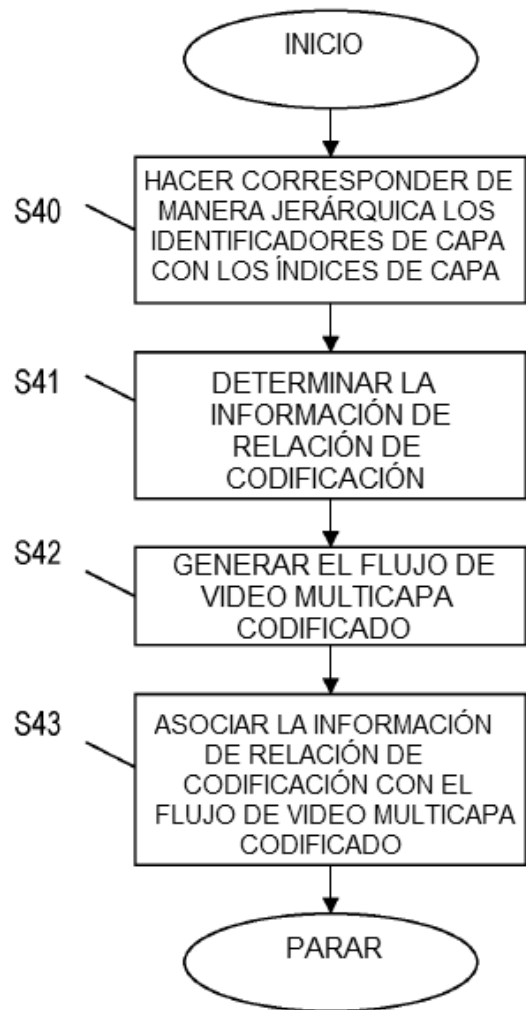


Fig. 9

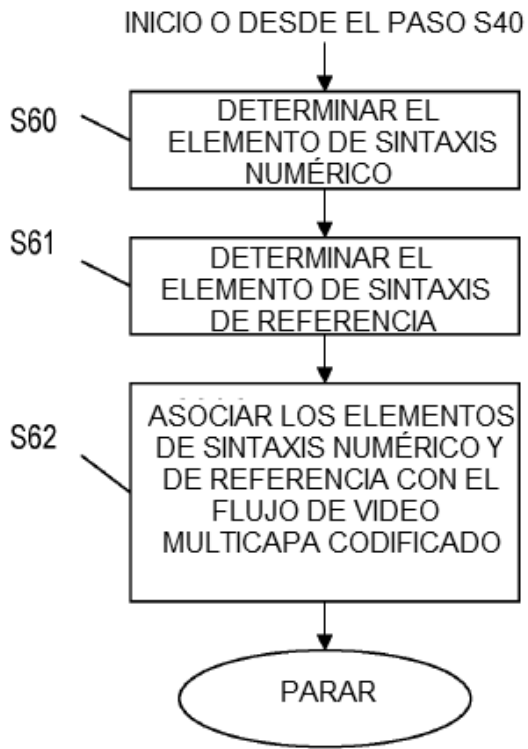


