

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 907 602**

51 Int. Cl.:

H04N 19/105 (2014.01)
H04N 19/136 (2014.01)
H04N 19/172 (2014.01)
H04N 19/187 (2014.01)
H04N 19/30 (2014.01)
H04N 19/33 (2014.01)
H04N 19/46 (2014.01)
H04N 19/70 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **29.12.2015** **PCT/IB2015/060036**
87 Fecha y número de publicación internacional: **07.07.2016** **WO16108188**
96 Fecha de presentación y número de la solicitud europea: **29.12.2015** **E 15875349 (1)**
97 Fecha y número de publicación de la concesión europea: **24.11.2021** **EP 3241352**

54 Título: **Predicción entre capas para la codificación y decodificación de vídeo escalable**

30 Prioridad:

31.12.2014 US 201462098919 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
25.04.2022

73 Titular/es:

NOKIA TECHNOLOGIES OY (100.0%)
Karakaari 7
02610 Espoo, FI

72 Inventor/es:

HANNUKSELA, MISKA MATIAS

74 Agente/Representante:

DEL VALLE VALIENTE, Sonia

ES 2 907 602 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

Predicción entre capas para la codificación y decodificación de vídeo escalable

5 Campo técnico

La presente invención se refiere a un método, un aparato y un medio legible por ordenador para decodificar un primer y segundo conjunto de imágenes de una primera y segunda representación de vídeo codificada, respectivamente, y un método que comprende proporcionar, en un formato de metapresentación, metadatos para al menos un primer y segundo conjunto de imágenes de una primera y segunda representación de vídeo codificada, respectivamente.

Antecedentes

Esta sección está destinada a proporcionar antecedentes o contexto a la invención que se menciona en las reivindicaciones. Por la presente memoria, la descripción puede incluir conceptos que podrían llevarse a cabo, pero que no son necesariamente los que se han concebido o llevado a cabo previamente. Por lo tanto, a menos que se indique lo contrario en la presente memoria, lo que se describe en esta sección no es una técnica anterior a la descripción y las reivindicaciones de esta solicitud y no se admite como una técnica anterior mediante la inclusión en esta sección.

La codificación de vídeo escalable se refiere a la estructura de codificación en la que una secuencia de bits puede contener múltiples representaciones del contenido a diferentes velocidades de bits, de resoluciones o de marcos. En estos casos, el receptor puede extraer la representación deseada dependiendo de sus características. Alternativamente, un servidor o un elemento de la red puede extraer las porciones de la secuencia de bits que deben transmitirse al receptor dependiendo de, por ejemplo, las características de la red o de las capacidades de procesamiento del receptor. Una secuencia de bits escalable consiste típicamente en una capa base que proporciona la calidad de vídeo más baja disponible y una o varias capas de mejora que aumentan la calidad de vídeo cuando se reciben y decodifican junto con las capas inferiores. Para mejorar la eficacia de la codificación de las capas de mejora, la representación codificada de esa capa depende típicamente de las capas inferiores.

Para que un cliente pueda cambiar entre diferentes calidades y resoluciones durante una sesión de transmisión, se pueden utilizar imágenes codificadas de punto de acceso aleatorio en los límites del segmento. Convencionalmente, solo se han utilizado imágenes de punto de acceso aleatorio instantáneo (RAP), como la imagen de renovación de decodificación instantánea (IDR), que inicia una estructura de predicción denominada grupo cerrado de imágenes (GOP) en los límites de los segmentos de las representaciones de transmisión dinámica adaptativa a través de HTTP (DASH). El uso de imágenes intra que parten de GOP abiertos, por ejemplo las imágenes de acceso aleatorio limpio (CRA) en H.265/HEVC, se ha mejorado en H.265/HEVC en comparación con los estándares más antiguos, ya que se ha especificado normativamente un proceso de decodificación que parte de una imagen CRA. Cuando la decodificación se inicia a partir de una imagen CRA, algunas imágenes, denominadas de acceso aleatorio omitido (RASL), que siguen a la imagen CRA en el orden de decodificación pero que anteceden a la imagen CRA en el orden de salida pueden no ser decodificables. En consecuencia, si se utilizaran GOP abiertos en los límites de los segmentos en DASH, el cambio de representación podría dar lugar a la incapacidad de decodificar las imágenes RASL y, por tanto, a un fallo de la velocidad de imágenes en la reproducción. Por ejemplo, si se utilizara una jerarquía de predicción de 8 imágenes y la velocidad de las imágenes fuera de 25 Hz, el vídeo se congelaría durante aproximadamente un tercio de segundo.

