

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **3 014 695**

51 Int. Cl.:

H04N 19/103 (2014.01)
H04N 19/147 (2014.01)
H04N 19/186 (2014.01)
H04N 19/46 (2014.01)
H04N 21/21 (2011.01)
H04N 21/20 (2011.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **25.11.2019** E 19211222 (5)

97 Fecha y número de publicación de la concesión europea: **12.02.2025** EP 3672255

54 Título: **Un aparato, un método y un programa informático de codificación y decodificación de vídeo**

30 Prioridad:

17.12.2018 FI 20186097

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

24.04.2025

73 Titular/es:

**NOKIA TECHNOLOGIES OY (100.00%)
Karakaari 7
02610 Espoo, FI**

72 Inventor/es:

LAINEMA, JANI

74 Agente/Representante:

DEL VALLE VALIENTE, Sonia

ES 3 014 695 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

Un aparato, un método y un programa informático de codificación y decodificación de vídeo

5 Campo técnico

La presente invención se refiere a un aparato, un método y un programa informático para la codificación y decodificación de vídeo.

10 Antecedentes

Las muestras de vídeo e imagen se codifican típicamente usando representaciones de color tales como YUV o YCbCr que consisten en un canal de luminancia y dos de crominancia. En estos casos, el canal de luminancia, que representa principalmente la iluminación de la escena, se codifica típicamente con una resolución determinada, mientras que los canales de crominancia, que representan típicamente diferencias entre ciertos componentes de color, se codifican a menudo con una segunda resolución inferior a la de la señal de luminancia.

La intención de este tipo de representación diferencial es descorrelacionar los componentes de color y poder comprimir los datos de manera más eficiente. Sin embargo, en muchos casos, aún queda algo de correlación entre los canales que podrían utilizarse para representar los datos de manera más eficiente.

El documento US-9.148.672 describe un método para un sistema de codificación de imagen/vídeo de múltiples canales, donde, en un proceso de generación de residuos, se aplica un conjunto de operaciones de números enteros a los datos residuales a través de los canales de entrada para generar datos transformados de residuos que tienen múltiples canales de salida. Los datos de residuos transformados asociados a un primer canal de salida están relacionados con la diferencia entre unos primeros datos de residuos asociados a un primer canal de entrada y unos segundos datos de residuos asociados a un segundo canal de entrada.

El documento WO 2014/071439 describe un método para decodificar una transformada de luma y una pluralidad de transformadas de croma a partir de un flujo de bits de vídeo. Las transformadas de croma contienen datos de croma para un único canal de color. Los valores del indicador de omisión de transformada tanto para la luminancia como para la croma se determinan para indicar si los datos de las transformadas de luminancia y croma están codificados en el flujo de bits de vídeo como una representación de dominio espacial. La transformada de luminancia se decodifica entonces según el indicador de omisión de transformada de luminancia determinado, es decir, omite un módulo de transformada si lo indica el indicador, y la pluralidad de transformadas de croma se decodifica según el indicador de omisión de transformada de croma determinado, es decir, si omite o no el módulo de transformada.

El documento US-2013/223518 describe un método de codificación de imágenes que incluye: el procesamiento de nodos en un nodo de una estructura de árbol y la codificación de un coeficiente de frecuencia de un bloque de imágenes de un nodo hoja en la estructura de árbol o un coeficiente de frecuencia de un bloque de imágenes de su nodo padre. El procesamiento de nodos incluye: cuando el procesamiento de nodos se realiza en un nodo padre que tiene nodos secundarios, (i) asignar (a) una posición de un bloque de imagen correspondiente a un nodo hijo actual de entre los nodos hijos y (b) una posición de un bloque de imagen correspondiente al nodo padre, a los argumentos del procesamiento de nodos, y (ii) llamar recursivamente al procesamiento de nodos para el nodo hijo actual, y cuando el procesamiento de nodos se realiza en un nodo hoja, (i) asignar (a) una posición de un bloque de imagen correspondiente al nodo hoja y (b) una posición de una imagen bloque correspondiente a un nodo padre del nodo hoja, a los argumentos del procesamiento de codificación, y (ii) llamar al procesamiento de codificación.

50 Resumen

Pues bien, para al menos resolver los problemas anteriores, en la presente memoria se presenta un método para codificación y decodificación de canales de color. Diversos aspectos incluyen un método, un aparato y un medio legible por ordenador que comprende un programa informático, o una señal almacenada en el mismo, que se caracteriza por lo que se indica en las reivindicaciones independientes. En las reivindicaciones dependientes y en las imágenes y la descripción correspondientes se describen varios detalles de las realizaciones.

Las reivindicaciones independientes exponen el alcance de protección pretendido para diversas realizaciones de la invención. Debe interpretarse que las realizaciones y las características, si las hubiera, descritas en esta memoria descriptiva que no estén dentro del alcance de las reivindicaciones independientes son ejemplos útiles para comprender diversas realizaciones de la invención.

Breve descripción de los dibujos

Para una mejor comprensión de la presente invención, ahora se hará referencia a modo de ejemplo a los dibujos adjuntos, en los que:

la Figura 1 muestra esquemáticamente un dispositivo electrónico que emplea realizaciones de la invención;

la Figura 2 muestra esquemáticamente un equipo de usuario adecuado para emplear realizaciones de la invención;

5 la Figura 3 muestra además dispositivos electrónicos de forma esquemática que emplean realizaciones de la invención conectados usando conexiones de red inalámbricas y alámbricas;

la Figura 4 muestra esquemáticamente un codificador adecuado para implementar realizaciones de la invención;

10 la Figura 5 muestra un diagrama de flujo de un método según una realización de la invención;

las Figuras 6a y 6b muestran un ejemplo de codificación residual según el estado de la técnica y según una realización de la invención, respectivamente;

15 la Figura 7 muestra un diagrama esquemático de un decodificador adecuado para implementar realizaciones de la invención; y

la Figura 8 muestra un diagrama esquemático de un sistema de comunicación multimedia de ejemplo dentro del cual pueden implementarse diversas realizaciones.

20

Descripción detallada de algunas realizaciones ilustrativas

En lo que sigue se describen con más detalle un aparato adecuado y unos posibles mecanismos para iniciar un cambio de punto de vista. A este respecto, en primer lugar, se hace referencia a las Figuras 1 y 2, donde la Figura 1 muestra un diagrama de bloques de un sistema de codificación de vídeo según una realización ilustrativa como un diagrama de bloques esquemático de un aparato ilustrativo o dispositivo electrónico 50, que puede incorporar un códec según una realización de la invención. La Figura 2 muestra una distribución de un aparato según una realización ilustrativa. Los elementos de las Figuras 1 y 2 se explicarán a continuación.

25

30 El dispositivo electrónico 50 puede ser, por ejemplo, un terminal móvil o equipo de usuario de un sistema de comunicación inalámbrica. Sin embargo, se apreciará que las realizaciones de la invención pueden implementarse dentro de cualquier dispositivo o aparato electrónico que pueda requerir codificación y decodificación o codificación o decodificación de imágenes de vídeo.

35 El aparato 50 puede comprender un alojamiento 30 para incorporar y proteger el dispositivo. El aparato 50 puede comprender además una pantalla 32 en forma de una pantalla de cristal líquido. En otras realizaciones de la invención, la pantalla puede ser cualquier tecnología de visualización adecuada para mostrar una imagen o vídeo. El aparato 50 puede comprender además un teclado numérico 34. En otras realizaciones de la invención se puede emplear cualquier mecanismo de interfaz de datos o de usuario adecuado. Por ejemplo, la interfaz de usuario puede implementarse como un teclado virtual o un sistema de entrada de datos como parte de una pantalla sensible al tacto.

40

El aparato puede comprender un micrófono 36 o cualquier entrada de audio adecuada que puede ser una entrada de señal digital o analógica. El aparato 50 puede comprender además un dispositivo de salida de audio que, en realizaciones de la invención, puede ser uno cualquiera de: un auricular 38, un altavoz o una conexión de salida de audio analógico o de audio digital. El aparato 50 también puede comprender una batería (o, en otras realizaciones de la invención, el dispositivo puede ser alimentado por cualquier dispositivo de energía móvil adecuado, tal como un panel solar, una pila de combustible o un generador de relojería). El aparato puede comprender además una cámara que sea capaz de registrar o capturar imágenes y/o vídeo. El aparato 50 puede comprender además un puerto de infrarrojos para la comunicación de línea de visión de corto alcance a otros dispositivos. En otras realizaciones, el aparato 50 puede comprender además cualquier solución de comunicación de corto alcance adecuada tal como, por ejemplo, una conexión inalámbrica Bluetooth o una conexión alámbrica USB/firewire.

45

50

El aparato 50 puede comprender un controlador 56, un procesador o una circuitería procesadora para controlar el aparato 50. El controlador 56 puede estar conectado a la memoria 58 que, en realizaciones de la invención, puede almacenar ambos datos en forma de datos de imagen y de audio y/o también puede almacenar instrucciones para la implementación en el controlador 56. El controlador 56 puede conectarse además a los circuitos 54 de códec adecuados para llevar a cabo la codificación y decodificación de datos de audio y/o vídeo o ayudar en la codificación y decodificación llevada a cabo por el controlador.

55

60 El aparato 50 puede comprender además un lector 48 de tarjetas y una tarjeta inteligente 46, por ejemplo, un UICC y lector de UICC para proporcionar información de usuario y ser adecuado para proporcionar información de autenticación para autenticación y autorización del usuario en una red.

65

El aparato 50 puede comprender circuitos 52 de interfaz de radio conectados al controlador y adecuados para generar señales de comunicación inalámbrica, por ejemplo, para comunicación con una red de comunicaciones celulares, un sistema de comunicaciones inalámbricas o una red de área local inalámbrica. El aparato 50 puede comprender

además una antena 44 conectada a los circuitos 52 de interfaz de radio para transmitir señales de radiofrecuencia generadas en los circuitos 52 de interfaz de radio a otro u otros aparatos y para recibir señales de radiofrecuencia de otro u otros aparatos.

5 El aparato 50 puede comprender una cámara capaz de grabar o detectar fotogramas individuales que a continuación se pasan al códec 54 o al controlador para su procesamiento. El aparato puede recibir los datos de imágenes de vídeo para su procesamiento desde otro dispositivo antes de la transmisión y/o almacenamiento. El aparato 50 también puede recibir de manera inalámbrica o alámbrica la imagen para codificar/decodificar. Los elementos estructurales del aparato 50 descritos anteriormente representan unos ejemplos de unos medios para realizar una función correspondiente.

10 Con respecto a la Figura 3, se muestra un ejemplo de un sistema dentro del cual se pueden utilizar realizaciones de la presente invención. El sistema 10 comprende múltiples dispositivos de comunicación que pueden comunicarse a través de una o más redes. El sistema 10 puede comprender cualquier combinación de redes cableadas o inalámbricas, incluidas, sin limitación, una red de telefonía celular inalámbrica (tal como una red GSM, UMTS, CDMA, etc.), una red de área local inalámbrica (WLAN), tal y como define cualquiera de los estándares IEEE 802.x, una red de área personal Bluetooth, una red de área local Ethernet, una red de área local en anillo con paso de testigo, una red de área amplia e internet.

15 El sistema 10 puede incluir tanto dispositivos de comunicación inalámbrica como alámbrica y/o aparatos 50 adecuados para implementar realizaciones de la invención.

20 Por ejemplo, el sistema mostrado en la Figura 3 muestra una red 11 de telefonía móvil y una representación de internet 28. La conectividad a internet 28 puede incluir, aunque no de forma limitativa, conexiones inalámbricas de largo alcance, conexiones inalámbricas de corto alcance y diversas conexiones alámbricas que incluyen, aunque no de forma limitativa, líneas telefónicas, líneas de cable, líneas eléctricas y rutas de comunicación similares.

25 Los dispositivos de comunicación de ejemplo mostrados en el sistema 10 pueden incluir, aunque no de forma limitativa, un dispositivo o aparato electrónico 50, una combinación de un personal digital assistant (asistente digital personal - PDA) y un teléfono móvil 14, un PDA 16, un integrated messaging device (dispositivo de mensajería integrada - IMD) 18, un ordenador 20 de sobremesa, una computadora portátil 22. El aparato 50 puede ser estacionario o móvil cuando se transporta por un individuo que está en movimiento. El aparato 50 también puede estar ubicado en un modo de transporte incluyendo, pero sin limitarse a, un automóvil, un camión, un taxi, un autobús, un tren, un barco, un avión, una bicicleta, una motocicleta o cualquier modo de transporte adecuado similar.

30 Las realizaciones también pueden implementarse en un decodificador de salón; es decir, un receptor de TV digital, que puede/puede no tener una pantalla o capacidades inalámbricas, en tabletas u personal computers (ordenadores personales - PC) (sobremesa), que tienen hardware o software o una combinación de las implementaciones de codificador/decodificador, en diversos sistemas operativos, y en conjuntos de chips, procesadores, DSP y/o sistemas integrados que ofrecen codificación basada en hardware/software.

35 Algunos o más aparatos pueden enviar y recibir llamadas y mensajes y comunicarse con proveedores de servicios a través de una conexión inalámbrica 25 a una estación base 24. La estación base 24 puede estar conectada a un servidor de red 26 que permite la comunicación entre la red 11 de telefonía móvil e internet 28. El sistema puede incluir dispositivos de comunicación y dispositivos de comunicación adicionales de diversos tipos.

40 Los dispositivos de comunicación pueden comunicarse usando diversas tecnologías de transmisión que incluyen, aunque no de forma limitativa, code division multiple access (acceso múltiple por división de código - CDMA), global systems for mobile communications (sistemas globales para comunicaciones móviles - GSM), universal mobile telecommunications system (sistema universal de telecomunicaciones móviles - UMTS), time divisional multiple access (acceso múltiple por división de tiempo - TDMA), frequency division multiple access (acceso múltiple por división de frecuencia - FDMA), transmission control protocol-internet protocol (protocolo de control de transmisión-protocolo de internet - TCP-IP), short messaging service (servicio de mensajería corta - SMS), multimedia messaging service (servicio de mensajería multimedia (MMS), correo electrónico, instant messaging service (servicio de mensajería instantánea - IMS), Bluetooth, IEEE 802.11 y cualquier tecnología de comunicación inalámbrica similar. Un dispositivo de comunicaciones involucrado en la implementación de diversas realizaciones de la presente invención puede comunicarse usando diversos medios que incluyen, aunque no de forma limitativa, conexiones de radio, infrarrojos, láser, cables y cualquier conexión adecuada.

45 En las redes de telecomunicaciones y de datos, un canal puede referirse o bien a un canal físico, o bien a un canal lógico. Un canal físico puede referirse a un medio de transmisión físico, tal como un cable, mientras que un canal lógico puede referirse a una conexión lógica a través de un medio multiplexado que sea capaz de transmitir varios canales lógicos. Se puede usar un canal para transmitir una señal de información, por ejemplo, un flujo de bits, desde uno o varios emisores (o transmisores) hasta uno o varios receptores.

60

Un flujo de transporte (TS) de MPEG-2, especificado en la norma ISO/IEC 13818-1 o, de manera equivalente, en la Recomendación H.222.0 de la UIT-T, es un formato para transportar audio, vídeo y otros medios, así como metadatos de programa u otros metadatos, en un flujo multiplexado. Se usa un identificador de paquete (PID) para identificar un flujo elemental (también conocido como flujo elemental paquetizado) dentro del TS. Así pues, se puede considerar que un canal lógico dentro de un TS de MPEG-2 corresponde a un valor de PID específico.

Entre las normas de formato de archivos de medios disponibles se incluyen el formato de archivo de medios de base de ISO (ISO/IEC 14496-12, que puede abreviarse como ISOBMFF) y el formato de archivo para un vídeo estructurado en unidades NAL (ISO/IEC 14496-15), que deriva del ISOBMFF.

A continuación se describen algunos conceptos, estructuras y especificaciones del ISOBMFF como ejemplo de un formato de archivo contenedor, basándose en el cual se pueden implementar las realizaciones. Los aspectos de la invención no se limitan al ISOBMFF, sino que, más bien, la descripción se da para proporcionar un fundamento posible sobre el que se pueda realizar total o parcialmente la invención.

El componente básico en el formato de archivo de medios de base de ISO se llama caja. Cada caja tiene una cabecera y una carga útil. La cabecera de la caja indica el tipo de caja y el tamaño de la caja en términos de bytes. Una caja puede encerrar otras cajas, y el formato de archivo ISO especifica qué tipos de caja se permiten dentro de una caja de un determinado tipo. Además, la presencia de algunas cajas puede ser obligatoria en cada archivo, mientras que la presencia de otras cajas puede ser opcional. De forma adicional, para algunos tipos de caja, puede permitirse tener más de una caja presente en un archivo. Por lo tanto, el formato de archivo de medios de base ISO puede considerarse que especifica una estructura jerárquica de cajas.

Según la familia ISO de formatos de archivo, un archivo incluye unos datos de medios y unos metadatos que están encapsulados en unas cajas. Cada caja se identifica mediante un código de cuatro caracteres (4CC), que comienza con una cabecera que informa sobre el tipo y el tamaño de la caja.

En archivos que se ajustan al formato de archivo de medios de base de ISO, los datos de medios pueden proporcionarse en una caja "mdat" de datos de medios y la caja "moov" de película puede usarse para encerrar los metadatos. En algunos casos, para que un archivo sea manejable, ambas cajas "mdat" y "moov" deben estar presentes. La caja "moov" de película puede incluir una o más pistas, y cada pista puede residir en una correspondiente caja "trak" de pista. Una pista puede ser uno de los muchos tipos, incluida una pista de medios que se refiere a unas muestras formateadas según un formato de compresión de medios (y a su encapsulación en el formato de archivo de medios de base de ISO).

Pueden usarse unos fragmentos de película cuando, por ejemplo, se graba un contenido en unos archivos ISO para, por ejemplo, evitar perder datos si una aplicación de grabación se queda colgada o sin espacio en memoria o si ocurre algún otro incidente. Sin fragmentos de película, puede producirse una pérdida de datos porque el formato de archivo puede requerir que todos los metadatos, por ejemplo, la caja de película, se escriban en una zona contigua del archivo. Además, cuando se graba un archivo, puede que no haya una suficiente cantidad de espacio en memoria (p. ej., una memoria de acceso aleatorio, o RAM) para almacenar temporalmente una caja de película para el tamaño del almacenamiento disponible, y recalcular el contenido de una caja de película cuando la película está cerrada puede llevar demasiado tiempo. Además, los fragmentos de película pueden permitir el registro y la reproducción simultánea de un archivo usando un analizador de archivos ISO normal. Además, puede que sea necesaria una duración más corta del almacenamiento temporal inicial para una descarga progresiva, p. ej., una recepción y una reproducción simultáneas de un archivo, cuando se usan fragmentos de película y la caja de película inicial es más pequeña en comparación con un archivo que tiene el mismo contenido de medios pero está estructurado sin fragmentos de película.

La característica de fragmentos de película puede permitir dividir los metadatos que, por lo demás, pueden residir en la caja de película en múltiples trozos. Cada trozo puede corresponder a un cierto período de tiempo de una pista. En otras palabras, la característica de fragmento de película puede permitir la intercalación de metadatos de archivo y datos de medios. Por consiguiente, el tamaño de la caja de película puede ser limitado y se realizan los casos de uso mencionados anteriormente.

En algunos ejemplos, las muestras de medios para los fragmentos de película pueden residir en una caja mdat, si se encuentran en el mismo archivo que la caja moov. Sin embargo, para los metadatos de los fragmentos de película, se puede proporcionar una caja moof. La caja moof puede incluir la información para una cierta duración del tiempo de reproducción que hubiera estado previamente en la caja moov. La caja moov aún puede representar una película válida por sí misma pero, además, puede incluir una caja mvex que indica que los fragmentos de película seguirán en el mismo archivo. Los fragmentos de película pueden extender la presentación que está asociada a la caja moov en el tiempo.