Fig. 11

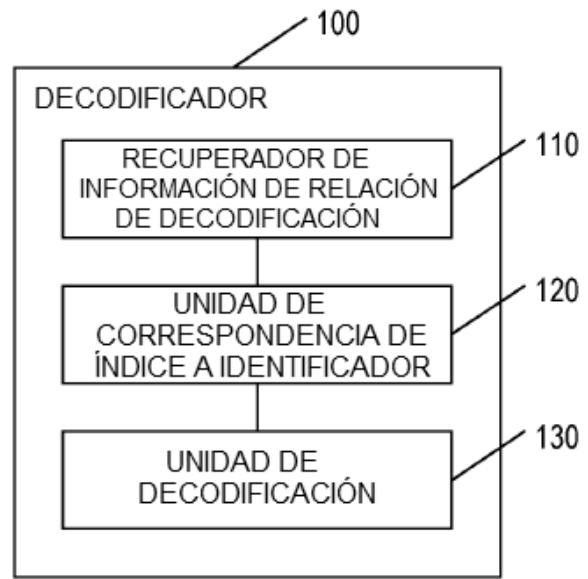


Fig. 12

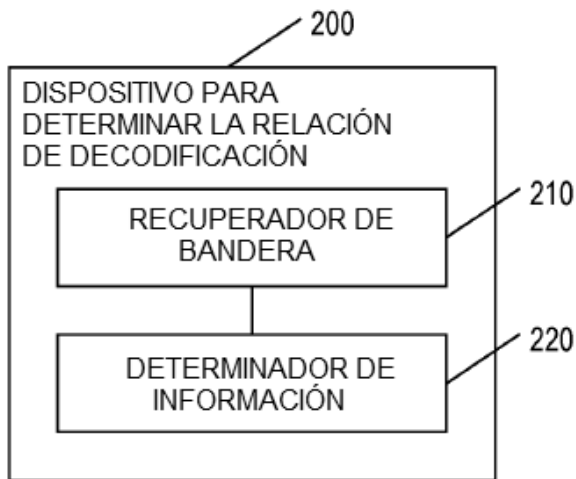