La solicitud de la patente EP 2-723-079 se refiere a la predicción entre capas. Las diferentes representaciones del mismo contenido pueden recibirse en diferentes capas, en donde se puede decidir si se aplica una predicción entre capas a una capa de potencia; y si es así, la predicción en un bloque actual de la capa de mejora puede realizarse basándose en la información de referencia que se generaliza y se genera a partir de una imagen de referencia, que se decodifica, de una capa de referencia, cuando se aplica la predicción entre capas. En otras palabras, las diferentes representaciones contienen temporalmente el mismo contenido pero con diferente calidad de imagen, resolución, tamaño, velocidad de marcos, etc.

Resumen

Ahora, con el fin de reducir al menos los problemas anteriores, se presentan en la presente memoria imágenes de acceso intraaleatorio codificadas por omisión.

Un método según una primera realización que comprende la decodificación de un primer conjunto de imágenes de una primera representación de vídeo codificada y la decodificación de al menos un segundo conjunto de imágenes de una segunda representación de vídeo codificada como se reivindica en la reivindicación 1.

Un aparato según una segunda realización que comprende medios para decodificar un primer conjunto de imágenes de una primera representación de vídeo codificada y medios para decodificar al menos un segundo conjunto de imágenes de una segunda representación de vídeo codificada, como se reivindica en la reivindicación 9.

Según una tercera realización, se proporciona un medio de almacenamiento legible por ordenador almacenado con un código en el mismo para ser utilizado por un aparato, que cuando es ejecutado por un procesador, hace que el aparato decodifique un primer conjunto de imágenes de una primera representación de vídeo codificada y decodifique al menos un segundo conjunto de imágenes de una segunda representación de vídeo codificada, como se reivindica en la reivindicación 17.

Según una cuarta realización se proporciona un método que comprende proporcionar, en una descripción de presentación de medios, metadatos para al menos un primer conjunto de imágenes de una primera representación de vídeo codificada y un segundo conjunto de imágenes de una segunda representación de vídeo codificada, como se reivindica en la reivindicación 18.

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 las realizaciones de la invención;

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

La Figura 3 muestra además esquemáticamente los dispositivos electrónicos que emplean las realizaciones de la invención, conectados mediante conexiones de red inalámbricas y por cable;

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

La Figura 5 muestra un ejemplo de una imagen compuesta por dos mosaicos;

La Figura 6 representa algunos detalles de ciertas representaciones de transmisión dinámica adaptativa a través de HTTP;

La Figura 7 muestra un ejemplo de cómo pueden usarse imágenes SP para conmutar entre dos flujos de bits diferentes;

La Figura 8a muestra un ejemplo de dos flujos de bits que tienen imágenes principales:

La Figura 8b ilustra un ejemplo de una situación en la que la decodificación de imágenes principales se omite durante la conmutación de una secuencia de bits a otra corriente de bits;

La Figura 8c ilustra un ejemplo de una situación en la que una imagen I de dos flujos de bits se recibe y decodifica para conmutar de una secuencia de bits a otra corriente de bits;

La Figura 8d ilustra un ejemplo de una situación en la que la decodificación de imágenes principales del otra secuencia de bits se omite durante la conmutación de una secuencia de bits a otra corriente de bits;

La Figura 8e ilustra un ejemplo de operación de decodificación durante la conmutación de una secuencia de bits a otra secuencia de bits, según una realización;

La Figura 9a ilustra un ejemplo de codificación y preparación de segmentos para un cambio de resolución adaptativa redundante, según una realización;

La Figura 9b ilustra una operación del método de cambio de resolución adaptativa redundante durante la conmutación ascendente, según una realización;

La Figura 9c ilustra un ejemplo de codificación y preparación de segmentos para el método de remuestreo de imágenes de referencia implícito, según una realización;

La Figura 9d ilustra una operación del método de remuestreo de imágenes de referencia implícito durante la conmutación ascendente, según una realización;

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

La Figura 11 muestra un diagrama de flujo de un método de decodificación según una realización de la invención;

La Figura 12 muestra un diagrama esquemático de un decodificador adecuado para implementar las realizaciones de la invención;

La Figura 13 muestra el diagrama esquemático de un ejemplo de sistema de comunicación multimedia dentro del cual pueden implementarse diversas realizaciones.