Dentro del fragmento de película puede haber un conjunto de fragmentos de pista, que incluyen cualquier parte de cero a una pluralidad por pista. A su vez, los fragmentos de pista pueden incluir cualquier parte de cero a una pluralidad de ejecuciones de pista, cuyo documento de cada una de las cuales es una ejecución contigua de muestras para esa

- 5 pista. Dentro de estas estructuras, muchos campos son opcionales y pueden ser predeterminados. Los metadatos que pueden incluirse en la caja moof pueden limitarse a un subconjunto de los metadatos que pueden incluirse en una caja moov y pueden codificarse de manera diferente en algunos casos. Los detalles con respecto a las cajas que pueden incluirse en una caja moof pueden encontrarse a partir de la especificación de formato de archivo de medios de base ISO. Un fragmento de película autónomo puede definirse como un fragmento que consta de una caja moof y una caja mdat que son consecutivas en el orden de archivos, donde la caja mdat contiene las muestras del fragmento de película (para las que la caja moof proporciona los metadatos) y no contiene muestras de ningún otro fragmento de película (i.e., ninguna otra caja moof).
- 10 El mecanismo de referencia de pistas se puede usar para asociar unas pistas con otras. TrackReferenceBox incluye un(as) caja(s), cada una de las cuales proporciona una referencia de la pista contenedora a un conjunto de otras pistas. Estas referencias se etiquetan mediante el tipo de caja (es decir, el código de cuatro caracteres de la caja) de la(s) caja(s) contenida(s).
- 15 El formato de archivo de medios de base de ISO contiene tres mecanismos para unos metadatos temporizados que pueden estar asociados a unas muestras particulares: grupos de muestras, pistas de metadatos temporizados e información auxiliar de muestra. Una especificación derivada puede proporcionar una funcionalidad parecida con uno o más de estos tres mecanismos.
- 20 Puede definirse una agrupación de muestras en el formato de medios de base ISO y sus derivados, tal como el formato de archivo AVC y el formato de archivo SVC, como una asignación de cada muestra en una pista para ser un miembro de un grupo de muestras, basándose en un criterio de agrupación. Un grupo de muestras en una agrupación de muestras no se limita a ser muestras contiguas y puede contener muestras no adyacentes. En vista de que puede haber más de una agrupación de muestras para las muestras en una pista, cada agrupación de muestras puede tener un campo tipo para indicar el tipo de agrupación. Las agrupaciones de muestras pueden estar representadas por dos estructuras de datos vinculadas: (1) SampleToGroupBox (caja sbgp) representa la asignación de muestras a grupos de muestras; y (2) SampleGroupDescriptionBox (caja spsd) contiene una entrada de grupo de muestras para cada grupo de muestras que describe las propiedades del grupo. Puede haber múltiples instancias de SampleToGroupBox y SampleGroupDescriptionBox que están basadas en distintos criterios de agrupación. Estas pueden distinguirse por un campo tipo que se usa para indicar el tipo de agrupación. SampleToGroupBox puede comprender un campo grouping_type_parameter que se puede usar para, por ejemplo, indicar un subtipo de la agrupación.
- 30 El formato de archivo Matroska es capaz de almacenar (aunque no está limitado a) cualquiera de unas pistas de vídeo, de audio, de imagen o de subtítulos en un archivo. Matroska se puede usar como formato de fundamento para unos formatos de archivo derivados, tales como WebM. Matroska utiliza como fundamento el metalenguaje binario extensible (EBML). El EBML especifica un formato alineado binario y de octetos (bytes) que se inspira en el principio de XML. El propio EBML es una descripción generalizada de la técnica del marcado binario. Un archivo Matroska consta de unos elementos (Elements) que forman un “documento” de EBML. Los elementos incorporan un ID de elemento, un descriptor del tamaño del elemento y los propios datos binarios. Los elementos pueden estar anidados.
- 35 Un elemento de segmento de Matroska es un contenedor para otros elementos de nivel superior (nivel 1). Un archivo Matroska puede comprender (pero no se limita a estar formado por) un segmento. En los archivos Matroska, los datos multimedia están organizados en conglomerados (o en elementos de conglomerado), cada uno de los cuales normalmente contiene unos pocos segundos de datos multimedia. Un conglomerado comprende unos elementos BlockGroup, que a su vez comprenden unos elementos de bloque. Un elemento de indicaciones comprende unos metadatos que pueden ayudar a la hora de realizar un acceso o búsqueda aleatorio y puede incluir unos punteros de archivo o unos respectivos sellos de tiempo para unos puntos de búsqueda.
- 40 El códec de vídeo consiste en un codificador que transforma el vídeo de entrada en una representación comprimida adecuada para el almacenamiento/transmisión y un decodificador que puede descomprimir la representación de vídeo comprimido de vuelta a una forma visible. Un codificador de vídeo y/o un decodificador de vídeo puede(n) ser también independiente(s) entre sí, es decir, no necesitan formar un códec. Típicamente, el codificador descarta alguna información en la secuencia de vídeo original para representar el vídeo en una forma más compacta (es decir, a una tasa de bits menor).
- 45 El códec de vídeo consiste en un codificador que transforma el vídeo de entrada en una representación comprimida adecuada para el almacenamiento/transmisión y un decodificador que puede descomprimir la representación de vídeo comprimido de vuelta a una forma visible. Un codificador de vídeo y/o un decodificador de vídeo puede(n) ser también independiente(s) entre sí, es decir, no necesitan formar un códec. Típicamente, el codificador descarta alguna información en la secuencia de vídeo original para representar el vídeo en una forma más compacta (es decir, a una tasa de bits menor).
- 50 Los codificadores de vídeo híbridos típicos, por ejemplo, muchas implementaciones de codificador según H.263 y H.264 del UIT-T, codifican la información de vídeo en dos fases. En primer lugar, se predicen los valores de píxel en una cierta área de imagen (o “bloque”), por ejemplo, mediante medios de compensación de movimiento (hallando e indicando un área en uno de los fotogramas de vídeo previamente codificados que corresponden estrechamente al bloque que se está codificando) o por medios espaciales (usando los valores de píxel alrededor del bloque a codificar de una manera especificada). En segundo lugar, se codifica el error de predicción, es decir, la diferencia entre el bloque previsto de píxeles y el bloque original de píxeles. Esto se hace típicamente transformando la diferencia en valores de píxel usando una transformada especificada (p. ej., Discrete Cosine Transform (Transformada de Coseno Discreta - DCT) o una variante de la misma), cuantificando los coeficientes y codificando por entropía los coeficientes cuantificados. Al variar la fidelidad del proceso de cuantificación, el codificador puede controlar el equilibrio entre la precisión de la representación de píxel (calidad de imagen) y el tamaño de la representación de vídeo codificada resultante (tamaño de archivo o tasa de bits de transmisión).
- 55 Los codificadores de vídeo híbridos típicos, por ejemplo, muchas implementaciones de codificador según H.263 y H.264 del UIT-T, codifican la información de vídeo en dos fases. En primer lugar, se predicen los valores de píxel en una cierta área de imagen (o “bloque”), por ejemplo, mediante medios de compensación de movimiento (hallando e indicando un área en uno de los fotogramas de vídeo previamente codificados que corresponden estrechamente al bloque que se está codificando) o por medios espaciales (usando los valores de píxel alrededor del bloque a codificar de una manera especificada). En segundo lugar, se codifica el error de predicción, es decir, la diferencia entre el bloque previsto de píxeles y el bloque original de píxeles. Esto se hace típicamente transformando la diferencia en valores de píxel usando una transformada especificada (p. ej., Discrete Cosine Transform (Transformada de Coseno Discreta - DCT) o una variante de la misma), cuantificando los coeficientes y codificando por entropía los coeficientes cuantificados. Al variar la fidelidad del proceso de cuantificación, el codificador puede controlar el equilibrio entre la precisión de la representación de píxel (calidad de imagen) y el tamaño de la representación de vídeo codificada resultante (tamaño de archivo o tasa de bits de transmisión).
- 60 Los codificadores de vídeo híbridos típicos, por ejemplo, muchas implementaciones de codificador según H.263 y H.264 del UIT-T, codifican la información de vídeo en dos fases. En primer lugar, se predicen los valores de píxel en una cierta área de imagen (o “bloque”), por ejemplo, mediante medios de compensación de movimiento (hallando e indicando un área en uno de los fotogramas de vídeo previamente codificados que corresponden estrechamente al bloque que se está codificando) o por medios espaciales (usando los valores de píxel alrededor del bloque a codificar de una manera especificada). En segundo lugar, se codifica el error de predicción, es decir, la diferencia entre el bloque previsto de píxeles y el bloque original de píxeles. Esto se hace típicamente transformando la diferencia en valores de píxel usando una transformada especificada (p. ej., Discrete Cosine Transform (Transformada de Coseno Discreta - DCT) o una variante de la misma), cuantificando los coeficientes y codificando por entropía los coeficientes cuantificados. Al variar la fidelidad del proceso de cuantificación, el codificador puede controlar el equilibrio entre la precisión de la representación de píxel (calidad de imagen) y el tamaño de la representación de vídeo codificada resultante (tamaño de archivo o tasa de bits de transmisión).
- 65 Los codificadores de vídeo híbridos típicos, por ejemplo, muchas implementaciones de codificador según H.263 y H.264 del UIT-T, codifican la información de vídeo en dos fases. En primer lugar, se predicen los valores de píxel en una cierta área de imagen (o “bloque”), por ejemplo, mediante medios de compensación de movimiento (hallando e indicando un área en uno de los fotogramas de vídeo previamente codificados que corresponden estrechamente al bloque que se está codificando) o por medios espaciales (usando los valores de píxel alrededor del bloque a codificar de una manera especificada). En segundo lugar, se codifica el error de predicción, es decir, la diferencia entre el bloque previsto de píxeles y el bloque original de píxeles. Esto se hace típicamente transformando la diferencia en valores de píxel usando una transformada especificada (p. ej., Discrete Cosine Transform (Transformada de Coseno Discreta - DCT) o una variante de la misma), cuantificando los coeficientes y codificando por entropía los coeficientes cuantificados. Al variar la fidelidad del proceso de cuantificación, el codificador puede controlar el equilibrio entre la precisión de la representación de píxel (calidad de imagen) y el tamaño de la representación de vídeo codificada resultante (tamaño de archivo o tasa de bits de transmisión).

En la predicción temporal, las fuentes de predicción son imágenes previamente decodificadas (también conocidas como imágenes de referencia). En la copia intrabloque (IBC; también conocida como predicción por copia intrabloque), se aplica una predicción de manera parecida a la predicción temporal, pero la imagen de referencia es la imagen actual, y en el proceso de predicción solo se puede hacer referencia a muestras previamente decodificadas. Se puede aplicar una predicción intercapa o entrevista de manera parecida a la predicción temporal, pero la imagen de referencia es una imagen decodificada procedente de otra capa escalable o de otra vista, respectivamente. En algunos casos, la interpredicción puede referirse únicamente a la predicción temporal, mientras que en otros casos la interpredicción puede referirse colectivamente a la predicción temporal y a cualquiera de la copia intrabloque, la predicción intercapa y la predicción entrevista, siempre y cuando se realicen con el mismo proceso que, o con un proceso similar a, el de la predicción temporal. La interpredicción o la predicción temporal puede denominarse a veces compensación de movimiento o predicción compensada en cuanto al movimiento.

La compensación de movimiento se puede realizar con precisión de muestra completa o submuestra. En el caso de una compensación de movimiento precisa de muestra completa, el movimiento se puede representar como un vector de movimiento con valores enteros para el desplazamiento horizontal y vertical y el proceso de compensación de movimiento copia de manera efectiva las muestras de la imagen de referencia utilizando esos desplazamientos. En el caso de una compensación de movimiento precisa en submuestras, los vectores de movimiento se representan mediante valores fraccionarios o decimales para los componentes horizontal y vertical del vector de movimiento. En el caso de que un vector de movimiento se refiera a una posición no entera en la imagen de referencia, típicamente se invoca un proceso de interpolación de submuestras para calcular los valores de muestra pronosticados en función de las muestras de referencia y la posición de submuestra seleccionada. El proceso de interpolación de submuestras consiste típicamente en un filtrado horizontal que compensa los desplazamientos horizontales con respecto a las posiciones de la muestra completa, seguido de un filtrado vertical que compensa los desplazamientos verticales con respecto a las posiciones de la muestra completa. Sin embargo, el procesamiento vertical también se puede realizar antes del procesamiento horizontal en algunos entornos.

La interpredicción, que también puede denominarse predicción temporal, compensación de movimiento o predicción compensada por movimiento, reduce la redundancia temporal. En la interpredicción, las fuentes de predicción son imágenes previamente decodificadas. La intrapredicción utiliza el hecho de que es probable que los píxeles adyacentes dentro de la misma imagen estén correlacionados. La intrapredicción puede realizarse en el dominio espacial o de la transformada, es decir, se pueden predecir valores de muestra o coeficientes de transformada. La intrapredicción se aprovecha típicamente en la intracodificación, donde no se aplica interpredicción.

Un resultado del procedimiento de codificación es un conjunto de parámetros de codificación, tales como vectores de movimiento y coeficientes de transformada cuantificados. Muchos parámetros pueden codificarse por entropía más eficientemente si se predicen en primer lugar de parámetros espacialmente o temporalmente vecinos. Por ejemplo, un vector de movimiento puede predecirse a partir de vectores de movimiento espacialmente adyacentes y puede codificarse solo la diferencia con respecto al predictor de vector de movimiento. La predicción de los parámetros de codificación y la intrapredicción se pueden denominar colectivamente predicción en imagen.

La Figura 4 muestra un diagrama de bloques de un codificador de vídeo adecuado para emplear realizaciones de la invención. La Figura 4 presenta un codificador para dos capas, pero se apreciará que el codificador presentado podría extenderse similarmente para codificar más de dos capas. La Figura 4 ilustra una realización de un codificador de vídeo que comprende una primera sección 500 de codificador para una capa base y una segunda sección 502 de codificador para una capa de mejora. Cada una de la primera sección 500 de codificador y la segunda sección 502 de codificador puede comprender elementos similares para codificar imágenes entrantes. Las secciones 500, 502 de codificador pueden comprender un predictor 302, 402, de píxeles un codificador 303, 403 de error de predicción y un decodificador 304, 404 de error de predicción. La Figura 4 también muestra una realización del predictor 302, 402 de píxeles que comprende un interpredictor 306, 406, un intrapredictor 308, 408, un selector 310, 410, de modo un filtro 316, 416 y una memoria 318, 418 de fotograma de referencia. El predictor 302 de píxeles de la primera sección 500 de codificador recibe 300 imágenes de capa base de un flujo de vídeo a codificar tanto en el interpredictor 306 (que determina la diferencia entre la imagen y un fotograma 318 de referencia compensado en movimiento) como en el intrapredictor 308 (que determina una predicción para un bloque de imagen basándose solo en las partes ya procesadas del fotograma o imagen actual). Las salidas tanto del interpredictor como del intrapredictor se pasan al selector 310 de modo. El intrapredictor 308 puede tener más de un modo de intrapredicción. Así pues, en cada modo se puede realizar la intrapredicción y proporcionarse la señal predicha al selector 310 de modo. El selector 310 de modo también recibe una copia de la imagen 300 de capa base. Por consiguiente, el predictor 402 de píxeles de la segunda sección 502 de codificador recibe 400 imágenes de capa de mejora de un flujo de vídeo a codificar tanto en el interpredictor 406 (que determina la diferencia entre la imagen y un fotograma 418 de referencia compensado en movimiento) como el intrapredictor 408 (que determina una predicción para un bloque de imagen basándose solo en las partes ya procesadas del fotograma o imagen actual). Las salidas tanto del interpredictor como del intrapredictor se pasan al selector 410 de modo. El intrapredictor 408 puede tener más de un modo de intrapredicción. Así pues, en cada modo se puede realizar la intrapredicción y proporcionarse la señal predicha al selector 410 de modo. El selector 410 de modo también recibe una copia de la imagen 400 de capa de mejora.

5 Dependiendo de qué modo de codificación se seleccione para codificar el bloque actual, la salida del interpredictor 306, 406 o la salida de uno de los modos opcionales intrapredictor o la salida de un codificador de superficie dentro del selector de modo se pasa a la salida del selector 310, 410 de modo. La salida del selector de modo se pasa a un primer dispositivo 321, 421 de suma. El primer dispositivo de suma puede restar la salida del predictor 302, 402 de píxeles de la imagen 300 de capa de base/imagen 400 de capa de mejora para producir una primera señal 320, 420 de error de predicción que se introduce en el codificado 303, 403 de error de predicción.

10 El predictor 302, 402 de píxeles recibe además de un reconstructor preliminar 339, 439 la combinación de la representación de predicción del bloque 312, 412 de imagen y la salida 338, 438 del decodificador 304, 404 de error de predicción. La imagen 314, 414 reconstruida preliminar puede pasarse al intrapredictor 308, 408 y a un filtro 316, 416. El filtro 316, 416 que recibe la representación preliminar puede filtrar la representación preliminar y emitir una imagen 340, 440 reconstruida final que puede guardarse en una memoria 318, 418 de fotograma de referencia. La memoria 318 de fotograma de referencia puede estar conectada al interpredictor 306 para ser utilizada como la imagen de referencia contra la cual se compara una imagen 300 de capa base futura en operaciones de interpredicción. Sujeto a que se seleccione la capa base e indique que es una fuente para la predicción de muestras intercapa y/o predicción de información de movimiento intercapa de la capa de mejora según algunas realizaciones, la memoria 318 de fotograma de referencia también puede estar conectada al interpredictor 406 para usarse como la imagen de referencia contra la que se compara unas imágenes 400 de capa de mejora futuras en operaciones de interpredicción. Además, la memoria 418 de fotograma de referencia puede estar conectada al interpredictor 406 para ser utilizada como la imagen de referencia contra la cual se compara una imagen 400 de capa de mejora futura en operaciones de interpredicción.

20 Los parámetros de filtrado del filtro 316 de la primera sección 500 de codificador pueden proporcionarse a la segunda sección 502 de codificador sometidos a que se seleccione la capa base y se indique para que sea la fuente para predecir los parámetros de filtración de la capa de mejora según algunas realizaciones.

25 El codificador 303, 403 de error de predicción comprende una unidad 342, 442 de transformada y un cuantificador 344, 444. La unidad 342, 442 de transformada transforma la primera señal 320, 420 de error de predicción a un dominio de transformada. La transformada es, por ejemplo, la transformada DCT. El cuantificador 344, 444 cuantifica la señal de dominio de transformada, p. ej., los coeficientes de DCT, para formar coeficientes cuantificados.

30 El decodificador 304, 404 de error de predicción recibe la salida del codificador 303, 403 de error de predicción y realiza los procesos opuestos del codificador 303, 403 de error de predicción para producir una señal 338, 438 de error de predicción decodificada que, cuando se combina con la representación de predicción del bloque 312, 412 de imagen en el segundo dispositivo 339, 439 de suma, produce la imagen 314, 414 reconstruida preliminar. Se puede considerar que el decodificador de error de predicción comprende un descuantificador 361, 461, que descuantifica los valores de coeficientes cuantificados, p. ej., coeficientes de DCT, para reconstruir la señal de transformada y una unidad 363, 463 de transformación inversa, que realiza la transformación inversa a la señal de transformada reconstruida en donde la salida de la unidad 363, 463 de transformación inversa contiene el bloque o bloques reconstruidos. El decodificador de error de predicción también puede comprender un filtro de bloques que puede filtrar el bloque o bloques reconstruidos según información decodificada adicional y parámetros de filtro.

35 El codificador 330, 430 de entropía recibe la salida del codificador 303, 403 de error de predicción y puede realizar una codificación de entropía/codificación de longitud variable adecuada en la señal para proporcionar capacidad de detección y corrección de errores. Las salidas de los codificadores 330, 430 de entropía pueden insertarse en un flujo de bits, p. ej., mediante un multiplexor 508.

40 La codificación/decodificación por entropía se puede realizar de muchas maneras. Por ejemplo, se puede aplicar una codificación/decodificación basada en el contexto, donde tanto el codificador como el decodificador modifican el estado de contexto de un parámetro de codificación basándose en unos parámetros de codificación previamente codificados/decodificados. La codificación basada en el contexto puede ser, por ejemplo, la codificación aritmética binaria adaptativa al contexto (CABAC) o la codificación de longitud variable basada en el contexto (CAVLC) o cualquier codificación por entropía similar. La codificación/decodificación por entropía se puede realizar, alternativa o adicionalmente, usando un plan de codificación de longitud variable, tal como la codificación/decodificación de Huffman o la codificación/decodificación de Exp-Golomb. La decodificación de unos parámetros de codificación procedentes de un flujo de bits codificado por entropía o de unas palabras en clave puede denominarse análisis.

45 El estándar H.264/AVC fue desarrollado por el Joint Video Team (Equipo de Vídeo Mixto - JVT) del Video Coding Experts Group (Grupo de Expertos de Codificación de Vídeo - VCEG) del Sector de Normalización de Telecomunicaciones de la International Telecommunication Union (Unión Internacional de Telecomunicaciones - ITU-T) y el Moving Picture Experts Group (Grupo de Expertos en Imágenes en Movimiento - MPEG) de la International Organisation for Standardization (Organización Internacional para la Estandarización - ISO) / International Electrotechnical Commission (Comisión Electrotécnica Internacional - IEC). El estándar H.264/AVC se publicó tanto por organizaciones de estandarización principales, y se denomina la Recomendación de ITU-T H.264 y el Estándar Internacional ISO/IEC 14496-10, también conocido como MPEG-4 Parte 10 Advanced Video Coding (Codificación de Vídeo Avanzada - AVC). Ha habido múltiples versiones del estándar H.264/AVC, integrando nuevas extensiones o

características a la especificación. Estas extensiones incluyen Scalable Video Coding (Codificación de Vídeo Escalable - SVC) y Multiview Video Coding (Codificación de Vídeo de Múltiples Vistas - MVC).

5 La versión 1 del estándar de codificación de Vídeo de Alta Eficiencia (H.265/HEVC) fue desarrollado por el Joint Collaborative Team - Video Coding (Equipo Colaborativo Mixto - Codificación de Vídeo - JCT-VC) de VCEG y MPEG. El estándar fue publicado por ambas organizaciones de estandarización principales, y se denomina Recomendación de ITU-T H.265 y el Estándar Internacional ISO/IEC 23008-2, también conocido como MPEG-H Parte 2 de High Efficiency Video Coding (Codificación de Vídeo de Alta Eficiencia - HEVC). Las versiones posteriores de H.265/HEVC incluyeron unas ampliaciones escalables, multivista, de intervalo de fidelidad y unas ampliaciones de codificación 10 tridimensional y de contenido de pantalla, que pueden abreviarse como SHVC, MV-HEVC, REXT, 3D-HEVC y SCC, respectivamente.

15 La SHVC, la MV-HEVC y la 3D-HEVC utilizan una especificación de base común, especificada en el anexo F de la versión 2 del estándar de HEVC. Este fundamento común comprende, por ejemplo, una sintaxis y una semántica de alto nivel que, por ejemplo, especifican algunas de las características de las capas del flujo de bits, tales como las dependencias intercapa, así como unos procesos de decodificación, tales como una construcción de listas de imágenes de referencia que incluyen unas imágenes de referencia intercapa y una obtención de recuento de orden de imágenes para un flujo de bits multicapa. El anexo F también se puede usar en posibles ampliaciones multicapa 20 posteriores de la HEVC. Cabe entender que, aunque en lo que sigue se pueden describir un codificador de vídeo, un decodificador de vídeo, unos métodos de codificación, unos métodos de decodificación, unas estructuras de flujo de bits y/o unas realizaciones haciendo referencia a unas ampliaciones específicas, tales como la SHVC y/o la MV-HEVC, generalmente son aplicables a cualquier ampliación multicapa de la HEVC e, incluso más generalmente, a cualquier plan de codificación de vídeo multicapa.

25 Algunas definiciones clave, secuencia de bits y estructuras de codificación, y conceptos de H.264/AVC y HEVC se describen en esta sección como un ejemplo de un codificador de vídeo, decodificador, método de codificación, método de decodificación y una estructura de flujo de bits, en donde pueden implementarse las realizaciones. Algunas de las definiciones clave, secuencia de bits y estructuras de codificación, y conceptos de H.264/AVC son las mismas que en HEVC, por lo tanto, se describen a continuación conjuntamente. Los aspectos de la invención no se limitan a 30 H.264/AVC o HEVC, sino más bien la descripción se proporciona para una base posible sobre la que la invención puede estar parcial o totalmente realizada.