Fig. 13

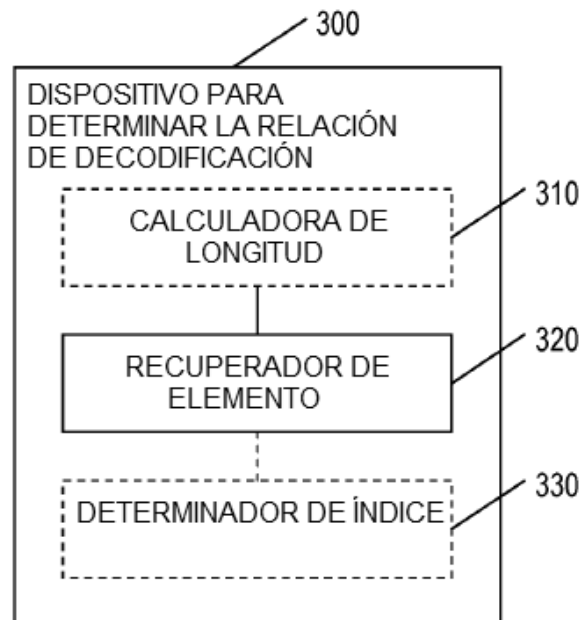


Fig. 14

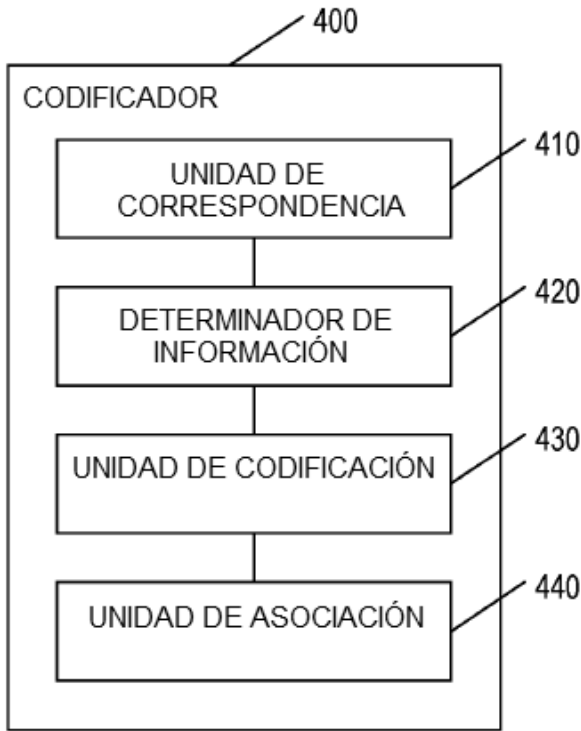


Fig. 15

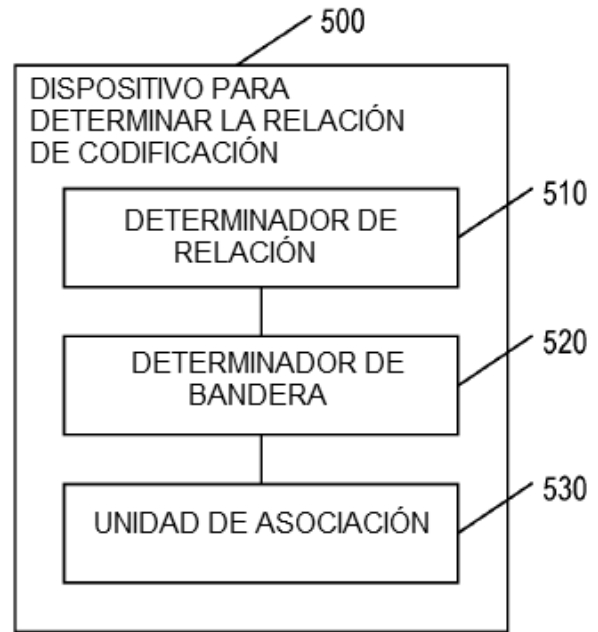


Fig. 16

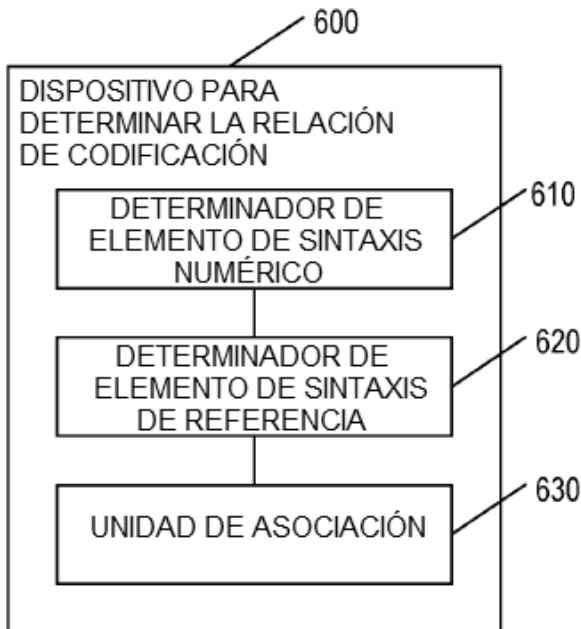