35 Similarmente a muchos estándares de codificación de vídeo anteriores, la sintaxis y la semántica de flujo de bits, así como el proceso de decodificación para flujos de bits sin de errores se especifican en H.264/AVC y HEVC. El proceso de codificación no se especifica, pero los codificadores deben generar flujos de bits conformes. La conformidad con el flujo de bits y el decodificador se puede verificar con el Hypothetical Reference Decoder (Decodificador de Referencia Hipotético - HRD). Los estándares contienen herramientas de codificación que ayudan a hacer frente a errores y pérdidas de transmisión, pero el uso de las herramientas en la codificación es opcional y no se ha especificado ningún proceso de decodificación para flujos de bits erróneos.

40 La unidad elemental para la entrada a un codificador H.264/AVC o HEVC y la salida de un decodificador H.264/AVC o HEVC, respectivamente, es una imagen. Una imagen dada como entrada a un codificador también puede denominarse imagen de origen, y una imagen decodificada por un decodificador puede denominarse como una imagen decodificada.

45 Las imágenes originarias y decodificadas comprenden cada una una o más matrices de muestras, tal como uno de los siguientes conjuntos de matrices de muestra:

- 50 – Únicamente luma (Y) (monocromo).
- Luma y dos croma (YCbCr o YCgCo).
- Verde, azul y Rojo (GBR, también conocido como RGB).
- 55 – Matrices que representan otros muestreos de color monocromático o tri-estímulo no especificados (por ejemplo, YZX, también conocido como XYZ).

60 A continuación, estas matrices pueden denominarse luma (o L o Y) y croma, donde las dos matrices de croma pueden denominarse Cb y Cr; independientemente del método de representación de color real en uso. El método de representación de color real en uso puede indicarse, p. ej., en un flujo de bits codificado, por ejemplo, usando la sintaxis de Video Usability Information (Información de Usabilidad de Vídeo - VUI) de H.264/AVC y/o HEVC. Un componente puede definirse como una matriz o una muestra única de una de las tres matrices de matrices de muestras (luma y dos croma) o la matriz o una única muestra de la matriz que componen una imagen en formato monocromo.

65 En H.264/AVC y HEVC, una imagen puede ser un fotograma o un campo. Un fotograma comprende una matriz de muestras de luma y posiblemente las muestras de croma correspondientes. Un campo es un conjunto de filas de

muestras alternativas de un fotograma y puede usarse como entrada de codificador, cuando la señal de origen está entrelazada. Las matrices de muestras de croma pueden estar ausentes (y, por lo tanto, el muestreo monocromático puede estar en uso) o las matrices de muestras de croma pueden submuestrearse cuando se comparan con las matrices de muestras de luma. Los formatos de croma pueden resumirse de la siguiente manera:

– En el muestreo monocromo hay solo una matriz de muestras, que puede considerarse nominalmente la matriz de luma.

– En el muestreo 4:2:0, cada una de las dos matrices de croma tiene la mitad de la altura y la mitad del ancho de la matriz de luma.

– En el muestreo 4:2:2, cada una de las dos matrices de croma tiene la misma altura y la mitad de la anchura de la matriz de luma.

– En el muestreo 4:4:4 cuando no hay planos de color separados en uso, cada una de las dos matrices de croma tiene la misma altura y anchura que la matriz de luma.

En H.264/AVC y HEVC, es posible codificar matrices de muestras como planos de color separados en el flujo de bits y decodificar respectivamente planos de color codificados por separado a partir del flujo de bits. Cuando se usan planos de color separados, cada uno de ellos se procesa por separado (por el codificador y/o el decodificador) como una imagen con muestreo monocromo.

Se puede definir una partición como una división de un conjunto en subconjuntos de modo que cada elemento del conjunto esté en exactamente uno de los subconjuntos.

Cuando se describe la operación de codificación y/o decodificación de HEVC, se pueden usar los siguientes términos. Un bloque de codificación puede definirse como un bloque NxN de muestras para cierto valor de N de tal modo que la división de un bloque de árbol de codificación en bloques de codificación es una partición. Un bloque de árbol de codificación (CTB) puede definirse como un bloque de NxN de muestras para cierto valor de N, de modo que la división de un componente en bloques de árbol de codificación es una partición. Una unidad de árbol de codificación (CTU) puede definirse como un bloque de árbol de codificación de muestras de luma, dos bloques de árbol de codificación correspondientes de muestras de croma de una imagen que tiene tres matrices de muestras, o un bloque de árbol de codificación de muestras de una imagen monocromo o una imagen que está codificada usando tres planos de color separados y estructuras de sintaxis usadas para codificar las muestras. Una unidad de codificación (CU) puede definirse como un bloque de codificación de muestras de luma, dos bloques de codificación correspondientes de muestras de croma de una imagen que tiene tres matrices de muestras, o un bloque de codificación de muestras de una imagen monocromo o una imagen que está codificada usando tres planos de color separados y estructuras de sintaxis usadas para codificar las muestras. Una CU con el tamaño máximo permitido se puede denominar LCU (largest coding unit - unidad de codificación más grande) o unidad de árbol de codificación (CTU) y la imagen de vídeo se divide en LCU no superpuestas.

Una CU consiste en una o más unidades de predicción (PU) que definen el proceso de predicción para las muestras dentro de la CU y una o más unidades de transformada (TU) que definen el proceso de codificación de error de predicción para las muestras en dicha CU. Típicamente, una CU consiste en un bloque cuadrado de muestras con un tamaño seleccionable de un conjunto predefinido de posibles tamaños de CU. Cada PU y TU se pueden dividir de forma adicional en PU y TU más pequeñas para aumentar la granularidad de los procesos de predicción y de codificación de errores de predicción, respectivamente. Cada PU tiene información de predicción asociada con ella que define qué tipo de predicción se debe aplicar para los píxeles dentro de esa PU (p. ej., información de vector de movimiento para PU interpredichas e información de direccionalidad de intrapredicción para PU intrapredichas).

Cada TU está asociada a una información que describe el proceso de decodificación de error de predicción para las muestras que están dentro de dicha TU (incluida, p. ej., una información de coeficiente de DCT). Normalmente se señala a nivel de CU si se aplica o no una codificación de error de predicción para cada CU. En caso de que no haya ningún error de predicción residual asociado con la CU, se puede considerar que no hay ninguna TU para dicha CU. La división de la imagen en las CU, y la división de las CU en PU y TU se indica típicamente en el flujo de bits que permite que el decodificador reproduzca la estructura prevista de estas unidades.

En HEVC, una imagen puede dividirse en piezas, que son rectangulares y contienen un número entero de LCU. En HEVC, la partición en piezas forma una cuadrícula regular, donde las alturas y anchuras de las piezas difieren entre sí por una LCU como máximo. En HEVC, se define un corte que es un número entero de unidades de árbol de codificación contenidas en un segmento de corte independiente y todos los segmentos de corte dependientes posteriores (si los hay) que preceden al siguiente segmento de corte independiente (si lo hay) dentro de la misma unidad de acceso. En HEVC, un segmento de corte se define como un número entero de unidades de árbol de codificación ordenadas consecutivamente en la exploración de piezas y contenidas en una única unidad de NAL. La división de cada imagen en segmentos de corte es una partición. En HEVC, un segmento de corte independiente se define como un segmento de corte para el cual los valores de los elementos de sintaxis de la cabecera de segmento

de corte no se infieren de los valores para un segmento de corte anterior, y un segmento de corte dependiente se define como un segmento de corte para el cual los valores de algunos elementos de sintaxis de la cabecera de segmento de corte se infieren de los valores para el segmento de corte independiente anterior en orden de decodificación. En HEVC, una cabecera de corte se define como la cabecera del segmento de corte del segmento de corte independiente que es un segmento de corte actual o es el segmento de corte independiente que precede a un segmento de corte dependiente actual, y una cabecera de segmento de corte se define como una parte de un segmento de corte codificado que contiene los elementos de datos pertenecientes a la primera o todas las unidades de árbol de codificación representadas en el segmento de corte. Las CU se exploran en el orden de barrido por tramas de las LCU dentro de las piezas o dentro de una imagen, si no se usan piezas. Dentro de una LCU, las CU tienen un orden de exploración específico.

Un conjunto de piezas constreñido en cuanto al movimiento (MCTS) es tal que el proceso de interpredicción está constreñido en cuanto a la codificación, de tal modo que no se usa ningún valor de muestra que esté fuera del conjunto de piezas constreñido en cuanto al movimiento ni ningún valor de muestra que esté en una posición de muestra fraccionaria que se obtenga usando uno o más valores de muestra que estén fuera del conjunto de piezas constreñido en cuanto al movimiento para la interpredicción de cualquier muestra que esté dentro del conjunto de piezas constreñido en cuanto al movimiento. Además, la codificación de un MCTS está constreñida de una manera tal que unos candidatos a vector de movimiento no se obtienen de unos bloques que están fuera del MCTS. Esto se puede imponer desactivando la predicción de vectores de movimiento temporal de la HEVC o impidiendo que el codificador utilice el candidato de TMVP o cualquier candidato de predicción de vectores de movimiento que venga después del candidato de TMVP en la lista de fusión o de candidatos de AMVP para PU que estén ubicadas directamente a la izquierda del límite derecho de pieza del MCTS, salvo el último que está en la parte inferior derecha del MCTS. En general, un MCTS puede definirse como que es un conjunto de piezas que es independiente de cualquier valor de muestra y dato codificado, tal como un vector de movimiento, que esté fuera del MCTS. En algunos casos, es posible que se requiera que un MCTS forme una zona rectangular. Debe entenderse que, dependiendo del contexto, un MCTS puede referirse al conjunto de piezas que está dentro de una imagen o al respectivo conjunto de piezas que hay en una secuencia de imágenes. El respectivo conjunto de piezas puede estar, aunque en general no necesita estarlo, coubicado en la secuencia de imágenes.

Obsérvese que el proceso de codificación y/o decodificación puede saturar las ubicaciones de muestra utilizadas en la interpredicción, de modo que una ubicación, que de otro modo estaría fuera de la imagen, se sature para apuntar a la correspondiente muestra límite de la imagen. Por lo tanto, si un límite de pieza también es un límite de imagen, en algunos casos de uso, los codificadores pueden permitir que unos vectores de movimiento atraviesen ese límite de manera efectiva o que un vector de movimiento provoque de manera efectiva una interpolación de muestra fraccionaria que se referiría a una ubicación fuera de ese límite, puesto que las ubicaciones de muestra se saturan sobre el límite. En otros casos de uso, específicamente si puede extraerse una pieza codificada de un flujo de bits donde está ubicada en una posición que es adyacente a un límite de imagen a otro flujo de bits donde la pieza está ubicada en una posición que no es adyacente al límite de imagen, los codificadores pueden constreñir los vectores de movimiento en los límites de imagen de manera similar a cualquier límite de MCTS.

El mensaje SEI de conjuntos de piezas constreñidos en cuanto al movimiento temporal de la HEVC se puede usar para indicar la presencia de conjuntos de piezas constreñidos en cuanto al movimiento en el flujo de bits.

El decodificador reconstruye el vídeo de salida aplicando medios de predicción similares al codificador para formar una representación predicha de los bloques de píxeles (usando el movimiento o la información espacial creada por el codificador y almacenada en la representación comprimida) y la decodificación de error de predicción (operación inversa de la codificación de error de predicción recuperando la señal de error de predicción cuantificada en el dominio de píxeles espaciales). Después de aplicar la predicción y la decodificación de errores de predicción, el decodificador suma las señales de predicción y de error de predicción (valores de píxel) para formar el fotograma de vídeo de salida. El decodificador (y codificador) también puede aplicar medios de filtrado adicionales para mejorar la calidad del vídeo de salida antes de pasar por su visualización y/o almacenarlo como referencia de predicción para los fotogramas próximos en la secuencia de vídeo.

El filtrado puede incluir, por ejemplo, uno más de lo siguientes: desbloqueo, desplazamiento adaptativo de muestras (SAO) y/o filtrado de bucle adaptativo (ALF). H.264/AVC incluye un desbloqueo, mientras que HEVC incluye tanto desbloqueo como SAO.

En los códecs de vídeo típicos, la información de movimiento se indica con vectores de movimiento asociados con cada bloque de imagen con compensación de movimiento, tal como una unidad de predicción. Cada uno de estos vectores de movimiento representa el desplazamiento del bloque de imagen en la imagen a codificar (en el lado del codificador) o a decodificar (en el lado del decodificador) y el bloque de origen de predicción en una de las imágenes previamente codificadas o decodificadas. Para representar vectores de movimiento de manera eficiente, aquellos que típicamente se codifican diferencialmente con respecto a vectores de movimiento predichos específicos de bloque. En códecs de vídeo típicos, los vectores de movimiento predichos se crean de una manera predefinida, por ejemplo, calculando la mediana de los vectores de movimiento codificados o decodificados de los bloques adyacentes. Otra forma de crear predicciones de vectores de movimiento es generar una lista de predicciones candidatas de bloques

adyacentes y/o bloques coubicados en imágenes de referencia temporal y señalar el candidato elegido como el predictor de vector de movimiento. Además de predecir los valores del vector de movimiento, puede predecirse qué imagen o imágenes de referencia se usan para la predicción compensada por movimiento y esta información de predicción puede representarse, por ejemplo, mediante un índice de referencia de imagen previamente codificada/decodificada. El índice de referencia se predice típicamente desde bloques adyacentes y/o bloques coubicados en la imagen de referencia temporal. Además, los códecs de vídeo de alta eficiencia típicos emplean un mecanismo de codificación/decodificación de información de movimiento adicional, a menudo llamado modo de fusión/fusionar, donde toda la información de campo de movimiento, que incluye el vector de movimiento y el índice de imagen de referencia correspondiente para cada lista de imágenes de referencia disponible, se predice y se usa sin ninguna modificación/corrección. Similarmente, se lleva a cabo la predicción de la información de campo de movimiento usando la información de campo de movimiento de bloques adyacentes y/o bloques coubicados en imágenes de referencia temporal y se señala la información de campo de movimiento usada entre una lista de la lista de candidatos de campo de movimiento rellena con información de campo de movimiento de bloques adyacentes/coubicados disponibles.

En códecs de vídeo típicos, el residuo de predicción después de la compensación del movimiento se transforma en primer lugar con un núcleo de transformada (como DCT) y, a continuación, se codifica. La razón de esto es que, a menudo, existe alguna correlación entre el residuo y la transformada que, en muchos casos, puede ayudar a reducir esta correlación y proporcionar una codificación más eficiente.

Los codificadores de vídeo típicos utilizan funciones de coste lagrangianas para encontrar unos modos de codificación óptimos, p. ej., el modo de codificación deseado para un bloque y unos vectores de movimiento asociados. Este tipo de función de coste usa un factor λ de ponderación para vincular la distorsión de imagen (exacta o estimada) debido a los métodos de codificación con pérdidas y la cantidad (exacta o estimada) de información que se requiere para representar los valores de píxel en un área de imagen:

$$C = D + \lambda R, \quad (1)$$

donde C es el coste de Lagrange que se va a minimizar, d es la distorsión de imagen (p. ej., error cuadrático medio) con los vectores de modo y movimiento considerados, y R el número de bits necesarios para representar los datos requeridos para reconstruir el bloque de imagen en el decodificador (incluida la cantidad de datos para representar los vectores de movimiento candidatos).

Los estándares y especificaciones de codificación de vídeo pueden permitir que los codificadores dividan una imagen codificada a cortes codificados o similares. La predicción en imagen está deshabilitada típicamente a través de los límites de corte. Por lo tanto, los cortes pueden considerarse una forma de dividir una imagen codificada a piezas que pueden decodificarse de manera independiente. En H.264/AVC y HEVC, la predicción en imagen puede desactivarse a través de los límites de corte. Por lo tanto, los cortes se pueden considerar como una forma de dividir una imagen codificada en piezas que pueden decodificarse de manera independiente, y los cortes, por lo tanto, a menudo se consideran unidades elementales para la transmisión. En muchos casos, los codificadores pueden indicar en el flujo de bits qué tipos de predicción en imagen se desconectan a través de límites de corte, y la operación del decodificador tiene en cuenta esta información, por ejemplo, cuando se concluye qué fuentes de predicción están disponibles. Por ejemplo, puede considerarse que las muestras de una CU vecina no están disponibles para la intrapredicción si la CU vecina reside en un corte diferente.

Una unidad elemental para la salida de un codificador H.264/AVC o HEVC y la entrada de un decodificador H.264/AVC o HEVC, respectivamente, es una unidad de Capa de Abstracción de Red (NAL). Para el transporte sobre redes orientadas por paquetes o el almacenamiento en archivos estructurados, las unidades NAL pueden encapsularse en paquetes o estructuras similares. Se ha especificado un formato de flujo de bytes en H.264/AVC y HEVC para entornos de transmisión o almacenamiento que no proporcionan estructuras de alineación de tramas. El formato de flujo de bits separa las unidades NAL entre sí anexando un código de inicio delante de cada unidad NAL. Para evitar la falsa detección de límites de unidad NAL, los codificadores ejecutan un algoritmo de prevención de emulación de código de inicio orientado a bytes, que añade un byte de prevención de emulación a la carga útil de unidad NAL si se hubiera producido un código de inicio de otra manera. Con el fin de permitir una operación de puerta de enlace sencilla entre sistemas orientados a paquetes y a flujos, la prevención de emulación de código de inicio siempre puede realizarse independientemente de si el formato de flujo de bytes está en uso o no. Una unidad NAL puede definirse como una estructura de sintaxis que contiene una indicación del tipo de datos a seguir y bytes que contienen ese dato en forma de una RBSP intercalada según sea necesario con bytes de prevención de emulación. Una carga útil de secuencia de bytes sin procesar (RBSP) puede definirse como una estructura de sintaxis que contiene un número entero de bytes que están encapsulados en una unidad NAL. Una RBSP está vacía o tiene la forma de una cadena de bits de datos que contienen elementos de sintaxis seguidos de un bit de parada de RBSP y seguido de cero o más bits posteriores iguales a 0.

Las unidades NAL consisten en una cabecera y una carga útil. En H.264/AVC y la HEVC, la cabecera de unidad NAL indica el tipo de la unidad NAL

En HEVC, se usa una cabecera de unidad NAL de dos bytes para todos los tipos de unidad NAL especificados. La cabecera de la unidad NAL contiene un bit reservado, una indicación de tipo de unidad NAL de seis bits, una indicación nuh_temporal_id_plus1 de tres bits para el nivel temporal (puede ser necesario que sea mayor o igual a 1) y un elemento de sintaxis nuh_layer_id de seis bits. El elemento de sintaxis temporal_id_plus1 puede considerarse como un identificador temporal para la unidad NAL, y una variable TemporalId basada en cero puede derivarse de la siguiente manera: $\text{TemporalId} = \text{temporal_id_plus1} - 1$. La abreviatura TID puede usarse para indistintamente con la variable TemporalId. TemporalId igual a 0 corresponde al nivel temporal más bajo. Se requiere que el valor de temporal_id_plus1 sea distinto de cero para evitar la emulación de código de inicio que implica los dos bytes de cabecera de unidad NAL. El flujo de bits creado excluyendo todas las unidades NAL de VCL que tienen un TemporalId mayor o igual a un valor seleccionado y que incluyen todas las demás unidades NAL de VCL permanece conforme. En consecuencia, una imagen que tenga un TemporalId igual a tid_value no utiliza ninguna imagen que tenga un TemporalId mayor que tid_value como referencia de interpredicción. Una subcapa o una subcapa temporal puede definirse como una capa temporal escalable (o una capa temporal, TL) de un flujo de bits temporal escalable, que consta de unidades VCL NAL con un valor particular de la variable TemporalId y las unidades NAL no-VCL asociadas. nuh_layer_id puede entenderse como un identificador de capa de escalabilidad.

Las unidades NAL pueden clasificarse en unidades NAL de Capa de Codificación de Vídeo (VCL) y unidades NAL no VCL. Las unidades NAL de VCL son típicamente unidades NAL de corte codificado. En HEVC, unidades NAL de VCL contienen elementos de sintaxis que representan una o más CU.

En HEVC, las abreviaturas para los tipos de imágenes pueden definirse como sigue: Temporal Sub-layer Access (imagen posterior - (TRAIL), Step-wise Temporal Sub-layer Access (acceso de subcapa temporal - TSA), Step-wise Temporal Sub-layer Access (acceso de subcapa temporal por etapas - STSA), Random Access Decodable Leading (imagen principal decodificable de acceso aleatorio - RADL), Random Access Skipped Leading (imagen principal de acceso aleatorio saltado - RASL), picture, Broken Link Access (imagen de acceso de enlace roto - BLA), Instantaneous Decoding Refresh (imagen de refresco de decodificación instantáneo - IDR), Clean Random Access (imagen de acceso aleatorio limpio - CRA).

Una imagen de punto de acceso aleatorio (RAP), que también puede denominarse imagen de punto de acceso intra aleatorio (IRAP) en una capa independiente contiene solo cortes intracodificados. Una imagen IRAP que pertenece a una capa predicha puede contener unos cortes P, B y I, no puede usar la interpredicción de otras imágenes en la misma capa predicha y puede usar una predicción intercapa de sus capas de referencia directa. En la presente versión de HEVC, una imagen IRAP puede ser una imagen BLA, una imagen CRA o una imagen IDR. La primera imagen en un flujo de bits que contiene una capa base es una imagen IRAP en la capa base. Siempre que los conjuntos de parámetros necesarios estén disponibles cuando sea necesario activarlos, una imagen IRAP en una capa independiente y todas las imágenes no RASL posteriores en la capa independiente en orden de decodificación se pueden decodificar correctamente sin realizar el proceso de decodificación de ninguna imagen que preceda a la Imagen IRAP en orden de decodificación. La imagen IRAP que pertenece a una capa predicha y todas las imágenes no RASL posteriores en orden de decodificación dentro de la misma capa predicha pueden decodificarse correctamente sin realizar el proceso de decodificación de cualquier imagen de la misma capa predicha que preceda a la imagen IRAP en orden de decodificación, cuando los conjuntos de parámetros necesarios están disponibles cuando deben activarse y cuando se ha inicializado la decodificación de cada capa de referencia directa de la capa predicha. Puede haber imágenes en un flujo de bits que contengan solo cortes intracodificados que no sean imágenes IRAP.

Una unidad NAL que no es VCL puede ser, por ejemplo, uno de los siguientes tipos: un conjunto de parámetros de secuencia, un conjunto de parámetros de imagen, una unidad NAL de información de mejora suplementaria (SEI), un delimitador de unidad de acceso, una unidad NAL de fin de secuencia, una unidad NAL de fin de flujo de bits o una unidad NAL de datos de relleno. Pueden ser necesarios conjuntos de parámetros para la reconstrucción de imágenes decodificadas, mientras que muchas de las otras unidades NAL no de VCL no son necesarias para la reconstrucción de valores de muestra decodificada.

Los parámetros que permanecen sin cambios a través de una secuencia de vídeo codificado pueden incluirse en un conjunto de parámetros de secuencia. Además de los parámetros que pueden ser necesarios por el proceso de decodificación, el conjunto de parámetros de secuencia puede contener opcionalmente información de usabilidad de vídeo (VUI), que incluye parámetros que pueden ser importantes para el almacenamiento en memoria intermedia, la temporización de salida de imagen, la representación y la reserva de recursos. En HEVC, un RBSP de conjunto de parámetros de secuencia incluye parámetros que pueden denominarse por uno o más RBSP de conjunto de parámetros de imagen o una o más unidades NAL SEI que contienen un mensaje SEI de período de almacenamiento en memoria intermedia. Un conjunto de parámetros de imagen contiene tales parámetros que probablemente no se cambien en varias imágenes codificadas. Un RBSP de conjunto de parámetros de imagen puede incluir parámetros que pueden denominarse por las unidades NAL de corte codificado de una o más imágenes codificadas.

En HEVC, un conjunto de parámetros de vídeo (VPS) puede definirse como una estructura de sintaxis que contiene elementos de sintaxis que se aplican a cero o más secuencias de vídeo codificado completas según lo determinado por el contenido de un elemento de sintaxis que se encuentra en el SPS al que se hace referencia por un elemento de

sintaxis que se encuentra en el PPS al que se hace referencia por un elemento de sintaxis encontrado en cada cabecera de segmento de corte.

5 Un RBSP de conjunto de parámetros de vídeo puede incluir parámetros que pueden referenciarse por uno o más RBSP de conjunto de parámetros de secuencia.

10 La relación y jerarquía entre el conjunto de parámetros de vídeo (VPS), el conjunto de parámetros de secuencia (SPS) y el conjunto de parámetros de imagen (PPS) puede describirse como sigue. VPS reside un nivel por encima del SPS en la jerarquía de conjunto de parámetros y en el contexto de escalabilidad y/o vídeo 3D. El VPS puede incluir parámetros que son comunes para todos los cortes en todas las capas (escalabilidad o vista) en toda la secuencia de vídeo codificado. SPS incluye los parámetros que son comunes para todos los cortes en una capa particular (escalabilidad o vista) en toda la secuencia de vídeo codificado, y pueden compartirse por múltiples capas (escalabilidad o vista). El PPS incluye los parámetros que son comunes para todos los cortes en una representación de capa particular (la representación de una capa de escalabilidad o de vista en una unidad de acceso) y es probable que se compartan por todos los cortes en múltiples representaciones de capa.

15 VPS puede proporcionar información acerca de las relaciones de dependencia de las capas en un flujo de bits, así como mucha otra información que es aplicable a todos los cortes en todas las capas (escalabilidad o vista) en toda la secuencia de vídeo codificado. Se puede considerar que VPS comprende dos partes, el VPS base y una extensión VPS, donde la extensión VPS puede estar opcionalmente presente.

20 La transmisión, señalización o almacenamiento fuera de banda puede usarse adicional o alternativamente para otros fines que la tolerancia contra errores de transmisión, tales como facilidad de acceso o negociación de sesión. Por ejemplo, una entrada de muestra de una pista en un archivo conforme al formato ISO Base Media File puede comprender conjuntos de parámetros, mientras que los datos codificados en el flujo de bits se almacenan en otra parte del archivo o en otro archivo. La frase a lo largo del flujo de bits (p. ej., que indica a lo largo del flujo de bits) o a lo largo de una unidad codificada de un flujo de bits (p. ej., que indica a lo largo de una pieza codificada) puede usarse en unas reivindicaciones y en unas realizaciones descritas para referirse a una transmisión, una señalización o un almacenamiento fuera de banda de una manera tal que los datos fuera de banda estén asociados con el flujo de bits o la unidad codificada, respectivamente. La decodificación de frase a lo largo del flujo de bits o a lo largo de una unidad codificada de un flujo de bits o similar puede referirse a decodificar los referidos datos fuera de banda (que pueden obtenerse de una transmisión, una señalización o un almacenamiento fuera de banda) que están asociados con el flujo de bits o la unidad codificada, respectivamente.

25 Una unidad NAL SEI puede contener uno o más mensajes SEI, que no son necesarios para la decodificación de imágenes de salida, pero pueden ayudar en procesos relacionados, tales como la temporización de salida de imágenes, la representación, la detección de errores, la ocultación de errores y la reserva de recursos. Se especifican varios mensajes SEI en H.264/AVC y HEVC, y los mensajes SEI de datos de usuario permiten a las organizaciones y empresas especificar mensajes SEI para su propio uso. H.264/AVC y HEVC contienen la sintaxis y la semántica para los mensajes SEI especificados, pero no se define ningún proceso para manejar los mensajes en el receptor. Por consiguiente, se requieren codificadores para seguir el estándar H.264/AVC o el estándar HEVC cuando crean mensajes SEI, y decodificadores conformes al estándar H.264/AVC o al estándar HEVC, respectivamente, no se requieren para procesar los mensajes SEI para la conformidad del orden de salida. Una de las razones para incluir la sintaxis y la semántica de los mensajes SEI en H.264/AVC y HEVC es permitir que diferentes especificaciones del sistema interpreten la información complementaria de manera idéntica y, por lo tanto, interfundan. Se pretende que las especificaciones del sistema puedan requerir el uso de mensajes SEI particulares tanto en el extremo de codificación como en el extremo de decodificación y, de forma adicional, se puede especificar el proceso para manejar mensajes SEI particulares en el receptor.

30 En la HEVC hay dos tipos de unidades NAL SEI, a saber, la unidad NAL SEI de sufijo y la unidad NAL SEI de prefijo, que tienen un valor `nal_unit_type` diferente la una de la otra. Los mensajes SEI contenidos en una unidad NAL SEI de sufijo están asociados a la unidad NAL VCL que precede, en orden de decodificación, a la unidad de NAL SEI de sufijo. El (los) mensaje(s) SEI contenido(s) en una unidad NAL SEI de prefijo están asociados a la unidad NAL VCL que va después, en orden de decodificación, de la unidad NAL SEI de prefijo.

35 Una imagen codificada es una representación codificada de una imagen.

40 En HEVC, una imagen codificada puede definirse como una representación codificada de una imagen que contiene todas las unidades de árbol de codificación de la imagen. En HEVC, una unidad de acceso (AU) se puede definir como un conjunto de unidades NAL que están asociadas entre sí según una regla de clasificación específica, son consecutivas en el orden de decodificación y contienen como máximo una imagen con cualquier valor específico de `nuh_layer_id`. Además de contener las unidades NAL VCL de la imagen codificada, una unidad de acceso también puede contener unidades NAL no VCL. Dicha regla de clasificación especificada puede asociar, por ejemplo, imágenes con el mismo valor de conteo de salida de imagen o de salida de imagen en la misma unidad de acceso.

65

- Un flujo de bits puede definirse como una secuencia de bits, en forma de un flujo de unidades NAL o un flujo de bytes, que forma la representación de imágenes codificadas y datos asociados que forman una o más secuencias de vídeo codificadas. Un primer flujo de bits puede ser seguido por un segundo flujo de bits en el mismo canal lógico, tal como en el mismo archivo o en la misma conexión de un protocolo de comunicación. Un flujo elemental (en el contexto de la codificación de vídeo) puede definirse como una secuencia de uno o más flujos de bits. El final del primer flujo de bits puede indicarse mediante una unidad NAL específica, que puede denominarse unidad NAL de final de flujo de bits (EOB) y que es la última unidad NAL del flujo de bits. En la HEVC y sus actuales borradores de ampliación, se requiere que la unidad NAL de EOB tenga `nuh_layer_id` igual a 0.
- En H.264/AVC, una secuencia de vídeo codificado se define como una secuencia de unidades de acceso consecutivas en orden de decodificación desde una unidad de acceso IDR, inclusive, a la siguiente unidad de acceso IDR, exclusiva, o al final del flujo de bits, lo que aparezca antes.
- En HEVC, una secuencia de vídeo codificado (CVS) se puede definir, por ejemplo, como una secuencia de unidades de acceso que consiste, en orden de decodificación, de una unidad de acceso de IRAP con `NoRasIOutputFlag` igual a 1, seguida por cero o más unidades de acceso que no son unidades de acceso IRAP con `NoRasIOutputFlag` igual a 1, que incluyen todas las unidades de acceso posteriores hasta, pero no incluida ninguna unidad de acceso posterior que sea una unidad de acceso IRAP con `NoRasIOutputFlag` igual a 1. Una unidad de acceso de IRAP puede definirse como una unidad de acceso en la que la imagen de la capa base es una imagen de IRAP. El valor de `NoRasIOutputFlag` es igual a 1 para cada imagen IDR y cada imagen BLA, y cada imagen IRAP que sea la primera imagen en esa capa particular en el flujo de bits en orden de decodificación es la primera imagen IRAP que va después de una unidad NAL de final de secuencia que tiene el mismo valor de `nuh_layer_id` en orden de decodificación. Puede haber medios para proporcionar el valor de `HandleCraAsBlaFlag` al decodificador desde una entidad externa, tal como un reproductor o un receptor, que puede controlar el decodificador. `HandleCraAsBlaFlag` puede establecerse a 1, por ejemplo, por un reproductor que busca una nueva posición en un flujo de bits o sintoniza en una difusión y comienza a decodificar y, a continuación, comienza a decodificar desde una imagen CRA. Cuando `HandleCraAsBlaFlag` es igual a 1 para una imagen CRA, la imagen CRA se maneja y decodifica como si fuera una imagen BLA.
- En HEVC, una secuencia de vídeo codificada puede especificarse adicional o alternativamente (a la memoria descriptiva anterior) para finalizar, cuando una unidad NAL específica, que puede denominarse unidad NAL de extremo de secuencia (EOS - *end of sequence*), aparece en el flujo de bits y tiene un `nuh_layer_id` igual a 0.
- Un grupo de imágenes (GOP - *group of pictures*) y sus características pueden definirse como sigue. Un GOP puede decodificarse independientemente de si se descodificaron cualesquiera imágenes anteriores. Un GOP abierto es un grupo de imágenes en el que las imágenes que preceden a la imagen intra inicial en orden de salida podrían no decodificarse correctamente cuando la decodificación comienza a partir de la imagen intra inicial del GOP abierto. En otras palabras, las imágenes de un GOP abierto pueden referirse (en la interpredicción) a imágenes que pertenecen a un GOP anterior. Un decodificador HEVC puede reconocer una imagen intra que inicia un GOP abierto, porque un tipo de unidad de NAL específica, tipo de unidad de NAL de CRA, puede usarse para sus cortes codificados. Un GOP cerrado es un grupo de imágenes en el que todas las imágenes pueden decodificarse correctamente cuando la decodificación comienza a partir de la imagen intra inicial del GOP cerrado. En otras palabras, ninguna imagen en un GOP cerrado se refiere a cualquier imagen en los GOP anteriores. En H.264/AVC y HEVC, un GOP cerrado puede comenzar a partir de una imagen IDR. En HEVC, un GOP cerrado también puede comenzar a partir de una imagen BLA_W_RADL o BLA_N_LP. Una estructura de codificación de GOP abierto es potencialmente más eficiente en la compresión en comparación con una estructura de codificación de GOP cerrada, debido a una mayor flexibilidad en la selección de imágenes de referencia.
- Una Decoded Picture Buffer (memoria intermedia de imágenes decodificadas - DPB) puede usarse en el codificador y/o en el decodificador. Hay dos razones para almacenar en memoria intermedia imágenes decodificadas, para referencias en interpredicción y para reordenar imágenes decodificadas en orden de salida. Como H.264/AVC y HEVC proporcionan una gran flexibilidad tanto para el marcaje de imagen de referencia como para la reordenación de salida, las memorias intermedias separadas para el almacenamiento en memoria intermedia de imagen de referencia y el almacenamiento en memoria intermedia de imagen de salida pueden desperdiciar recursos de memoria. Por lo tanto, la DPB puede incluir un proceso de almacenamiento en memoria intermedia de imágenes decodificadas unificado para las imágenes de referencia y la reordenación de salida. Una imagen decodificada puede retirarse de la DPB cuando ya no se usa como referencia y no se necesita para la salida.
- En muchos modos de codificación de H.264/AVC y HEVC, la imagen de referencia para la interpredicción se indica con un índice a una lista de imágenes de referencia. El índice puede codificarse con codificación de longitud variable, que habitualmente provoca que un índice más pequeño tenga un valor más corto para el elemento de sintaxis correspondiente. En H.264/AVC y HEVC, se generan dos listas de imágenes de referencia (la lista 0 de imágenes de referencia y la lista 1 de imágenes de referencia) para cada corte bipredictivo (B), y se forma una lista de imágenes de referencia (la lista 0 de imágenes de referencia) para cada corte intercodificado (P).
- Muchos estándares de codificación, incluido H.264/AVC y HEVC, pueden tener un proceso de decodificación para derivar un índice de imagen de referencia a una lista de imágenes de referencia, que puede usarse para indicar cuál

de las múltiples imágenes de referencia se usa para la interpredicción para un bloque particular. Un índice de imagen de referencia puede codificarse por un codificador en el flujo de bits en algunos modos de intercodificación o puede derivarse (por un codificador y un decodificador), por ejemplo, usando bloques vecinos en algunos otros modos de intercodificación.

5 Se pueden obtener varios vectores de movimiento candidatos para una sola unidad de predicción. Por ejemplo, la HEVC de predicción de vectores de movimiento incluye dos planes de predicción de vectores de movimiento, a saber, la predicción avanzada de vectores de movimiento (AMVP) y el modo de fusión. En la AMVP o en el modo de fusión se obtiene una lista de candidatos a vectores de movimiento para una PU. Hay dos tipos de candidatos: candidatos espaciales y candidatos temporales; los candidatos temporales también pueden denominarse candidatos TMVP.

10 La obtención de lista de candidatos puede realizarse, por ejemplo, de la siguiente manera, aunque debe entenderse que pueden existir otras posibilidades para la obtención de lista de candidatos. Si la ocupación de la lista de candidatos no es máxima, los candidatos espaciales se incluyen primero en la lista de candidatos si están disponibles y no existen ya en la lista de candidatos. Después de eso, si la ocupación de la lista de candidatos aún no es máxima, se incluye un candidato temporal en la misma. Si el número de candidatos no alcanza aún el número máximo permitido, se agregan los candidatos bipredictivos combinados (para cortes B) y un vector de movimiento nulo. Una vez que se ha construido la lista de candidatos, el codificador decide la información de movimiento final a partir de unos candidatos basándose en, por ejemplo, una decisión de optimización de la razón tasa-distorsión (RDO) y codifica el índice del candidato seleccionado en el flujo de bits. Asimismo, el decodificador decodifica del flujo de bits el índice del candidato seleccionado, construye la lista de candidatos y usa el índice decodificado para seleccionar de la lista de candidatos un predictor de vector de movimiento.

15 En la HEVC, la AMVP y el modo de fusión se pueden caracterizar de la siguiente manera. En la AMVP, el codificador indica si se usará una unipredicción o una bipredicción y qué imágenes de referencia se usarán, así como codifica una diferencia entre vectores de movimiento. En el modo de fusión, solo se codifica en el flujo de bits el candidato elegido de la lista de candidatos, lo que indica que la unidad de predicción actual tiene la misma información de movimiento que el predictor indicado. Así, en el modo de fusión se crean unas regiones compuestas por bloques de predicción vecinos que comparten idéntica información de movimiento, lo cual solo se señala una vez para cada región.

20 En lo que sigue se proporciona un ejemplo de la operación de predicción avanzada de vectores de movimiento, aunque también son posibles otras realizaciones similares de la predicción avanzada de vectores de movimiento con, por ejemplo, distintos conjuntos de posiciones candidatas y ubicaciones candidatas con conjuntos de posiciones candidatas. Hay que entender también que otro modo de predicción, tal como el modo de fusión, puede funcionar de manera parecida. Se pueden derivar dos predictores de vector de movimiento espaciales (MVP) y se puede derivar un predictor de vector de movimiento temporal (TMVP). De entre las posiciones pueden seleccionarse: tres posiciones candidatas a predictor de vector de movimiento espacial que están ubicadas encima del bloque de predicción actual (B_0 , B_1 , B_2) y dos a la izquierda (A_0 , A_1). El primer predictor de vector de movimiento que esté disponible (p. ej., resida en el mismo corte, esté intercodificado, etc.) en un orden predefinido de cada conjunto de posiciones candidatas, (B_0 , B_1 , B_2) o (A_0 , A_1), puede seleccionarse para representar ese sentido de predicción (arriba o izquierda) en la competencia entre vectores de movimiento. Un índice de referencia para el predictor de vector de movimiento temporal puede indicarse por el codificador en la cabecera de corte (p. ej., como un elemento de sintaxis `collocated_ref_idx`). El primer predictor de vector de movimiento que esté disponible (p. ej., esté intercodificado) en un orden predefinido de posibles ubicaciones candidatas temporales, por ejemplo, en el orden (C_0 , C_1), puede seleccionarse como fuente para un predictor de vector de movimiento temporal. El vector de movimiento que se ha obtenido de la primera ubicación candidata disponible en la imagen coubicada puede ajustarse en escala según las proporciones de las diferencias entre recuentos de orden de imágenes de la imagen de referencia del predictor de vector de movimiento temporal, la imagen coubicada y la imagen actual. Además, se puede realizar una comprobación de redundancia entre los candidatos para eliminar candidatos idénticos, que pueden conducir a la inclusión de un vector de movimiento cero en la lista de candidatos. El predictor de vector de movimiento puede indicarse en el flujo de bits, por ejemplo, indicando la dirección del predictor de vector de movimiento espacial (hacia arriba o hacia la izquierda) o la selección del candidato del predictor de vector de movimiento temporal. La imagen coubicada también puede denominarse imagen coubicada, fuente para la predicción de vectores de movimiento o imagen fuente para la predicción de vectores de movimiento.

25 Los tipos de parámetro de movimiento o la información de movimiento puede(n) incluir, sin limitación, uno o más de los siguientes tipos:

– una indicación de un tipo de predicción (p. ej., intrapredicción, unipredicción, bipredicción) y/o una serie de imágenes de referencia;

– una indicación de un sentido de predicción, tal como la intrapredicción (también conocida como predicción temporal), la predicción intercapa, la predicción entrevista, la predicción de síntesis de vistas (VSP) y la predicción intercomponente (que puede indicarse por imagen de referencia y/o por tipo de predicción y donde, en algunas realizaciones, la predicción entrevista y la predicción de síntesis de vistas pueden considerarse conjuntamente como un sentido de predicción) y/o

– una indicación de un tipo de imagen de referencia, tal como una imagen de referencia a corto plazo y/o una imagen de referencia a largo plazo y/o una imagen de referencia intercapa (que puede indicarse por, p. ej., imagen de referencia)

5 – un índice de referencia a una lista de imágenes de referencia y/o cualquier otro identificador de una imagen de referencia (que puede indicarse por, p. ej., imagen de referencia y cuyo tipo puede depender del sentido de predicción y/o del tipo de imagen de referencia y que puede ir acompañado de otros datos relevantes, tales como la lista de imágenes de referencia o similares a los que aplique el índice de referencia);

10 – un componente de vector de movimiento horizontal (que puede indicarse por, p. ej., bloque de predicción o índice de referencia o similar);

15 – un componente de vector de movimiento vertical (que puede indicarse por, p. ej., bloque de predicción o índice de referencia o similar);

20 – uno o más parámetros, tales como la diferencia entre recuentos de orden de imágenes y/o una separación de cámara relativa entre la imagen que contiene o está asociada a los parámetros de movimiento y su imagen de referencia, que pueden usarse para ajustar a escala el componente de vector de movimiento horizontal y/o el componente de vector de movimiento vertical en uno o más procesos de predicción de vectores de movimiento (donde dichos uno o más parámetros pueden indicarse por, p. ej., cada imagen de referencia o cada índice de referencia o similar);

25 – unas coordenadas de un bloque al que aplica(n) los parámetros de movimiento y/o la información de movimiento, por ejemplo, las coordenadas de la muestra superior izquierda del bloque en unidades de muestra de luma;

30 – unas medidas (p. ej., una anchura y una altura) de un bloque al que aplica(n) los parámetros de movimiento y/o la información de movimiento.

En general, los mecanismos de predicción de vectores de movimiento, tales como aquellos mecanismos de predicción de vectores de movimiento que se han presentado anteriormente como ejemplos, pueden incluir una predicción o una herencia de ciertos parámetros de movimiento predefinidos o indicados.

35 Se puede considerar que un campo de movimiento asociado con una imagen comprende un conjunto de informaciones de movimiento producidas para cada bloque codificado de la imagen. Por ejemplo, un campo de movimiento puede ser accesible por coordenadas de un bloque. Un campo de movimiento puede usarse, por ejemplo, en la TMVP o en cualquier otro mecanismo de predicción de movimiento donde se use una fuente o una referencia para la predicción distinta de la imagen (de)codificada actual.