Fig. 17

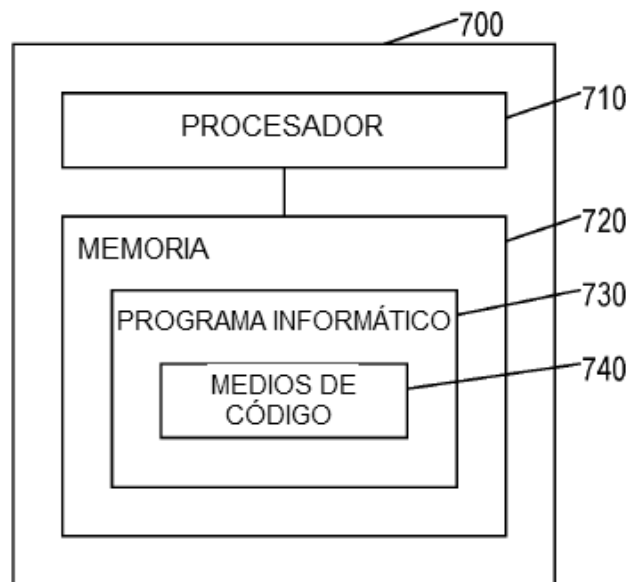


Fig. 18

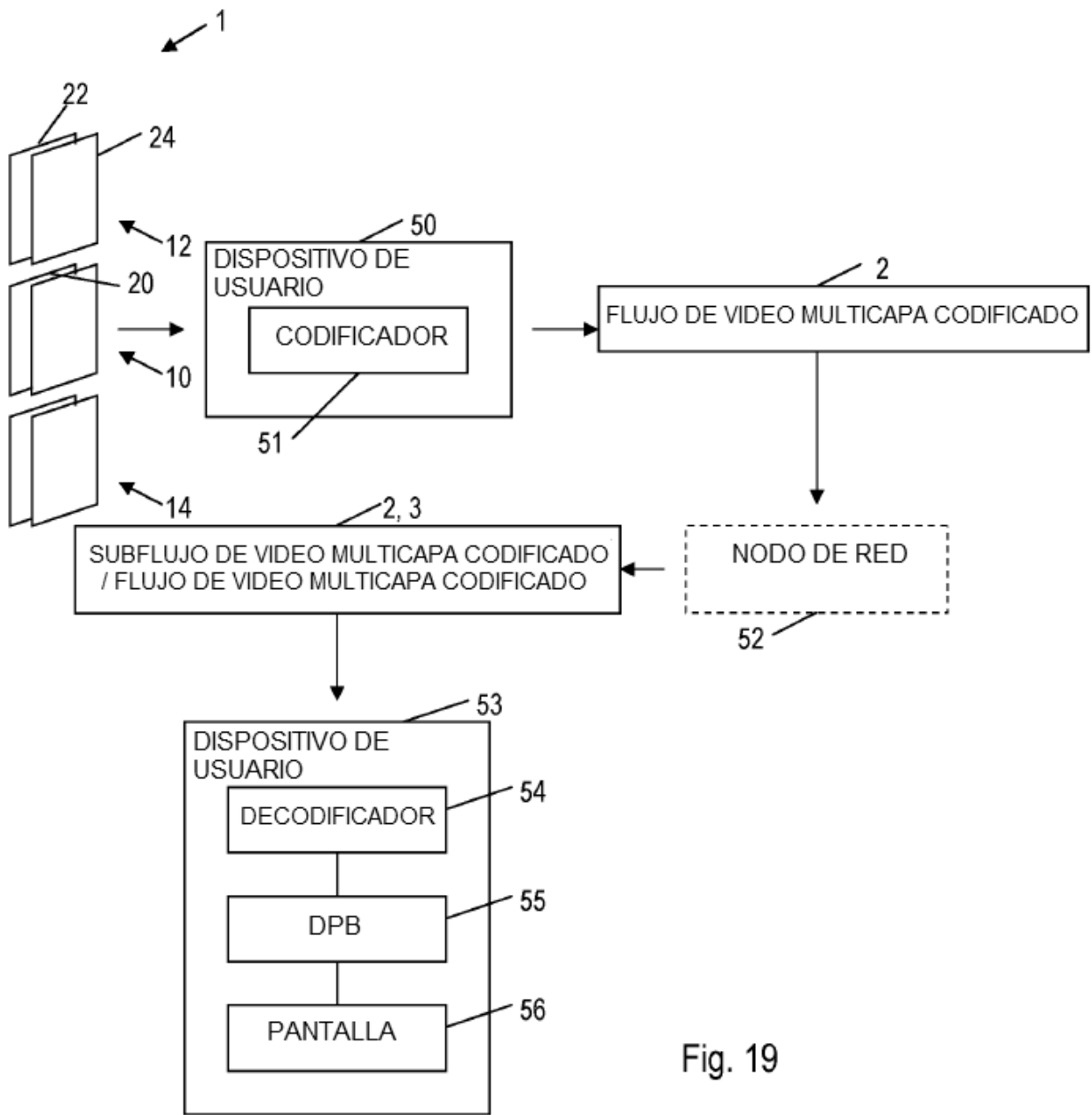


Fig. 19

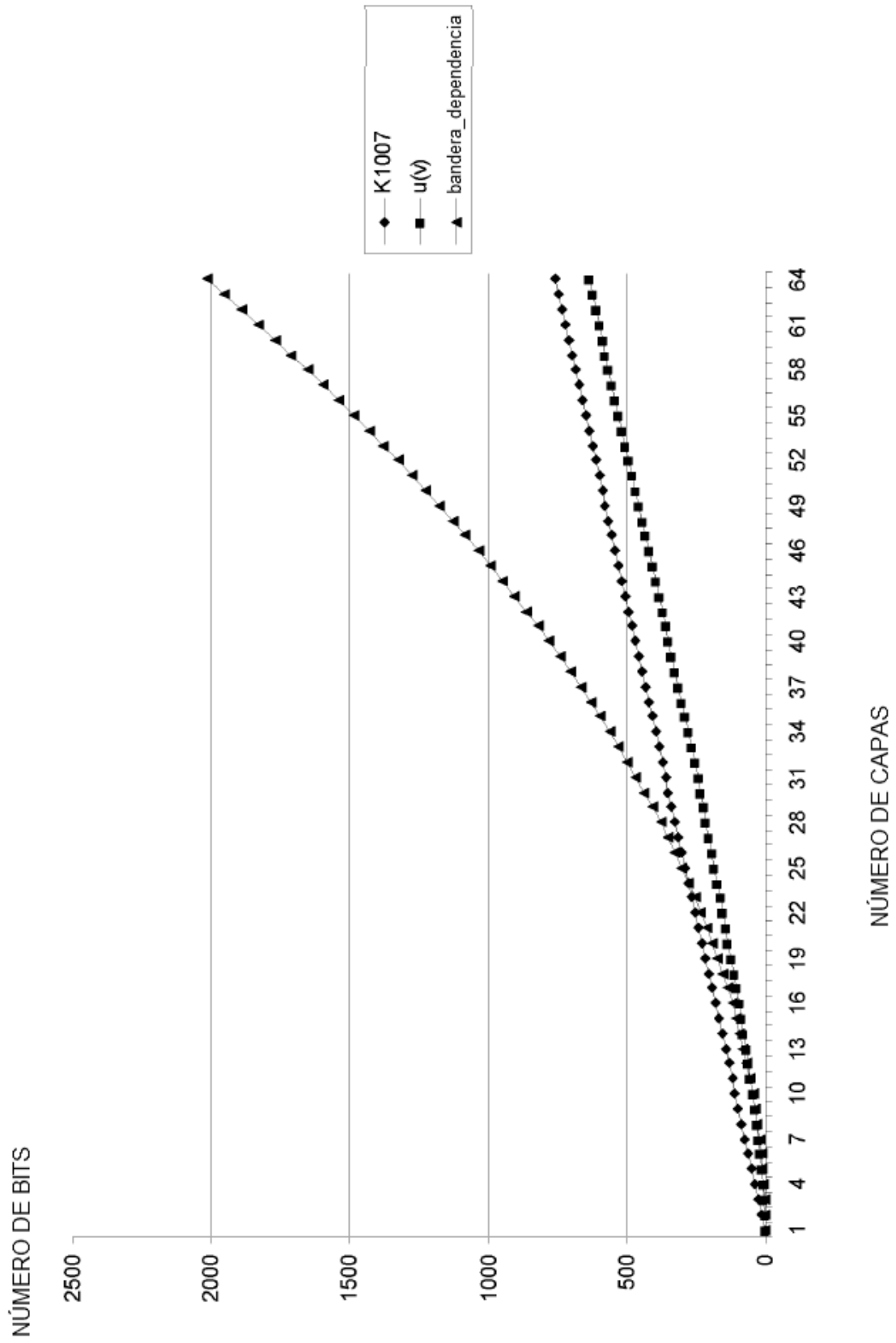


Fig. 20