40 Se pueden aplicar distintas granularidades o unidades espaciales para representar y/o almacenar un campo de movimiento. Por ejemplo, se puede usar una cuadrícula regular de unidades espaciales. Por ejemplo, una imagen se puede dividir en unos bloques rectangulares de cierto tamaño (con la posible excepción de los bloques en los bordes de la imagen, tales como en el borde derecho y en el borde inferior). Por ejemplo, el tamaño de la unidad espacial puede ser igual al tamaño más pequeño para el que el codificador puede indicar un movimiento diferenciado en el flujo de bits, tal como un bloque de 4x4 en unidades de muestra de luma. Por ejemplo, se puede usar un denominado campo de movimiento comprimido, donde la unidad espacial puede ser igual a un tamaño predefinido o indicado, tal como un bloque de 16x16 en unidades de muestra de luma, cuyo tamaño puede ser más grande que el tamaño más pequeño para indicar un movimiento diferenciado. Por ejemplo, puede(n) implementarse un codificador y/o un decodificador de HEVC de una manera tal que se realice una reducción de almacenamiento de datos de movimiento (MDSR) o una compresión de campo de movimiento para cada campo de movimiento decodificado (antes de usarse el campo de movimiento para realizar cualquier predicción entre imágenes). En una implementación de HEVC, la MDSR puede reducir la granularidad de los datos de movimiento a bloques de 16x16 en unidades de muestra de luma manteniendo el movimiento aplicable a la muestra superior izquierda del bloque de 16x16 en el campo de movimiento comprimido. El codificador puede codificar un(as) indicación(ones) relacionada(s) con la unidad espacial del campo de movimiento comprimido como uno o más elementos sintácticos y/o valores de elemento sintáctico en, por ejemplo, una estructura sintáctica a nivel de secuencia, tal como un conjunto de parámetros de vídeo o un conjunto de parámetros de secuencia. En algunos métodos y/o dispositivos de (de)codificación, un campo de movimiento puede representarse y/o almacenarse según la partición en bloques de la predicción de movimiento (p. ej., según unidades de predicción del estándar de HEVC). En algunos métodos y/o dispositivos de (de)codificación se puede aplicar una combinación de una cuadrícula regular y una partición de bloques para que el movimiento que esté asociado a particiones que son mayores que un tamaño de unidad espacial predefinido o indicado se represente y/o almacene asociado a esas particiones, mientras que el movimiento que esté asociado a particiones que son más pequeñas que, o no están alineadas con, una cuadrícula o un tamaño de unidad espacial predefinido o indicado se represente y/o almacene para las unidades predefinidas o indicadas.

La codificación de vídeo escalable puede referirse a la estructura de codificación donde un flujo de bits puede contener múltiples representaciones del contenido, por ejemplo, a diferentes tasas de bits, resoluciones o tasas de fotogramas. En estos casos, el receptor puede extraer la representación deseada dependiendo de sus características (p. ej., resolución que coincida mejor con el dispositivo de visualización). Alternativamente, un servidor o un elemento de red puede extraer las porciones del flujo de bits para transmitirse al receptor dependiendo de, p. ej., las características de red o capacidades de procesamiento del receptor. Una representación decodificada significativa puede producirse al decodificar solo ciertas partes de un flujo de bits escalable. Un flujo de bits escalable consiste típicamente en una “capa base” que proporciona el vídeo de calidad más baja disponible y una o más capas de mejora que mejoran la calidad de vídeo cuando se reciben y decodifican junto con las capas inferiores. Para mejorar la eficiencia de codificación para las capas de mejora, la representación codificada de esa capa depende típicamente de las capas inferiores. Por ejemplo, el movimiento y la información de modo de la capa de mejora pueden predecirse a partir de capas inferiores. Similarmente, los datos de píxeles de las capas inferiores pueden usarse para crear predicción para la capa de mejora.

En algunos esquemas de codificación de vídeo escalable, una señal de vídeo puede codificarse en una capa base y una o más capas de mejora. Una capa de mejora puede mejorar, por ejemplo, la resolución temporal (es decir, la tasa de fotogramas), la resolución espacial o simplemente la calidad del contenido de vídeo representado por otra capa o parte de la misma. Cada capa junto con todas sus capas dependientes es una representación de la señal de vídeo, por ejemplo, a una determinada resolución espacial, resolución temporal y nivel de calidad. En este documento, se hace referencia a una capa escalable junto con todas sus capas dependientes como una “representación de capa escalable”. La porción de un flujo de bits escalable correspondiente a una representación de capa escalable puede extraerse y decodificarse para producir una representación de la señal original a cierta fidelidad.

Los modos de escalabilidad o las dimensiones de escalabilidad pueden incluir, aunque no de forma limitativa, los siguientes:

- Escalabilidad de calidad: Las imágenes de capa base se codifican a una calidad más baja que las imágenes de la capa de mejora, lo que puede lograrse, por ejemplo, utilizando un valor de parámetro de cuantificación mayor (es decir, un tamaño de etapa de cuantificación mayor para la cuantificación de coeficientes de transformada) en la capa base que en la capa de mejora. La escalabilidad de calidad se puede categorizar de forma adicional en una escalabilidad de grano fino o de granularidad fina (FGS), escalabilidad de grano medio o de granularidad media (MGS), y/o escalabilidad de grano grueso o de granularidad gruesa (CGS), como se describe a continuación.

- Escalabilidad espacial: Las imágenes de capa base se codifican a una resolución más baja (es decir, tienen menos muestras) que las imágenes de la capa de mejora. La escalabilidad espacial y la escalabilidad de calidad, particularmente su tipo de escalabilidad de grano grueso, a veces se pueden considerar el mismo tipo de escalabilidad.

- Escalabilidad de profundidad de bits: las imágenes de capa base se codifican a menor profundidad de bits (p. ej., 8 bits) que las imágenes de la capa de mejora (p. ej., 10 o 12 bits).

- Escalabilidad de intervalo dinámico: las capas escalables representan un intervalo dinámico diferente y/o unas imágenes obtenidas usando una función de mapeo de tonos diferente y/o una función de transferencia óptica diferente.

- Escalabilidad de formato de croma: las imágenes de capa base proporcionan una resolución espacial más baja en matrices de muestras de croma (p. ej., codificadas en formato de croma 4:2:0) que las imágenes de la capa de mejora (p.ej., formato 4:4:4).

- Escalabilidad de gama de colores: las imágenes de capa de mejora tienen un rango de representación de color más rico/más amplio que el de las imágenes de capa base - por ejemplo, la capa de mejora puede tener una gama de colores de UHD TV (ITU-R BT.2020) y la capa base pueden tener la gama de colores ITU-R BT.709.

- Escalabilidad de vista, que también puede denominarse codificación de múltiples vistas. La capa base representa una primera vista, mientras que una capa de mejora representa una segunda vista. Una vista puede definirse como una secuencia de imágenes que representa una cámara o un punto de vista. Puede considerarse que en vídeo estereoscópico o de dos vistas, se presenta una secuencia o vista de vídeo para el ojo izquierdo mientras se presenta una vista en paralelo para el ojo derecho.

- Escalabilidad de profundidad, que también puede denominarse codificación de profundidad mejorada. Una capa o algunas capas de un flujo de bits pueden representar la vista o vistas de textura, mientras que otra capa o capas pueden representar la vista o vistas de profundidad.

- Escalabilidad de la región de interés (como se describe a continuación).

– Escalabilidad de entrelazado a progresivo (también conocida como escalabilidad de campo a fotograma): El material de contenido de origen entrelazado codificado de la capa base se mejora con una capa de mejora para representar contenido de origen progresivo. El contenido de origen entrelazado codificado en la capa base puede comprender campos codificados, fotogramas codificados que representan pares de campos o una mezcla de ellos. En la escalabilidad de entrelazado a progresivo, la imagen de la capa base puede remuestrearse para que se convierta en una imagen de referencia adecuada para una o más imágenes de la capa de mejora.

– Escalabilidad de códec híbrido (también conocida como escalabilidad estándar de codificación): En la escalabilidad de códec híbrido, la sintaxis de flujo de bits, la semántica y el proceso de decodificación de la capa base y la capa de mejora se especifican en diferentes estándares de codificación de vídeo. Por lo tanto, las imágenes de la capa base se codifican según un estándar o formato de codificación diferente que las imágenes de la capa de mejora. Por ejemplo, la capa base puede codificarse según H.264/AVC y una capa de mejora puede codificarse según una ampliación multicapa de HEVC.

Debe entenderse que muchos de los tipos de escalabilidad pueden combinarse y aplicarse juntos. Por ejemplo, la escalabilidad de gama de colores y la escalabilidad de profundidad de bits se pueden combinar.

El término capa puede usarse en contexto de cualquier tipo de escalabilidad, incluidas escalabilidad de vista y mejoras de profundidad. Una capa de mejora puede referirse a cualquier tipo de mejora, tal como SNR, espacial, de múltiples vistas, profundidad, profundidad de bits, formato de croma y/o mejora de gama de colores. Una capa base puede referirse a cualquier tipo de una secuencia de vídeo de base, tal como una vista de base, una capa base para la escalabilidad de SNR/espacial, o una vista de base de textura para la codificación de vídeo de profundidad mejorada.

Algunos planes de codificación de vídeo escalables pueden requerir que las imágenes IRAP estén alineadas a través de las capas de una manera tal que todas las imágenes en una unidad de acceso sean imágenes IRAP o que ninguna imagen de una unidad de acceso sea una imagen IRAP. Otros planes de codificación de vídeo escalables, tales como las ampliaciones multicapa de la HEVC, pueden permitir imágenes IRAP que no estén alineadas, es decir, que una o más imágenes en una unidad de acceso sean imágenes IRAP, mientras que una o más otras imágenes en una unidad de acceso no sean imágenes IRAP. Pueden usarse flujos de bits escalables con imágenes IRAP o similares que no estén alineadas a través de las capas para, por ejemplo, proporcionar unas imágenes IRAP más frecuentes en la capa base, donde pueden tener un tamaño codificado más pequeño debido a, por ejemplo, una menor resolución espacial. Puede incluirse un proceso o mecanismo para el arranque por capas de la decodificación en un esquema de decodificación de vídeo. Por lo tanto, los decodificadores pueden iniciar la decodificación de un flujo de bits cuando una capa base contiene una imagen IRAP y empieza a decodificar por etapas otras capas cuando contienen imágenes IRAP. En otras palabras, en un comienzo por capas del mecanismo o proceso de decodificación, los decodificadores aumentan progresivamente el número de capas decodificadas (donde las capas pueden representar una mejora en la resolución espacial, el nivel de calidad, unas vistas, unos componentes adicionales, tales como la profundidad, o una combinación) a medida que en el proceso de decodificación se decodifican unas imágenes posteriores procedentes de unas capas de mejora adicionales. El aumento progresivo del número de capas decodificadas puede percibirse, por ejemplo, como una mejora progresiva de la calidad de la imagen (en caso de escalabilidad de calidad y espacial).

Un remitente, una pasarela, un cliente u otra entidad puede seleccionar las capas y/o subcapas transmitidas de un flujo de bits de vídeo escalable. Las expresiones extracción de capa, extracción de capas o reducción de capas pueden referirse a la transmisión de menos capas de las que hay disponibles en el flujo de bits recibido por el remitente, la pasarela, el cliente u otra entidad. Aumento de capas puede referirse a la transmisión de una(s) capa(s) adicional(es) en comparación con las transmitidas antes del aumento de capas realizado por el remitente, la pasarela, el cliente u otra entidad, es decir, el reinicio de la transmisión de una o más capas cuya transmisión se interrumpió anteriormente en la reducción de capas. De manera parecida a la reducción y/o al aumento de capas, el remitente, la pasarela, el cliente u otra entidad puede realizar una reducción y/o un aumento de subcapas temporales. El remitente, la pasarela, el cliente u otra entidad también puede realizar una reducción y/o un aumento tanto de capas como de subcapas. La reducción y/o el aumento de capas y subcapas se puede(n) llevar a cabo en la misma unidad de acceso o similar (i.e., de manera prácticamente simultánea) o se puede(n) llevar a cabo en distintas unidades de acceso o similares (i.e., prácticamente en momentos diferenciados).

La escalabilidad se puede posibilitar de dos maneras básicas. O bien introduciendo nuevos modos de codificación para realizar la predicción de valores de píxeles o sintaxis a partir de capas inferiores de la representación escalable o colocando las imágenes de capa inferior a una memoria intermedia de imagen de referencia (p. ej., una memoria intermedia de imágenes decodificadas, DPB) de la capa superior. El primer enfoque puede ser más flexible y, por lo tanto, puede proporcionar una mejor eficiencia de codificación en la mayoría de los casos. Sin embargo, la segunda escalabilidad basada en fotograma de referencia, el enfoque puede implementarse eficientemente con cambios mínimos en los códecs de una sola capa mientras aún se consigue la mayoría de las ganancias de eficiencia de codificación disponibles. Esencialmente, un códec de escalabilidad basado en fotograma de referencia puede implementarse utilizando la misma implementación de hardware o software para todas las capas, teniendo cuidado de la gestión de la DPB por medios externos.

Un codificador de vídeo escalable para la escalabilidad de calidad (también conocida como la Signal-to-Noise ([señal a ruido o SNR]) y/o escalabilidad espacial puede implementarse como sigue. Para una capa base, puede usarse un codificador y decodificador de vídeo no escalable convencional. Las imágenes reconstitutivas/decodificadas de la capa base se incluyen en la memoria intermedia de imágenes de referencia y/o en las listas de imágenes de referencia para una capa de mejora. En caso de escalabilidad espacial, la imagen de capa base reconstruida/decodificada puede sobremuestrearse antes de su inserción en las listas de imágenes de referencia para una imagen de capa de mejora. Las imágenes decodificadas de la capa base pueden insertarse en una lista o listas de imágenes de referencia para la codificación/decodificación de una imagen de capa de mejora similarmente a las imágenes de referencia decodificadas de la capa de mejora. Por consiguiente, el codificador puede elegir una imagen de referencia de capa base como una referencia de interpredicción e indicar su uso con un índice de imagen de referencia en el flujo de bits codificado. El decodificador decodifica del flujo de bits, por ejemplo, a partir de un índice de imagen de referencia, que se usa una imagen de capa base como referencia interpredicción para la capa de mejora. Cuando se usa una imagen de capa base decodificada como referencia de predicción para una capa de mejora, se denomina imagen de referencia intercapa.

Aunque el párrafo anterior describe un códec de vídeo escalable con dos capas de escalabilidad con una capa de mejora y una capa base, debe entenderse que la descripción puede generalizarse a cualesquiera dos capas en una jerarquía de escalabilidad con más de dos capas. En este caso, una segunda capa de mejora puede depender de una primera capa de mejora en los procesos de codificación y/o decodificación y, por lo tanto, la primera capa de mejora puede considerarse como la capa base para la codificación y/o decodificación de la segunda capa de mejora. Además, debe entenderse que puede haber imágenes de referencia intercapa de más de una capa en una memoria intermedia de imágenes de referencia o listas de imágenes de referencia de una capa de mejora, y cada una de estas imágenes de referencia intercapa puede considerarse que reside en una capa base o una capa de referencia para la capa de mejora que se codifica y/o decodifica. Además, hay que entender que pueden tener lugar, en vez o adicionalmente, otros tipos de procesamiento intercapa distintos del sobremuestreo de imágenes de capa de referencia. Por ejemplo, la profundidad de bits de las muestras de la imagen de capa de referencia puede convertirse en la profundidad de bits de la capa de mejora y/o los valores de muestra pueden someterse a una asignación del espacio de color de la capa de referencia al espacio de color de la capa de mejora.

Un esquema de codificación y/o decodificación de vídeo escalable puede usar codificación y/o decodificación de múltiples bucles, que puede caracterizarse como sigue. En la codificación/decodificación, una imagen de capa base puede reconstruirse/decodificarse para usarse como una imagen de referencia de compensación de movimiento para las imágenes posteriores, en orden de codificación/decodificación, dentro de la misma capa o como referencia para la predicción intercapa (o entrevista o intercomponente). La imagen de la capa base reconstruida/decodificada puede almacenarse en la DPB. Una imagen de capa de mejora puede reconstruirse/decodificarse de la misma manera para usarse como una imagen de referencia de compensación de movimiento para las imágenes posteriores, en orden de codificación/decodificación, dentro de la misma capa o como referencia para la predicción intercapa (o entrevista o intercomponente) para las capas de mejora superiores, si la hay. Además de los valores de muestra de reconstrucción/decodificados, los valores de elementos de sintaxis de la capa base/de referencia o las variables derivadas de los valores de elementos de sintaxis de la capa base/de referencia pueden usarse en la predicción intercapa/intercomponente/intervista.

La predicción intercapa puede definirse como una predicción de una manera que es dependiente de unos elementos de dato (p. ej., unos valores de muestra o unos vectores de movimiento) de unas imágenes de referencia de una capa diferente de la capa de la imagen actual (que se está codificando o decodificando). Existen muchos tipos de predicción intercapa, y se pueden aplicar en un codificador/decodificador de vídeo escalable. Los tipos disponibles de predicción intercapa pueden depender, por ejemplo, del perfil de codificación según el cual se esté codificando el flujo de bits o una capa particular dentro del flujo de bits o, cuando se esté decodificando, del perfil de codificación al que se indica que se ajusta el flujo de bits o una capa particular dentro del flujo de bits. Alternativa o adicionalmente, los tipos disponibles de predicción intercapa pueden depender de los tipos de escalabilidad o del tipo de códec escalable o de modificación de estándar de codificación de vídeo o (p. ej., SHVC, MV-HEVC o 3D-HEVC) que se esté utilizando.

Una capa de referencia directa puede definirse como una capa que puede usarse para la predicción intercapa de otra capa para la que la capa es la capa de referencia directa. Una capa predicha directa puede definirse como una capa para la que otra capa es una capa de referencia directa. Una capa de referencia indirecta puede definirse como una capa que no es una capa de referencia directa de una segunda capa pero es una capa de referencia directa de una tercera capa que es una capa de referencia directa o una capa de referencia indirecta de una capa de referencia directa de la segunda capa para la cual la capa es la capa de referencia indirecta. Una capa predicha indirecta puede definirse como una capa para la que otra capa es una capa de referencia indirecta. Una capa independiente se puede definir como una capa que no tiene capas de referencia directa. En otras palabras, una capa independiente no se predice usando una predicción intercapa. Una capa no de base se puede definir como cualquier otra capa que no sea la capa base, y la capa base se puede definir como la capa más baja en el flujo de bits. Una capa no de base independiente puede definirse como una capa que es tanto una capa independiente como una capa no de base.

En algunos casos, los datos en una capa de mejora pueden truncarse después de una cierta ubicación, o incluso en posiciones arbitrarias, donde cada posición de truncamiento puede incluir datos adicionales que representan una calidad visual cada vez mayor. Dicha escalabilidad se denomina escalabilidad (de granularidad) de grano fino (FGS).

5 Similarmente a la MVC, en MV-HEVC, las imágenes de referencia intermedia pueden incluirse en la lista o listas de imágenes de referencia de la imagen actual que se está codificando o decodificando. La SHVC utiliza una operación de decodificación multilazo (a diferencia de la ampliación de SVC de H.264/AVC). Puede considerarse que la SHVC usa un enfoque basado en índices de referencia, es decir, una imagen de referencia intermedia puede estar incluida en una o más listas de imágenes de referencia de la imagen actual que se está codificando o decodificando (tal y como se ha descrito anteriormente).

10 Para la codificación de capas de mejora, pueden usarse los mismos conceptos y herramientas de codificación de la capa base de HEVC en la SHVC, la MV-HEVC y/o similares. Sin embargo, las herramientas de predicción intermedia adicionales, que emplean datos ya codificados (que incluyen muestras de imágenes reconstruidas y parámetros de movimiento, también conocidos como información de movimiento) en la capa de referencia para codificar eficientemente una capa de mejora, pueden integrarse en el códec SHVC, MV-HEVC y/o similares.

15 Como se ha descrito anteriormente, las muestras de vídeo e imagen se codifican típicamente usando representaciones de color tales como YUV o YCbCr que consisten en un canal de luminancia y dos de crominancia. En estos casos, el canal de luminancia, que representa principalmente la iluminación de la escena, se codifica típicamente con una resolución determinada, mientras que los canales de crominancia, que representan típicamente diferencias entre ciertos componentes de color, se codifican a menudo con una segunda resolución inferior a la de la señal de luminancia. La intención de este tipo de representación diferencial es descorrelacionar los componentes de color y poder comprimir los datos de manera más eficiente. Sin embargo, en muchos casos, aún queda algo de correlación entre los canales que podrían utilizarse para representar los datos de manera más eficiente.

A continuación, se presenta un método mejorado para codificación de canales de color.

20 En la Figura 5, se muestra un método según un aspecto, comprendiendo el método determinar (500) una señal residual para al menos una muestra; determinar (502) si dicha señal residual representa un residuo para muestras en más de un canal; y, si es afirmativo, aplicar (504) dicha señal residual para al menos una primera muestra en un primer canal para generar una primera muestra reconstruida; y aplicar (506) dicha señal residual para al menos una segunda muestra en un segundo canal para generar una segunda muestra reconstruida.

25 Según una realización, se aplica una señal residual combinada para los canales de crominancia de una imagen fija o una secuencia de vídeo.

30 Por lo tanto, se proporciona un método para codificar conjuntamente la señal residual para dos o más canales de color, especialmente los canales de crominancia. Se puede usar un identificador de flujo de bits para detectar los casos donde la misma señal residual se aplica a múltiples canales de crominancia. Al indicar y decodificar una indicación de que la misma señal residual se puede aplicar a múltiples canales de crominancia, se puede mejorar aún más la eficiencia de compresión de la codificación/decodificación de vídeo.

35 El método y las realizaciones relacionadas se aplican igualmente a las operaciones realizadas por un codificador o un decodificador, salvo que se indique lo contrario en la presente memoria. El método y las realizaciones relacionadas pueden implementarse de diferentes maneras. Por ejemplo, el orden de las operaciones descritas anteriormente se puede cambiar o las operaciones se pueden intercalar de diferentes maneras. Además, se pueden aplicar diferentes operaciones adicionales en diferentes etapas del procesamiento. Por ejemplo, se puede aplicar un filtrado, escalado, mapeo u otro procesamiento adicional al resultado final o intermedio de las operaciones descritas. El resultado final o intermedio de las operaciones descritas anteriormente también puede combinarse además con los resultados de otras operaciones.

40 La determinación de la señal residual se puede realizar de diversas maneras, en donde la implementación real de la determinación de la señal residual típicamente difiere entre un decodificador y un codificador. Por ejemplo, un decodificador de vídeo o imagen puede decodificar elementos sintácticos de un flujo de bits que describe los coeficientes de transformada que pueden transformarse adicionalmente usando, por ejemplo, una transformada discreta de coseno (DCT) inversa en un bloque de valores de muestra residuales. Naturalmente, también se pueden usar otras transformadas, como la transformada sinusoidal discreta (DST) o una transformada de ondícula, o se puede omitir la etapa de transformada, y los valores de muestra residuales se pueden construir directamente en función de los elementos sintácticos que describen las diferencias de muestra para muestras residuales individuales o un bloque de muestras residuales.

45 La señal residual puede determinarse de diversas maneras también en un codificador de vídeo o imagen. En el caso de que el residuo se comparta entre dos o más canales, se puede generar un residuo conjunto, por ejemplo, promediando o usando un promedio ponderado de las señales residuales y, opcionalmente, transformarse, cuantificarse con la precisión deseada, cuantificarse inversamente y transformarse inversamente en el dominio de la

muestra. Un codificador de vídeo o imagen puede configurarse, por ejemplo, para añadir la mitad del residuo desde un canal y deducir la mitad del residuo a partir de otro canal al generar una señal residual conjunta o un bloque de muestras residuales conjuntas. Como parte del proceso de cuantificación, se pueden usar procesos de cuantificación optimizados para la distorsión de la velocidad. En tales casos, se puede considerar la naturaleza multicanal del residuo conjunto y el proceso de cuantificación se puede adaptar para, por ejemplo, ponderar el error de cuantificación estimado de manera diferente al caso de la codificación residual monocanal.

La determinación de si el residuo representa un residuo para las muestras en más de un canal se puede realizar de diversas maneras. En un codificador de vídeo o imagen, se pueden probar diferentes modos de funcionamiento y se puede seleccionar uno con, por ejemplo, el coste de distorsión de velocidad más bajo para una unidad de codificación, unidad de predicción, unidad de transformada u otro bloque de muestras. La información relevante se puede señalar en un flujo de bits, desde donde un decodificador de vídeo/imagen puede obtener la información y decodificarla. La señalización de flujo de bits y la decodificación de la información también se pueden realizar de diferentes maneras.

Según una realización, el método comprende además incluir un indicador en el flujo de bits para indicar que se cumple al menos una condición predefinida para la decodificación residual. Tal indicador puede denominarse indicador residual combinado. Por ejemplo, se puede usar un indicador identificador de 1 bit o un indicador identificador codificado por contexto de 1 binario como indicación en un flujo de bits, en donde dicho indicador residual combinado se puede incluir ventajosamente en el flujo de bits en una o más condiciones predefinidas.

Según una realización, se decodifica un indicador residual combinado y, en respuesta a que el indicador residual combinado sea 1 o verdadero, se decodifica un único bloque residual y se aplica el residuo tanto a un primer canal como a un segundo canal. Por lo tanto, un ejemplo de tales condiciones incluye indicar o decodificar el indicador solo si primero se indica que ambos canales relacionados con la crominancia tienen un residuo asociado a ellos.

Según una realización, se decodifica un indicador de bloque codificado para un primer canal, se decodifica un indicador de bloque codificado para un segundo canal y, en respuesta a que ambos sean 1 o verdaderos, se decodifica un indicador residual combinado y, en respuesta a que el indicador residual combinado sea 1 o verdadero, se decodifica un único bloque residual y se aplica tanto al primer como al segundo canal. Un ejemplo de sintaxis de flujo de bits para este tipo de indicación se proporciona a continuación como transform_unit_example_1, donde primero se indica con un indicador de bloques codificados (cbf) que ambos canales relacionados con la crominancia cb y cr tienen un residuo asociado a ellos, y después se indica que se puede aplicar un residuo conjunto a ambos canales.

```

transform_unit_example_1(x0, y0) {
...
tu_cbf_cb[ x0 ][ y0 ]
tu_cbf_cr[ x0 ][ y0 ]
si(tu_cbf_cb[ x0 ][ y0 ] && tu_cbf_cr[ x0 ][ y0 ]) {
tu_cb_cr_joint_residual[ x0 ][ y0 ]
}
...
}
    
```

Según una realización, se decodifica un indicador residual combinado y, en respuesta a que el indicador residual combinado sea 1 o verdadero, se decodifica un único bloque residual y se añade al bloque de predicción en un primer canal y se deduce a partir del bloque de predicción en un segundo canal. En este caso, el residuo del segundo canal se puede omitir en el flujo de bits y el residuo del segundo canal se puede calcular en su lugar a partir del residuo conjunto recibido para el primer canal, por ejemplo, a partir de la sintaxis anterior.

Según una realización, se decodifica un identificador asociado a un primer canal de color y, en respuesta a ese identificador, la señal residual de un segundo canal se deduce a partir de la señal de predicción en el primer canal. Por lo tanto, otro ejemplo de indicación condicional para el indicador es indicar o decodificar primero un identificador que indique si hay residuos para el primer canal. Si dicha indicación implica que hay un residuo para el primer canal, entonces se usa una bandera para indicar si dicho residuo es un residuo conjunto para dos o más canales o solo se aplica para el primer canal. En este ejemplo, no hay necesidad de decodificar un elemento sintáctico que indique un residuo codificado para el segundo canal en el caso donde se indique un residuo conjunto. Un ejemplo de sintaxis de flujo de bits para este tipo de indicación se proporciona a continuación como transform_unit_example_2, donde se usa un indicador de bloque codificado (cbf) como identificador para indicar que hay un residuo asociado con el canal cb relacionado con la crominancia. El elemento sintáctico tu_cb_cr_joint_residual se usa para indicar que hay un residuo de unión para ambos canales cb y cr relacionados con la crominancia.

5	transform_unit_example_2(x0, y0) {
	...
	tu_cbf_cb [x0][y0]
	si(tu_cbf_cb[x0][y0]) {
	tu_cb_cr_joint_residual [x0][y0]
	}
10	si(! tu_cb_cr_joint_residual[x0][y0]) {
	tu_cbf_cr [x0][y0]
	}
	...
15	}

Otra posibilidad más para la señalización condicional es indicar primero que uno de los canales tiene un residuo. Este tipo de ejemplo se muestra a continuación como transform_unit_example_3. En este caso, indicar el residuo solo para el segundo canal usando el indicador tu_cbf_cr está activando la decodificación del indicador tu_cb_cr_joint_residual. Naturalmente, también se podría usar la bandera para el primer canal (tu_cbf_cb) en su lugar.

25	transform_unit_example_3(x0, y0) {
	...
	tu_cbf_cb [x0][y0]
	tu_cbf_cr [x0][y0]
30	si(tu_cbf_cb[x0][y0] == 0 && tu_cbf_cr[x0][y0] == 1) {
	tu_cb_cr_joint_residual [x0][y0]
	}
	...
35	}

En otro ejemplo, la indicación de un modo residual conjunto se puede realizar antes de los indicadores de bloques codificados. En este caso, a ambos indicadores de bloques codificados tu_cbf_cb y tu_cbf_cr se les puede asignar un valor verdadero o 1 en el caso de que se señalice el modo residual conjunto y no sea necesario codificar o decodificar explícitamente los indicadores cbf. Por lo tanto, la realización mencionada anteriormente, donde se decodifica un indicador residual combinado y, en respuesta a que el indicador residual combinado sea 1 o verdadero, también se puede aplicar en la presente memoria la decodificación de un único bloque residual y la aplicación del residuo tanto a un primer canal como a un segundo canal.

En otro ejemplo, la indicación de un modo residual conjunto se puede realizar dentro de la sintaxis de codificación residual de uno de los bloques residuales. Esto se ilustra en transform_unit_example_4, que contiene identificadores para indicadores de bloques codificados para los bloques Cb y Cr; y residual_coding_example_4, que contiene una comprobación de que el índice del canal actual coincide con el de un primer canal (en este ejemplo, el canal Cr en comparación con cidX == 2) y un indicador de bloque codificado para que el segundo canal (en este ejemplo, el canal Cb con tu_cbf_cb) sea 1 o verdadero. En el caso de que ambas condiciones sean verdaderas, se decodifica una indicación del modo residual conjunto y, en el caso de que se evalúe como verdadero, se pueden omitir otras operaciones de análisis para el residuo de Cr, ya que el residuo de Cr se puede derivar del residuo de Cb. Alternativamente, puede existir alguna señalización adicional para indicar cómo se relacionan los residuos entre sí.

55	transform_unit_example_4(x0, y0) {
	...
	tu_cbf_cb [x0][y0]
60	tu_cbf_cr [x0][y0]
	...
	}
65	residual_coding_example_4(x0, y0, log2TbWidth, log2TbHeight, cidX) {
	...

```

5      si(cldx == 2 && tu_cbf_cb[ x0 ][ y0 ]) {
      tu_cb_cr_joint_residual[ x0 ][ y0 ]
      si(tu_cb_crjoint_residual [ x0 ][ y0 ]) {
      return
      }
10     ...
      }

```

Según una realización, el modo residual combinado se aplica a un subconjunto de bloques determinado por la señalización de flujo de bits.

15 Según una realización, el modo residual combinado se aplica solo a bloques con un modo de predicción que pertenece a un conjunto predeterminado de modos de predicción. El conjunto predeterminado de modos de predicción puede incluir, por ejemplo, todos o un subconjunto de todos los modos de intrapredicción.

20 Según una realización, el modo residual combinado se aplica solo a bloques con un modo de codificación residual que pertenece a un conjunto predeterminado de modos de codificación residual. El conjunto predeterminado de modos de codificación residual puede incluir, por ejemplo, todos o un subconjunto de todos los modos de codificación de transformada. La indicación del modo residual combinado puede codificarse o decodificarse para una sola unidad de transformada o cubrir múltiples unidades de transformada. Por ejemplo, la indicación se puede dar en un nivel de unidad de árbol de codificación, unidad de codificación, unidad de predicción o árbol de transformada raíz que cubre potencialmente una multitud de unidades de transformada.

25 Si se determina que una señal residual representa un residuo para las muestras en más de un canal, se puede decir que la unidad de codificación, el bloque de codificación, la unidad de predicción, el bloque de predicción, la unidad de transformada o el bloque de transformada para los que se realiza la determinación se encuentra en una unidad o bloque de "modo residual combinado" o de "modo residual compartido" o en una unidad o bloque de "modo residual conjunto". Similarmente, se puede decir que el "modo residual combinado/compartido/conjunto" está "activado" para esa unidad o bloque.

30 Según una realización, el método comprende además aplicar medios de posprocesamiento a dicha al menos primera muestra reconstruida para generar al menos una primera muestra de salida en dicho primer canal; y aplicar medios de posprocesamiento a dicha al menos segunda muestra reconstruida para generar al menos una segunda muestra de salida en dicho segundo canal.

35 Según una realización, el método comprende además establecer el parámetro de cuantificación o el parámetro de cuantificación inversa usado en la codificación residual en función de si se aplica un modo de codificación residual combinado para un bloque de muestras o no. El parámetro de cuantificación inversa o el tamaño de la etapa de cuantificación inversa se pueden establecer ventajosamente más pequeños para los bloques para los que se aplica el modo de codificación residual combinado, en comparación con aquellos bloques para los que el modo está desactivado. Similarmente, el residuo combinado se puede escalar hacia arriba antes de la codificación y/o hacia abajo después de la decodificación para lograr efectos similares.

40 Según una realización, la información adicional se decodifica a partir de un flujo de bits y se usa junto con el residuo conjunto para determinar los bloques de muestras reconstruidos. Tal información adicional podría incluir, por ejemplo, valores escalares o matriciales de escalado, parámetros para mapear valores de muestras de forma lineal o no lineal; o valores de muestra o desplazamientos de muestra adicionales codificados por transformada o no codificados por transformada.

45 Si se determina que la señal residual representa un residuo para muestras en más de un canal, el residuo puede aplicarse a dichos canales de diferentes maneras. Según una realización, se decodifica un indicador de bloque codificado para un primer canal, se decodifica un indicador de bloque codificado para un segundo canal y, en respuesta a que el indicador de bloque codificado del primer canal sea 0 o falso y el indicador de bloque codificado para el segundo canal sea 1 o verdadero, se añade una versión a escala del residuo del segundo canal a la predicción del primer canal. Por ejemplo, para un primer canal se puede usar una escala escalar representada por el valor s_u , mientras que se puede usar un valor escalar s_v diferente para un segundo canal. En forma de matriz, esto se puede describir para un bloque de muestras como:

$$O_u = P_u + s_u * R_{uv}$$

$$O_v = P_v + s_v * R_{uv}$$

65

donde O_u y O_v representan los bloques de muestras reconstruidos (o muestras individuales si se usa un tamaño de bloque de 1x1 muestras) en los canales U y V, respectivamente. Similarmente, P_u y P_v representan los bloques de predicción de muestras correspondientes en estos canales, mientras que R_{uv} representa el residuo conjunto.

5 Según una realización, la señal residual asociada a un primer canal de color se decodifica, un identificador asociado a un segundo canal de color se decodifica y, en respuesta a ese identificador, la señal residual del primer canal se deduce a partir de la señal de predicción del segundo canal. Por lo tanto, los valores escalares s_u y s_v se pueden seleccionar ventajosamente para vídeo YUV o UCBcr como 1 y -1, lo que permite que las ecuaciones se representen de una forma más compacta:

10

$$O_u = P_u + R_{uv}$$

$$O_v = P_v + R_{uv}$$

15 Según una realización, la diferencia entre la señal residual del primer canal y la señal residual del segundo canal se codifica en un flujo de bits.

Según una realización, una señal residual decodificada se añade a un bloque de muestras en un primer canal y se deduce a partir de un bloque de muestras en un segundo canal.

20

Las Figuras 6a y 6b muestran un ejemplo que ilustra los beneficios del modo residual combinado/compartido según las realizaciones. La Figura 6a ilustra la codificación convencional de señales residuales. En este ejemplo, hay dos canales de crominancia, ambos con señal residual individual. Los bloques de salida O_u y O_v se generan añadiendo los bloques residuales a los bloques predichos en cada canal. La Figura 6b, a su vez, ilustra la operación en un modo residual compartido. En este ejemplo, las ponderaciones +1 y -1 están asociadas al primer y el segundo canal, respectivamente. Es decir, el residuo señalado se añade al bloque de predicción en el primer canal y se deduce a partir del bloque de predicción en el segundo canal para formar dos bloques de salida O_u y O_v .

25

La solución se puede generalizar naturalmente para usar matrices de ponderaciones en lugar de ponderaciones escalares s_u y s_v . Alternativamente, o además, las ponderaciones pueden depender de la señal y derivarse para cada muestra por separado usando, por ejemplo, mapeos lineales, lineales por partes o no lineales. Un ejemplo de este tipo de mapeo consiste en calcular un modelo lineal utilizando un conjunto de muestras reconstruidas circundantes de bloques reconstruidos O_u y O_v y aplicar ese modelo para calcular los valores de $s_u(x, y)$ y $s_v(x, y)$ en las coordenadas x, y dentro del bloque de muestras, dando:

35

$$O_u = P_u + s_u(x, y) * R_{uv}$$

$$O_v = P_v + s_v(x, y) * R_{uv}$$

40 o:

$$O_u(x, y) = P_u(x, y) + s_u(x, y) * R_{uv}(x, y)$$

$$O_v(x, y) = P_v(x, y) + s_v(x, y) * R_{uv}(x, y)$$

45

o expresar las ponderaciones como una función de la señal de predicción:

$$O_u(x, y) = P_u(x, y) + s_u(P_u(x, y)) * R_{uv}(x, y)$$

50

$$O_v(x, y) = P_v(x, y) + s_v(P_v(x, y)) * R_{uv}(x, y)$$

Cuando se aplica en el codificador de vídeo o imagen, la determinación de la señal residual combinada o conjunta puede incluir calcular una diferencia o diferencias entre las muestras de entrada originales o procesadas y los valores de muestra pronosticados. También puede incluir operaciones adicionales, como el filtrado o las operaciones no lineales basadas en muestras aplicadas a las instancias intermedias del residuo combinado. El residuo de la unión puede transformarse en un dominio diferente (por ejemplo, usando DCT) y cuantificarse en el caso de la codificación por transformada o simplemente cuantificarse si se aplica el modo de codificación similar a la omisión de transformada. Si se aplica una cuantificación optimizada para la distorsión de la velocidad al residuo combinado, un codificador puede ajustar ventajosamente el parámetro lambda utilizado para relacionar el error de reconstrucción y los bits necesarios en la codificación para que sean diferentes de los utilizados para los modos de codificación residual tradicionales, ya que en el caso de la codificación residual combinada, el residuo se aplica a múltiples bloques de codificación. Alternativamente, un codificador puede seleccionar ajustar la estimación del error de cuantificación al realizar selecciones de modo.

65

La Figura 7 muestra un diagrama de bloques de un decodificador de vídeo adecuado para emplear realizaciones de la invención. La Figura 7 representa una estructura de un decodificador de dos capas, pero se apreciará que las operaciones de decodificación pueden emplearse similarmente en un decodificador monocapa.

5 El decodificador 550 de vídeo comprende una primera sección 552 de decodificador para una capa base y una segunda sección 554 de decodificador para una capa predicha. Un bloque 556 ilustra un demultiplexor para distribuir información sobre unas imágenes de capa base a la primera sección 552 de decodificador y para distribuir información sobre unas imágenes de capa predicha a la segunda sección 554 de decodificador. La referencia P'n representa una representación predicha de un bloque de imagen. La referencia D'n representa una señal de error de predicción reconstruida. Los bloques 704, 804 ilustran imágenes reconstruidas preliminares (I'n). La referencia R'n representa una imagen reconstruida final. Los bloques 703, 803 ilustran la transformada inversa (T^{-1}). Los bloques 702, 802 ilustran la cuantificación inversa (Q^{-1}). Los bloques 701, 801 ilustran la decodificación por entropía (E^{-1}). Los bloques 705, 805 ilustran una memoria de fotograma de referencia (RFM). Los bloques 706, 806 ilustran la predicción (P) (ya sea interpredicción o intrapredicción). Los bloques 707, 807 ilustran el filtrado (F). Los bloques 708, 808 pueden usarse para combinar una información de errores de predicción decodificada con unas imágenes de capa predicha/capa base predichas para obtener las imágenes reconstruidas preliminares (I'n). Se puede dar salida 709 a unas imágenes de capa base reconstruidas y filtradas preliminares de la primera sección 552 de decodificador y se puede dar salida a 809 a unas imágenes de capa base reconstruidas y filtradas preliminares de la primera sección 554 de decodificador.

20 En la presente memoria, debe interpretarse que el decodificador abarca cualquier unidad operativa que sea capaz de llevar a cabo las operaciones de decodificación, tal como un reproductor, un receptor, una pasarela, un demultiplexor y/o un decodificador.

25 Como un aspecto adicional, se proporciona un aparato que comprende: al menos un procesador y al menos una memoria, dicha al menos una memoria almacenada con un código en la misma, que cuando es ejecutado por dicho al menos un procesador, hace que el aparato realice al menos: determinar una señal residual para al menos una muestra; determinar si dicha señal residual representa un residuo para muestras en más de un canal; aplicar, en respuesta a que dicha señal residual represente un residuo para muestras en más de un canal, dicha señal residual para al menos una primera muestra en un primer canal para generar una primera muestra reconstruida; y aplicar dicha señal residual para al menos una segunda muestra en un segundo canal para generar una segunda muestra reconstruida.

35 Un aparato de este tipo comprende además un código, almacenado en dicha al menos una memoria, que cuando es ejecutado por dicho al menos un procesador, hace que el aparato realice una o más de las realizaciones descritas en la presente memoria.

40 La Figura 8 es una representación gráfica de un sistema de comunicación multimedia de ejemplo dentro del cual pueden implementarse diversas realizaciones. Una fuente 1510 de datos proporciona una señal de origen en un formato digital analógico, no comprimido digital o comprimido digital, o cualquier combinación de estos formatos. Un codificador 1520 puede incluir o estar conectado a un preprocesamiento, tal como una conversión y/o un filtrado de formato de datos de la señal de origen. El codificador 1520 codifica la señal de origen en un flujo de bits de medios codificados. Debe observarse que un flujo de bits a decodificar puede recibirse directa o indirectamente desde un dispositivo remoto ubicado dentro de prácticamente cualquier tipo de red. De forma adicional, el flujo de bits puede recibirse desde hardware o software local. El codificador 1520 puede ser capaz de codificar más de un tipo de medio, tal como audio y vídeo, o puede requerirse más de un codificador 1520 para codificar diferentes tipos de medios de la señal de origen. El codificador 1520 también puede obtener una entrada producida sintéticamente, tal como gráficos y texto, o puede ser capaz de producir flujos de bits codificados de medios sintéticos. A continuación, solo se considera el procesamiento de un flujo de bits de medios codificados de un tipo de medios para simplificar la descripción. Debe observarse, sin embargo, que típicamente los servicios de difusión en tiempo real comprenden varios flujos (típicamente al menos un flujo de audio, de vídeo y de subtítulos de texto). También debe tenerse en cuenta que el sistema puede incluir muchos codificadores, pero en la figura solo se representa un codificador 1520 para simplificar la descripción sin una falta de generalidad. Debe entenderse además que, aunque el texto y los ejemplos contenidos en la presente memoria pueden describir específicamente un proceso de codificación, un experto en la técnica entendería que los mismos conceptos y principios también se aplican al proceso de decodificación correspondiente y viceversa.