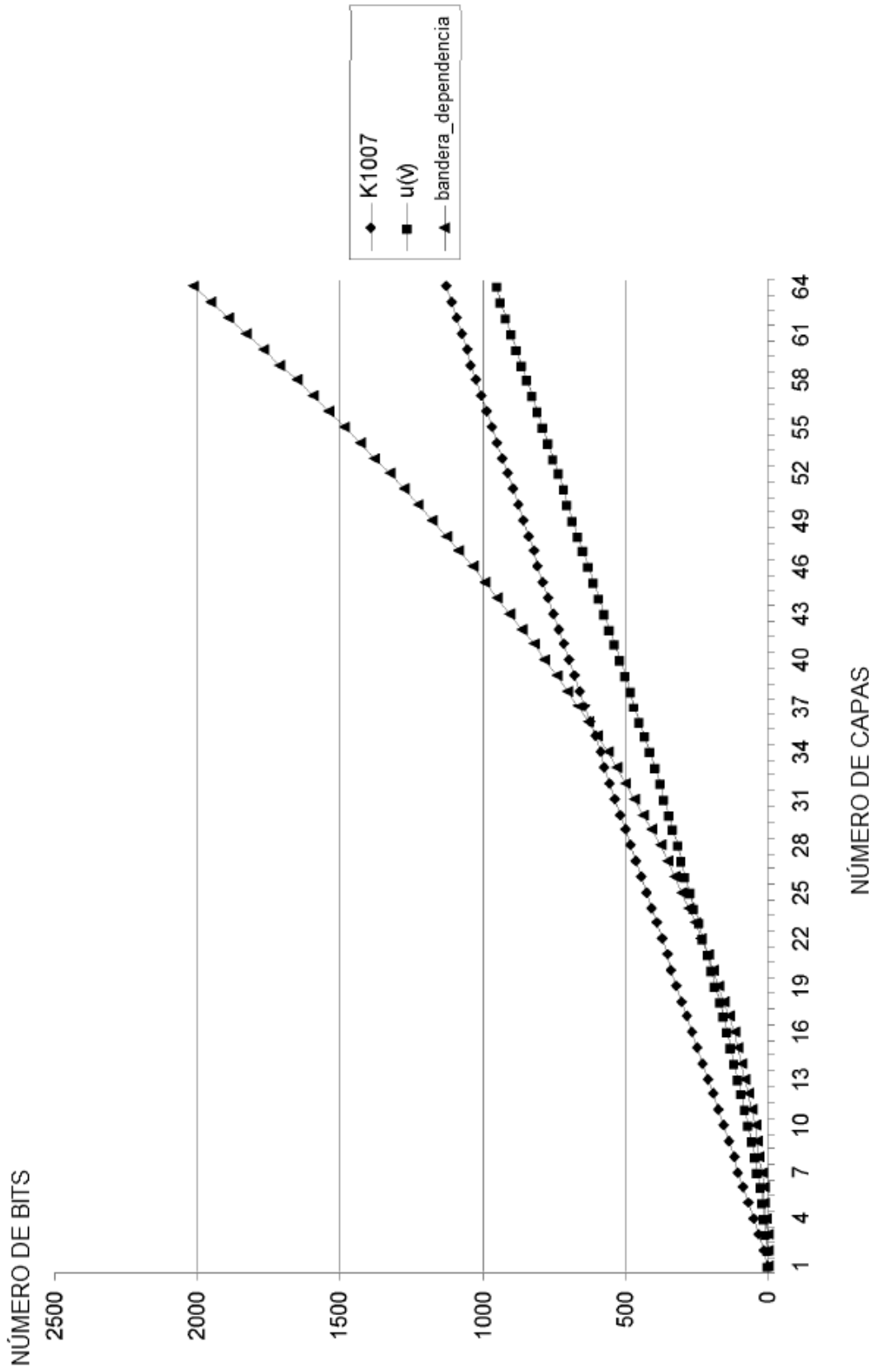


Fig. 21

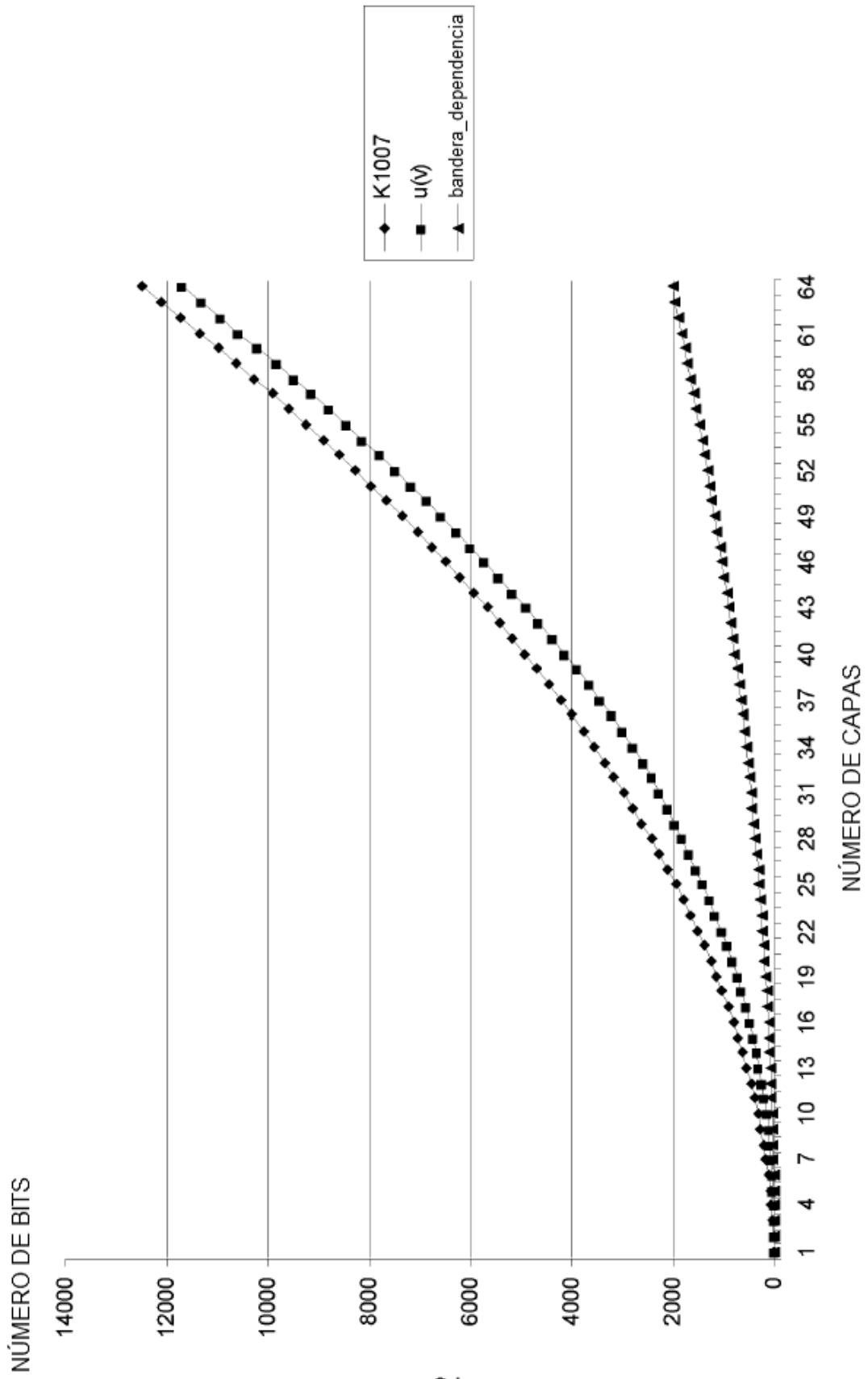


Fig. 22