55 El flujo de bits de medios codificados puede transferirse a un almacenamiento 1530. El almacenamiento 1530 puede comprender cualquier tipo de memoria masiva para almacenar el flujo de bits de medios codificados. El formato del flujo de bits de medios codificado en el almacenamiento 1530 puede ser un formato de flujo de bits autónomo elemental, o uno o más flujos de bits de medios codificados pueden encapsularse en un archivo contenedor, o el flujo de bits de medios codificado puede encapsularse en un formato de segmento adecuado para la DASH (o un sistema de transmisión en continuo similar) y almacenarse como una secuencia de segmentos. Si uno o más flujos de bits de medios están encapsulados en un archivo contenedor, puede usarse un generador de archivos (no mostrado en la figura) para almacenar el uno más flujos de bits de medios en el archivo y crear metadatos de formato de archivo, que también pueden almacenarse en el archivo. El codificador 1520 o el almacenamiento 1530 pueden comprender el generador de archivos, o el generador de archivos está conectado operativamente al codificador 1520 o al

almacenamiento 1530. Algunos sistemas operan en “vivo”, es decir, omiten el almacenamiento y la transferencia del flujo de bits de medios codificados desde el codificador 1520 directamente al emisor 1540. El flujo de bits de medios codificados puede entonces transferirse al emisor 1540, también denominado servidor, basándose en la necesidad. El formato usado en la transmisión puede ser un formato de flujo de bits autónomo elemental, un formato de flujo de paquetes o un formato de segmento adecuado para la DASH (o un sistema de transmisión en continuo parecido), o uno o más flujos de bits de medios codificados pueden estar encapsulados en un archivo contenedor. El codificador 1520, el almacenamiento 1530 y el servidor 1540 pueden residir en el mismo dispositivo físico o pueden incluirse en dispositivos separados. El codificador 1520 y el servidor 1540 pueden funcionar con contenido en tiempo real en vivo, en cuyo caso el flujo de bits de medios codificados típicamente no se almacena permanentemente, sino más bien se almacena en memoria intermedia durante pequeños períodos de tiempo en el codificador 1520 de contenido y/o en el servidor 1540 para suavizar variaciones en el retardo de procesamiento, retardo de transferencia y tasa de bits codificada.

El servidor 1540 envía el flujo de bits de medios codificados usando una pila de protocolos de comunicación. La pila puede incluir, aunque no de forma limitativa, uno o más del Real-Time Transport Protocol (protocolo de transporte en tiempo real - RTP), el User Datagram Protocol (protocolo de datagramas de usuario - UDP), el Hypertext Transfer Protocol (protocolo de transferencia de hipertexto - HTTP), el Transmission Control Protocol (protocolo de control de transmisión - TCP) y el Internet Protocol (protocolo de internet - IP). Cuando la pila de protocolos de comunicación está orientada a paquetes, el servidor 1540 encapsula el flujo de bits de medios codificados en paquetes. Por ejemplo, cuando se usa RTP, el servidor 1540 encapsula el flujo de bits de medios codificados en paquetes RTP según un formato de carga útil del RTP. Típicamente, cada tipo de medio tiene un formato de carga útil del RTP especializado. Debe observarse nuevamente que un sistema puede contener más de un servidor 1540, pero por motivos de simplicidad, la siguiente descripción solo considera un servidor 1540.

Si el contenido de medios está encapsulado en un archivo contenedor para el almacenamiento 1530 o para introducir los datos en el emisor 1540, el emisor 1540 puede comprender o estar unido operativamente a un “analyzer de archivos de envío” (no mostrado en la figura). En particular, si el archivo de contenedor no se transmite como tal pero al menos uno del flujo de bits de medios codificados contenido está encapsulado para el transporte a través de un protocolo de comunicación, un analyzer de archivos de envío ubica las partes apropiadas del flujo de bits de medios codificados a transportar a través del protocolo de comunicación. El analyzer de archivos de envío también puede ayudar a crear el formato correcto para el protocolo de comunicación, tal como las cabeceras de paquetes y cargas útiles. El archivo contenedor multimedia puede contener unas instrucciones de encapsulación, tales como unas pistas de indicación en el ISOBMFF, para una encapsulación del al menos uno de los flujos de bits de medios contenidos en el protocolo de comunicación.

El servidor 1540 puede o no estar conectado a una pasarela 1550 a través de una red de comunicación, que puede ser, por ejemplo, una combinación de una CDN, internet y/o una o más redes de acceso. La puerta de enlace también, o alternativamente, se puede denominar caja intermedia. Para la DASH, la pasarela puede ser un servidor de borde (de una CDN) o un proxy web. Se observa que el sistema puede comprender generalmente cualquier número de puertas de enlace o similar, pero por motivos de simplicidad, la siguiente descripción solo considera una puerta 1550 de enlace. La puerta 1550 de enlace puede realizar diferentes tipos de funciones, tales como la traducción de un flujo de paquetes según una pila de protocolos de comunicación a otra pila de protocolos de comunicación, la fusión y bifurcación de flujos de datos, y la manipulación de flujo de datos según las capacidades de enlace descendente y/o receptor, tal como controlar la tasa de bits del flujo reenviado según las condiciones de red de enlace descendente predominantes. En diversas realizaciones, la pasarela 1550 puede ser una entidad de servidor.

El sistema incluye uno o más receptores 1560, normalmente capaces de recibir, demodular y desencapsular la señal transmitida en un flujo de bits de medios codificados. El flujo de bits de medios codificados puede transferirse a un almacenamiento 1570 de grabación. El almacenamiento 1570 de grabación puede comprender cualquier tipo de memoria masiva para almacenar el flujo de bits de medios codificados. El almacenamiento 1570 de grabación puede comprender alternativa o aditivamente memoria de cálculo, tal como memoria de acceso aleatorio. El formato del flujo de bits de medios codificados en el almacenamiento 1570 de grabación puede ser un formato de flujo de bits autónomo elemental, o bien uno o más flujos de bits de medios codificados pueden encapsularse en un archivo contenedor. Si hay múltiples flujos de bits de medios codificados, tales como un flujo de audio y un flujo de vídeo, asociados entre sí, típicamente se usa un archivo de contenedor y el receptor 1560 comprende o está unido a un generador de archivo de contenedor que produce un archivo de contenedor a partir de flujos de entrada. Algunos sistemas operan en “vivo”, es decir, omiten el almacenamiento 1570 de grabación y la transferencia del flujo de bits de medios codificados desde el receptor 1560 directamente al decodificador 1580. En algunos sistemas, solo se mantiene la parte más reciente del flujo registrado, p. ej., el extracto de los 10 minutos más recientes del flujo registrado, en el almacenamiento 1570 de grabación, mientras que cualquier dato registrado anteriormente se descarta del almacenamiento 1570 de grabación.

El flujo de bits de medios codificados puede transferirse desde el almacenamiento 1570 de grabación al decodificador 1580. Si hay muchos flujos de bits de medios codificados, tales como un flujo de audio y un flujo de vídeo, asociados entre sí y encapsulados en un archivo contenedor o se encapsula un único flujo de bits de medios en un archivo contenedor, p. ej., para un acceso más fácil, se usa un analyzer de archivos (no mostrado en la figura) para desencapsular cada flujo de bits de medios codificados desde el archivo contenedor. El almacenamiento 1570 de

grabación o un decodificador 1580 puede comprender el analizador de archivos, o el analizador de archivos está unido o bien al almacenamiento 1570 de grabación o bien al decodificador 1580. También debe tenerse en cuenta que el sistema puede incluir muchos decodificadores, pero en este punto solo se describe un decodificador 1570 para simplificar la descripción sin una falta de generalidad

El flujo de bits de medios codificados puede procesarse además mediante un decodificador 1570, cuya salida es uno o más flujos de medios no comprimidos. Finalmente, un representador 1590 puede reproducir los flujos de medios no comprimidos con un altavoz o una pantalla, por ejemplo. El receptor 1560, el almacenamiento 1570 de grabación, el decodificador 1570 y el representador 1590 pueden residir en el mismo dispositivo físico o pueden incluirse en dispositivos separados.

Un remitente 1540 y/o una pasarela 1550 puede(n) configurarse para realizar un cambio entre distintas representaciones, por ejemplo, para cambiar entre distintas ventanas gráficas de un contenido de vídeo de 360 grados, cambiar de vista o realizar una adaptación de tasa de bits y/o un inicio rápido, y/o un remitente 1540 y/o una pasarela 1550 puede(n) configurarse para seleccionar la(s) representación(ones) transmitida(s). El cambio entre distintas representaciones puede tener lugar por múltiples razones, tales como para responder a solicitudes del receptor 1560 o a condiciones predominantes, tales como el caudal, de la red por la cual se transmite el flujo de bits. En otras palabras, el receptor 1560 puede iniciar el cambio entre representaciones. Una solicitud del receptor puede ser, por ejemplo, una solicitud de un segmento o un subsegmento de una representación diferente que la anterior o una solicitud para un cambio de capas y/o subcapas de escalabilidad transmitidas o un cambio de un dispositivo de renderizado que tiene capacidades diferentes en comparación con el anterior. Una solicitud de un segmento puede ser una solicitud GET de HTTP. Una solicitud de un subsegmento puede ser una solicitud GET de HTTP con un intervalo de bytes. Adicional o alternativamente, puede usarse un ajuste de tasa de bits o una adaptación de tasa de bits para, por ejemplo, proporcionar un denominado inicio rápido en servicios de transmisión en continuo, donde la tasa de bits del flujo transmitido es menor que la tasa de bits de canal después de iniciar o acceder aleatoriamente a la transmisión en continuo para iniciar inmediatamente la reproducción y para lograr un nivel de ocupación de memoria intermedia que tolere retardos y/o retransmisiones de paquetes ocasionales. La adaptación de tasa de bits puede incluir múltiples operaciones de representación o de cambio ascendente de capa y representación o de cambio descendente de capa que tienen lugar en diversos órdenes.

Un decodificador 1580 puede configurarse para realizar un cambio entre distintas representaciones, por ejemplo, para cambiar entre distintas ventanas de visualización de un contenido de vídeo de 360 grados, cambiar de vistas o realizar una adaptación de tasa de bits y/o un inicio rápido, y/o un decodificador 1580 puede configurarse para seleccionar la(s) representación(ones) transmitida(s). El cambio entre distintas representaciones puede tener lugar por múltiples razones, tales como conseguir una operación de decodificación más rápida o adaptar el flujo de bits transmitido, por ejemplo, en términos de tasa de bits, a las condiciones predominantes, tales como el caudal, de la red por la cual se transmite el flujo de bits. Puede que sea necesaria una operación de decodificación más rápida si, por ejemplo, el dispositivo que incluye el decodificador 1580 está realizando múltiples tareas y utilizando recursos de cálculo para fines distintos que la decodificación del flujo de bits de vídeo. En otro ejemplo, podría ser necesaria una operación de decodificación más rápida cuando el contenido se reproduce a un ritmo más rápido que la velocidad de reproducción normal, p. ej., dos veces o tres veces más rápido que la tasa de reproducción en tiempo real convencional.

En lo anterior se han descrito algunas realizaciones haciendo referencia y/o usando la terminología de la HEVC. Hay que entender que pueden realizarse igualmente realizaciones con cualquier codificador de vídeo y/o decodificador de vídeo.

En lo anterior, cuando se han descrito las realizaciones de ejemplo con referencia a un codificador, debe entenderse que el flujo de bits resultante y el decodificador pueden tener elementos correspondientes en ellos. De la misma manera, cuando se han descrito las realizaciones de ejemplo con referencia a un decodificador, debe entenderse que el codificador puede tener una estructura y/o un programa informático para generar el flujo de bits para que se decodifique por el decodificador. Por ejemplo, se han descrito algunas realizaciones que están relacionadas con la generación de un bloque de predicción como parte de la codificación. Se pueden, igualmente, realizar unas realizaciones generando un bloque de predicción como parte de la decodificación, con la diferencia de que unos parámetros de codificación, tales como el desfase horizontal y el desfase vertical, se decodifican a partir del flujo de bits en vez de ser determinados por el codificador.

Las realizaciones de la invención descritas anteriormente describen el códec en términos de un aparato codificador y decodificador separado para ayudar al entendimiento de los procesos implicados. Sin embargo, se apreciará que el aparato, las estructuras y las operaciones pueden implementarse como un único aparato/estructura/operación de codificador-decodificador. Además, es posible que el codificador y el decodificador puedan compartir algunos o todos los elementos comunes.

Aunque los ejemplos anteriores describen realizaciones de la invención que funcionan dentro de un códec dentro de un dispositivo electrónico, se apreciará que la invención, como se define en las reivindicaciones, puede implementarse como parte de cualquier códec de vídeo. Por lo tanto, por ejemplo, las realizaciones de la invención pueden

implementarse en un códec de vídeo que puede implementar la codificación de vídeo a través de rutas de comunicación fijas o por cable.

5 Por lo tanto, el equipo de usuario puede comprender un códec de vídeo tal como los descritos en las realizaciones de la invención anteriores. Se apreciará que el término equipo de usuario está destinado a cubrir cualquier tipo adecuado de equipo de usuario inalámbrico, tal como teléfonos móviles, dispositivos portátiles de procesamiento de datos o navegadores web portátiles.

10 Además, los elementos de una public land mobile network (red móvil pública terrestre - PLMN) también pueden comprender códecs de vídeo como se ha descrito anteriormente.

15 En general, las diversas realizaciones de la invención pueden implementarse en hardware o circuitos de propósito especial, software, lógica o cualquier combinación de los mismos. Por ejemplo, algunos aspectos pueden implementarse en hardware, mientras que otros aspectos pueden implementarse en firmware o software que puede ejecutarse por un controlador, microprocesador u otro dispositivo informático, aunque la invención no se limita a los mismos. Aunque diversos aspectos de la invención pueden ilustrarse y describirse como diagramas de bloques, diagramas de flujo, o usando alguna otra representación gráfica, se entiende que estos bloques, aparatos, sistemas, técnicas o métodos descritos en la presente memoria pueden estar implementados, como ejemplos no limitativos, en hardware, software, firmware, circuitos o lógica de propósito especial, hardware de propósito general o controlador u otros dispositivos informáticos, o alguna combinación de los mismos.

20 Las realizaciones de esta invención pueden implementarse mediante software informático ejecutable por un procesador de datos del dispositivo móvil, tal como en la entidad de procesador, o mediante hardware, o mediante una combinación de software y hardware. Además, en este sentido, cabe señalar que cualesquiera bloques del flujo lógico como en las figuras puede representar etapas del programa, o circuitos lógicos interconectados, bloques y funciones, o una combinación de etapas de programa y circuitos lógicos, bloques y funciones. El software puede almacenarse en medios físicos tales como chips de memoria, o bloques de memoria implementados dentro del procesador, medios magnéticos tales como disco duro o disquetes, y medios ópticos tales como, por ejemplo, DVD y sus variantes de datos, CD.

30 La memoria puede ser de cualquier tipo adecuado para el entorno técnico local y puede implementarse usando cualquier tecnología de almacenamiento de datos adecuada, tal como dispositivos de memoria basados en semiconductores, dispositivos y sistemas de memoria magnéticos, dispositivos y sistemas de memoria ópticos, memoria fija y memoria extraíble. Los procesadores de datos pueden ser de cualquier tipo adecuado para el entorno técnico local, y pueden incluir uno o más de ordenadores de propósito general, ordenadores de propósito especial, microprocesadores, digital signal processors (procesadores de señales digitales - DSP) y procesadores basados en arquitectura de procesador de múltiples núcleos, como ejemplos no limitativos.

40 Las realizaciones de las invenciones pueden ponerse en práctica en diversos componentes tales como módulos de circuito integrado. El diseño de circuitos integrados es, en gran medida, un proceso altamente automatizado. Hay herramientas de software complejas y potentes disponibles para convertir un diseño de nivel lógico en un diseño de circuito de semiconductores listo para grabarse y formarse en un sustrato semiconductor.

45 Los programas, tales como los proporcionados por Synopsys, Inc. de Mountain View, California y Cadence Design, de San Jose, California, enrutan automáticamente conductores y localizan componentes en un chip de semiconductor usando reglas de diseño bien establecidas, así como bibliotecas de módulos de diseño previamente almacenados. Una vez que se ha completado el diseño para un circuito de semiconductor, el diseño resultante, en un formato electrónico normalizado (p. ej., Opus, GDSII o similar) puede transmitirse a una instalación de fabricación de semiconductores o "fab" para su fabricación.

50 La descripción anterior ha proporcionado, a modo de ejemplo y como ejemplos no limitativos, una descripción completa e informativa de la realización ilustrativa de esta invención. Sin embargo, diversas modificaciones y adaptaciones pueden resultar evidentes para los expertos en las técnicas relevantes a la vista de la descripción anterior, cuando se lee junto con los dibujos adjuntos y las reivindicaciones adjuntas. Sin embargo, todas esas modificaciones y similares de las enseñanzas de esta invención estarán todavía dentro del alcance de esta invención.

60 En general, las diversas realizaciones de la invención pueden implementarse en hardware o circuitos de propósito especial, software, lógica o cualquier combinación de los mismos. Por ejemplo, algunos aspectos pueden implementarse en hardware, mientras que otros aspectos pueden implementarse en firmware o software que puede ejecutarse por un controlador, microprocesador u otro dispositivo informático, aunque la invención no se limita a los mismos. Aunque diversos aspectos de la invención pueden ilustrarse y describirse como diagramas de bloques, diagramas de flujo, o usando alguna otra representación gráfica, se entiende que estos bloques, aparatos, sistemas, técnicas o métodos descritos en la presente memoria pueden estar implementados, como ejemplos no limitativos, en hardware, software, firmware, circuitos o lógica de propósito especial, hardware de propósito general o controlador u otros dispositivos informáticos, o alguna combinación de los mismos.

65

Las realizaciones de esta invención pueden implementarse mediante software informático ejecutable por un procesador de datos del dispositivo móvil, tal como en la entidad de procesador, o mediante hardware, o mediante una combinación de software y hardware. Además, en este sentido, cabe señalar que cualesquiera bloques del flujo lógico como en las figuras puede representar etapas del programa, o circuitos lógicos interconectados, bloques y funciones, o una combinación de etapas de programa y circuitos lógicos, bloques y funciones. El software puede almacenarse en medios físicos tales como chips de memoria, o bloques de memoria implementados dentro del procesador, medios magnéticos tales como disco duro o disquetes, y medios ópticos tales como, por ejemplo, DVD y sus variantes de datos, CD.

La memoria puede ser de cualquier tipo adecuado para el entorno técnico local y puede implementarse usando cualquier tecnología de almacenamiento de datos adecuada, tal como dispositivos de memoria basados en semiconductores, dispositivos y sistemas de memoria magnéticos, dispositivos y sistemas de memoria ópticos, memoria fija y memoria extraíble. Los procesadores de datos pueden ser de cualquier tipo adecuado para el entorno técnico local, y pueden incluir uno o más de ordenadores de propósito general, ordenadores de propósito especial, microprocesadores, digital signal processors (procesadores de señales digitales - DSP) y procesadores basados en arquitectura de procesador de múltiples núcleos, como ejemplos no limitativos.

Las realizaciones de las invenciones pueden ponerse en práctica en diversos componentes tales como módulos de circuito integrado. El diseño de circuitos integrados es, en gran medida, un proceso altamente automatizado. Hay herramientas de software complejas y potentes disponibles para convertir un diseño de nivel lógico en un diseño de circuito de semiconductores listo para grabarse y formarse en un sustrato semiconductor.

Los programas, tales como los proporcionados por Synopsys, Inc. de Mountain View, California y Cadence Design, de San Jose, California, enrutan automáticamente conductores y localizan componentes en un chip de semiconductor usando reglas de diseño bien establecidas, así como bibliotecas de módulos de diseño previamente almacenados. Una vez que se ha completado el diseño para un circuito de semiconductor, el diseño resultante, en un formato electrónico normalizado (p. ej., Opus, GDSII o similar) puede transmitirse a una instalación de fabricación de semiconductores o "fab" para su fabricación.

La descripción anterior ha proporcionado, a modo de ejemplo y como ejemplos no limitativos, una descripción completa e informativa de la realización ilustrativa de esta invención. Sin embargo, diversas modificaciones y adaptaciones pueden resultar evidentes para los expertos en las técnicas relevantes a la vista de la descripción anterior, cuando se lee junto con los dibujos adjuntos y las reivindicaciones adjuntas. Sin embargo, todas esas modificaciones y similares de las enseñanzas de esta invención estarán todavía dentro del alcance de esta invención.

REIVINDICACIONES

1. Un aparato que comprende:
 - 5 medios para decodificar un flujo de bits que comprende elementos sintácticos que describen coeficientes de transformada o diferencias de muestra para muestras residuales; medios para determinar (500) un bloque de muestras residuales al menos para un primer canal de crominancia basándose en los elementos sintácticos; **caracterizado porque** el aparato comprende medios para decodificar (502) un indicador residual combinado que indica si dicho bloque de muestras residuales representa el residuo para un bloque de muestras en más de un canal de crominancia; medios para aplicar (504), en respuesta a que dicho bloque de muestras residuales represente el residuo para un bloque de muestras en más de un canal de crominancia, dicho bloque de muestras residuales para al menos un primer bloque de muestras en el primer canal de crominancia para generar un primer bloque de muestras reconstruidas; y medios para aplicar (506) dicho bloque de muestras residuales para al menos un segundo bloque de muestras en un segundo canal de crominancia para generar un segundo bloque de muestras reconstruidas.
- 20 2. El aparato según la reivindicación 1, que comprende además: medios para aplicar dicho bloque de muestras residuales para al menos el primer bloque de muestras en el primer canal de crominancia y al menos el segundo bloque de muestras en el segundo canal de crominancia de una imagen fija o una secuencia de vídeo.
- 25 3. El aparato según la reivindicación 1 o 2, que comprende además:
 - medios para decodificar el indicador residual combinado; medios para decodificar, en respuesta a que el indicador residual combinado sea 1 o verdadero, un único bloque residual; y
 - 30 medios para aplicar el bloque de muestras residuales tanto al primer canal de crominancia como al segundo canal de crominancia.
4. El aparato según cualquier reivindicación anterior, que comprende además:
 - 35 medios para decodificar un indicador de bloque codificado para el primer canal de crominancia y un indicador de bloque codificado para el segundo canal de crominancia; medios para decodificar, en respuesta a que ambos indicadores de bloques codificados sean 1 o verdaderos, el indicador residual combinado; medios para decodificar, en respuesta a que el indicador residual combinado sea 1 o verdadero, un
 - 40 único bloque residual; y medios para aplicar dicho único bloque residual tanto para el primer canal de crominancia como para el segundo canal de crominancia.
- 45 5. El aparato según cualquier reivindicación anterior, que comprende además:
 - medios para decodificar el indicador residual combinado; medios para decodificar, en respuesta a que el indicador residual combinado sea 1 o verdadero, un
 - 50 único bloque residual; medios para añadir el único bloque residual a un bloque de predicción en el primer canal de crominancia; y medios para deducir el único bloque residual a partir de un bloque de predicción en el segundo canal de crominancia.
- 55 6. El aparato según cualquier reivindicación anterior, que comprende además: medios para aplicar el bloque de muestras residuales a un subconjunto de bloques en el primer y el segundo canal de crominancia para los que el indicador residual combinado se indica en un flujo de bits.
7. El aparato según la reivindicación 6, que comprende además:
 - 60 medios para aplicar el bloque combinado de muestras residuales a bloques con un modo de predicción que pertenece a un conjunto predeterminado de modos de predicción.
8. El aparato según la reivindicación 6, que comprende además:
 - medios para aplicar el bloque combinado de muestras residuales a bloques con un modo de codificación residual que pertenece a un conjunto predeterminado de modos de codificación residual.
- 65 9. El aparato según cualquier reivindicación anterior, que comprende además:

- 5 medios para aplicar el posprocesamiento a dicha al menos primera muestra reconstruida para generar al menos una primera muestra de salida en dicho primer canal de crominancia; y
medios para aplicar el posprocesamiento a dicha al menos segunda muestra reconstruida para generar al menos una segunda muestra de salida en dicho segundo canal de crominancia.
10. El aparato según cualquier reivindicación anterior, que comprende además:
medios para usar una versión a escala del bloque de muestras residuales de al menos uno del primer y el segundo canal de crominancia al generar el bloque combinado de muestras residuales.
- 10 11. Un método para predicción con compensación de movimiento, comprendiendo el método:

15 decodificar un flujo de bits que comprende elementos sintácticos que describen coeficientes de transformada o diferencias de muestra para muestras residuales;
determinar (500) un bloque de muestras residuales al menos para un primer canal de crominancia basándose en los elementos sintácticos; **caracterizado por**
decodificar (502) un indicador residual combinado que indica si dicho bloque de muestras residuales representa el residuo para un bloque de muestras en más de un canal de crominancia; y si es afirmativo
20 aplicar (504) dicho bloque de muestras residuales para al menos un primer bloque de muestras en el primer canal de crominancia para generar un primer bloque de muestras reconstruidas; y
aplicar (506) dicho bloque de muestras residuales para al menos un segundo bloque de muestras en un segundo canal de crominancia para generar un segundo bloque de muestras reconstruidas.
- 25 12. El método según la reivindicación 11, que comprende además:
aplicar dicho bloque de muestras residuales para al menos el primer bloque de muestras en el primer canal de crominancia y al menos el segundo bloque de muestras en el segundo canal de crominancia de una imagen fija o una secuencia de vídeo.
- 30 13. El método según la reivindicación 12, que comprende además:

decodificar el indicador residual combinado;
decodificar, en respuesta a que el indicador residual combinado sea 1 o verdadero, un único bloque residual; y
35 aplicar el bloque de muestras residuales tanto al primer canal de crominancia como al segundo canal de crominancia.
- 40 14. Un producto de programa informático incorporado en un medio legible por ordenador no transitorio, que comprende un código de programa informático configurado para, cuando se ejecuta en al menos un procesador, hacer que un aparato o un sistema:

45 decodifique un flujo de bits que comprende elementos sintácticos que describen coeficientes de transformada o diferencias de muestra para muestras residuales;
determine (500) un bloque de muestras residuales al menos para un primer canal de crominancia basándose en los elementos sintácticos; **caracterizado por** un código de programa informático configurado para
decodificar (502) un indicador residual combinado que indica si dicho bloque de muestras residuales representa el residuo para un bloque de muestras en más de un canal de crominancia;
50 aplicar (504), en respuesta a que dicho bloque de muestras residuales represente el residuo para un bloque de muestras en más de un canal de crominancia, dicho bloque de muestras residuales para al menos un primer bloque de muestras en el primer canal de crominancia para generar un primer bloque de muestras reconstruidas; y
aplicar (506) dicho bloque de muestras residuales para al menos un segundo bloque de muestras en un segundo canal de crominancia para generar un segundo bloque de muestras reconstruidas.
- 55
- 60
- 65

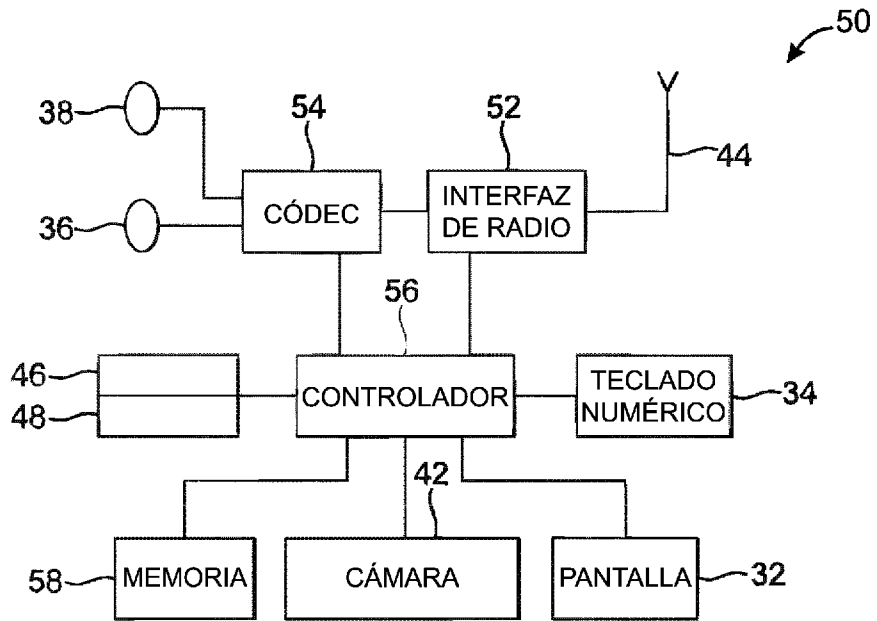


Figura 1

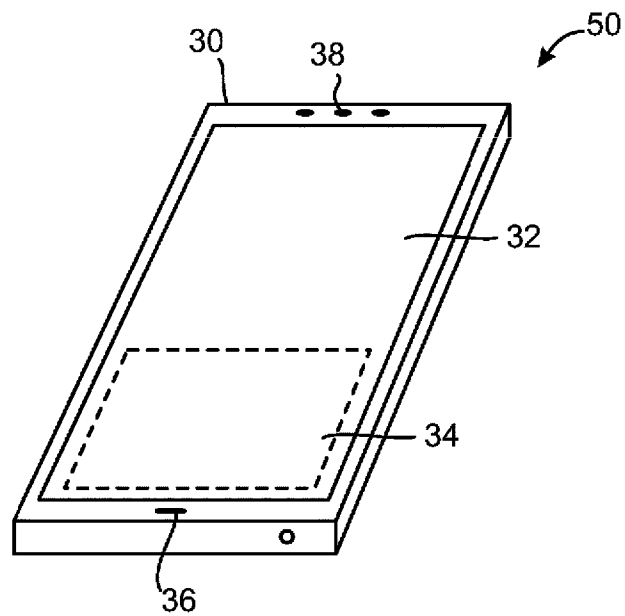


Figura 2

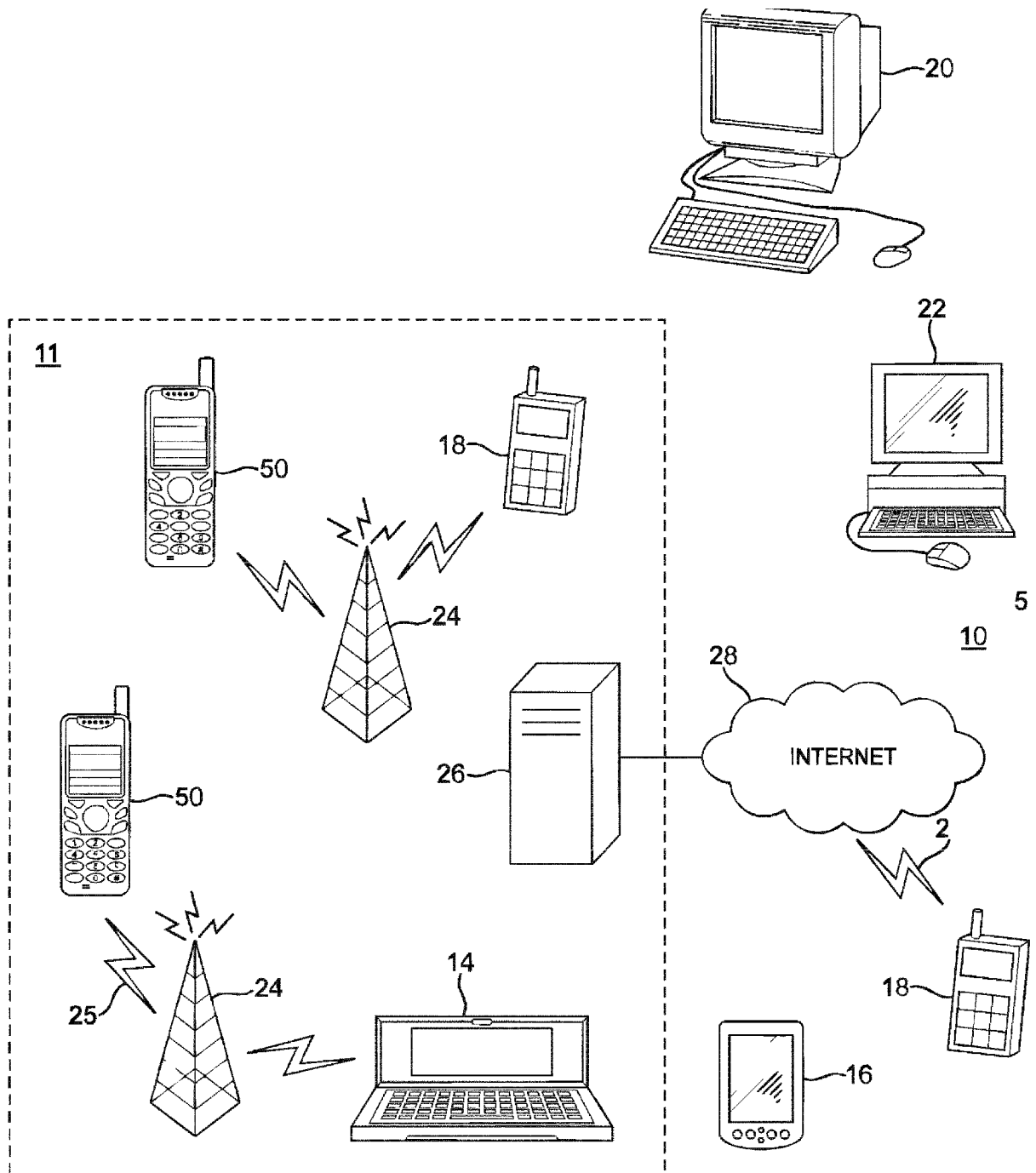


Figura 3

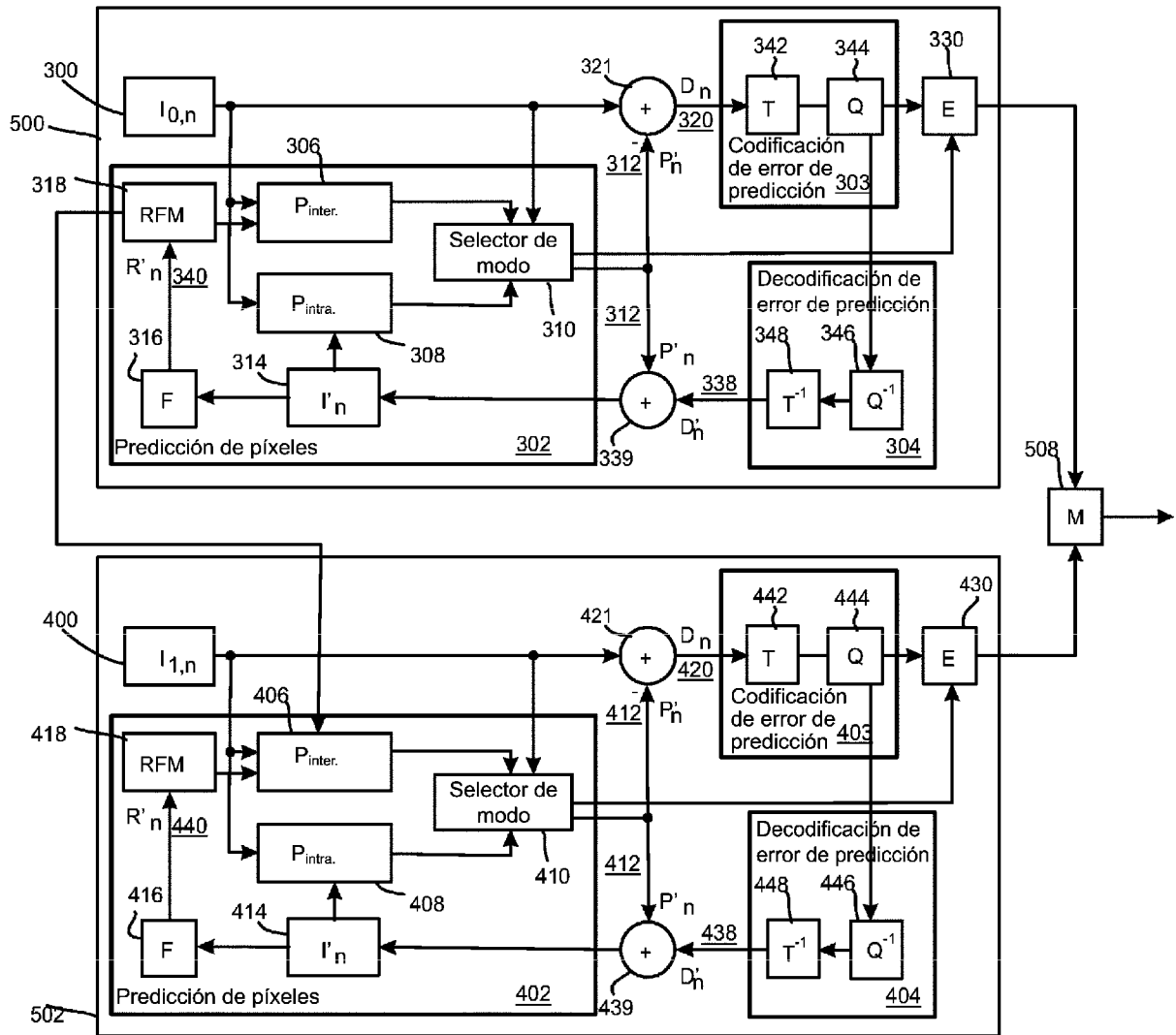


Figura 4

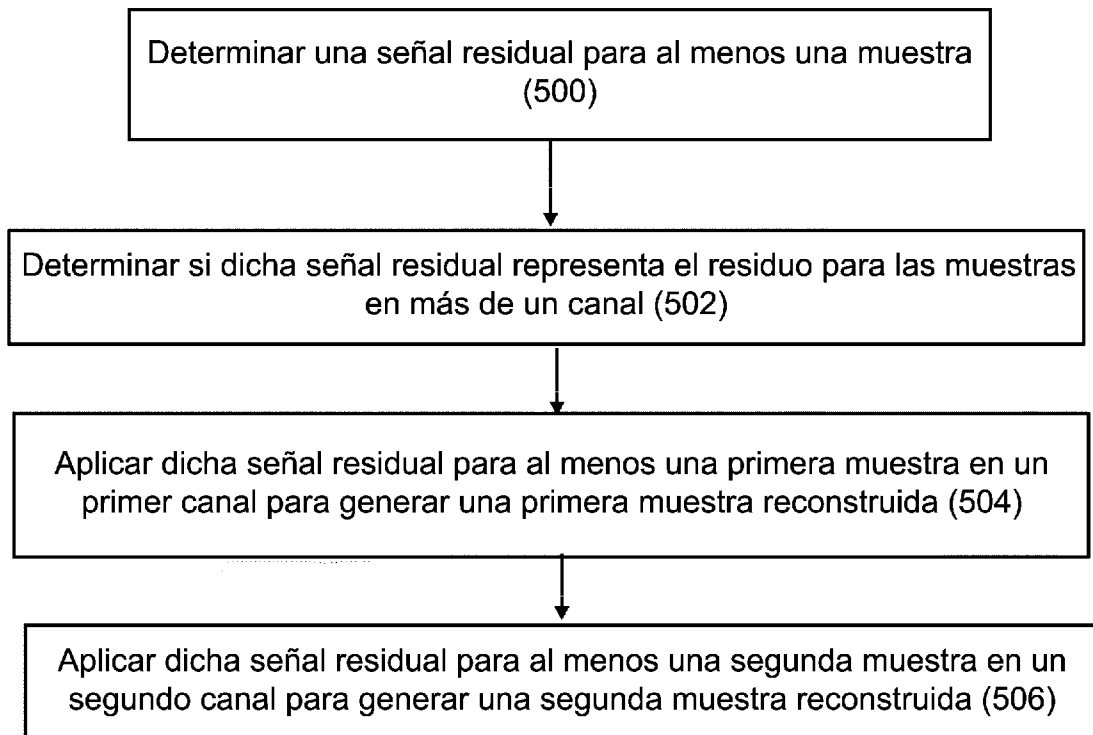


Figura 5

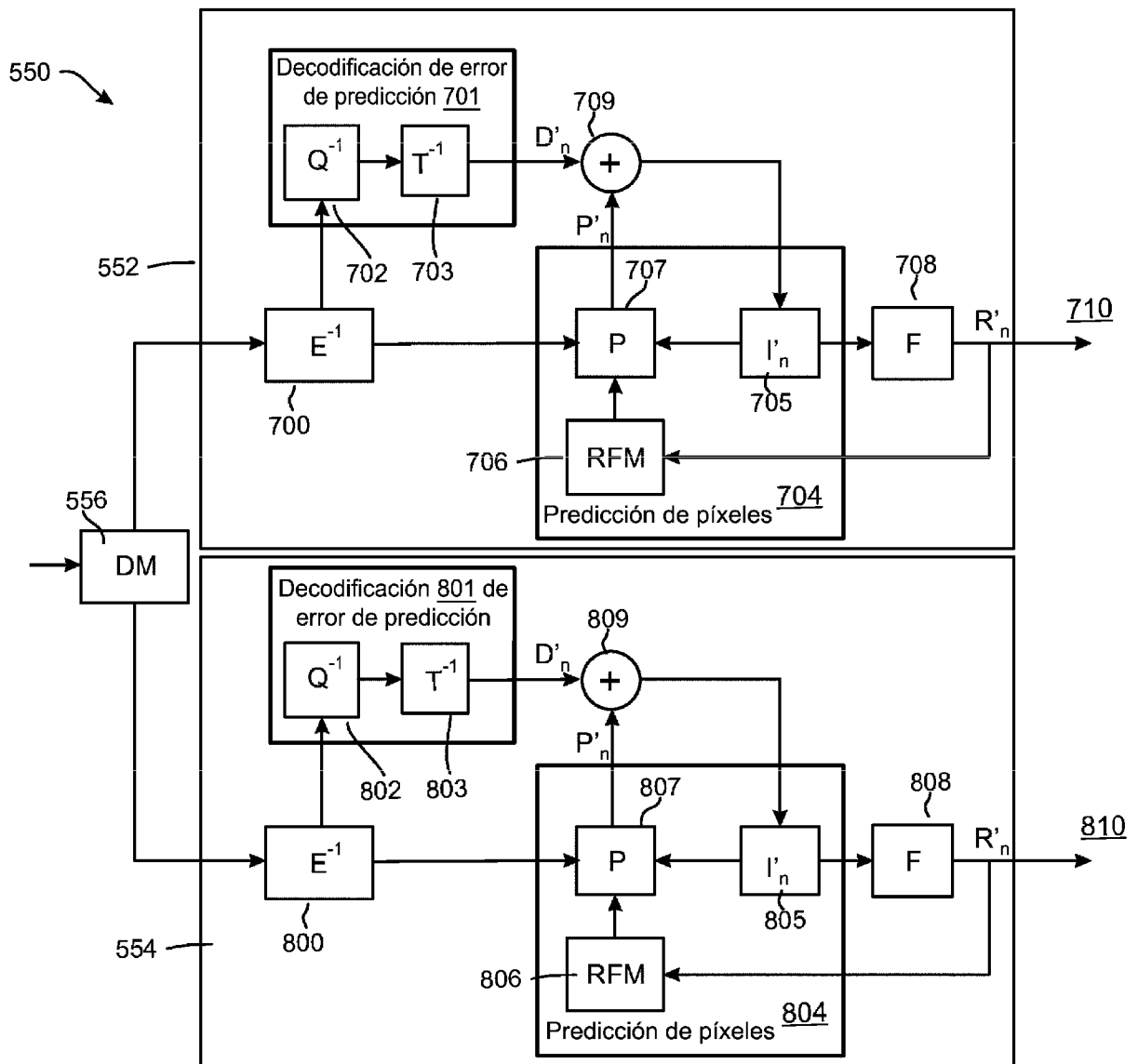


Figura 7

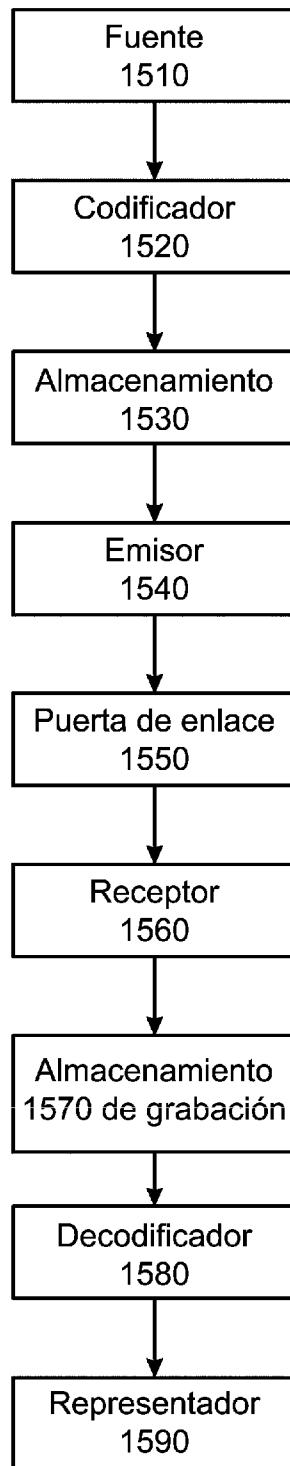


Figura 8