Descripción detallada de algunas realizaciones ilustrativas

A continuación se describe en más detalle un aparato adecuado y los mecanismos posibles para cambiar entre representaciones de vídeo heterogéneas en, por ejemplo, la transmisión dinámica adaptativa a través de HTTP. 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 ejemplar o dispositivo electrónico 50, que puede incorporar un códec según una realización de la invención. La Figura 2 muestra el diseño de un aparato según una realización ilustrativa. Los elementos de las figuras 1 y 2 se explicarán a continuación.

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 decodificación o decodificación de imágenes de vídeo.

El aparato 50 puede comprender una carcasa 30 para incorporar y proteger el dispositivo. El aparato 50 puede comprender además una pantalla 32 en forma de 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 34. En otras realizaciones de la invención se puede emplear cualquier mecanismo adecuado de datos o de interfaz de usuario. Por ejemplo, la interfaz de usuario puede implementarse como un teclado virtual o un sistema de entrada de datos como parte de una pantalla táctil.

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 cualquiera de: un auricular 38, un altavoz o una conexión de salida de audio analógica o de audio digital. El aparato 50 también puede comprender una batería 40 (o en otras realizaciones de la invención el aparato puede ser alimentado por cualquier dispositivo de energía móvil adecuado, tal como una célula solar, una célula de combustible o un generador de relojería). El aparato puede comprender además una cámara 42 capaz de grabar o capturar imágenes y/o vídeo. El aparato 50 puede comprender además un puerto de infrarrojos para la comunicación de visión de corto alcance con 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 por cable USB/firewire.

El aparato 50 puede comprender un controlador 56 o un procesador para controlar el aparato 50. El controlador 56 puede estar conectado a la memoria 58 que, en las realizaciones de la invención, puede almacenar tanto datos en forma de imagen como de audio y/o también puede almacenar instrucciones para su implementación en el controlador 56. El controlador 56 puede estar conectado además al sistema de circuitos del códec 54, adecuada para llevar a cabo la codificación y decodificación de los datos de audio y/o vídeo, o para ayudar a la codificación y decodificación realizadas por el controlador.

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

El aparato 50 puede comprender un sistema de circuitos de interfaz de radio 52 conectado al controlador y adecuado para generar señales de comunicación inalámbrica, por ejemplo, para la comunicación con una red de comunicaciones móviles, un sistema de comunicaciones inalámbricas o una red de área local inalámbrica. El aparato 50 puede comprender además una antena 44 conectada al sistema de circuitos de interfaz de radio 52 para transmitir señales de radiofrecuencia generadas en el circuito de interfaz de radio 52 a otro(s) aparato(s) y para recibir señales de radiofrecuencia de otro(s) aparato(s).

El aparato 50 puede incluir una cámara capaz de grabar o detectar marcos individuales que luego 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 su transmisión y/o almacenamiento. El aparato 50 también puede recibir de forma inalámbrica o mediante una conexión por cable la imagen para su codificación/decodificación.

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, pero no se limita a, una red de telefonía móvil inalámbrica (como una red GSM, UMTS, CDMA, etc.), una red de área local inalámbrica (WLAN) tal como la definida por cualquiera de las normas IEEE 802.x, una red de área personal Bluetooth, una red de área local Ethernet, una red de área local token ring, una red de área amplia e internet.

El sistema 10 puede incluir dispositivos y/o aparatos 50 de comunicación, tanto cableados como inalámbricos, adecuados para implementar las realizaciones de la invención.

5 Por ejemplo, el sistema mostrado en la Figura 3 muestra una red de telefonía móvil 11 y una representación de internet 28. La conectividad a internet 28 puede incluir, entre otras, conexiones inalámbricas de largo alcance, conexiones inalámbricas de corto alcance y diversas conexiones cableadas, pero no se limita a, líneas telefónicas, líneas de cable, líneas eléctricas y vías de comunicación similares.

10 Los dispositivos de comunicación a modo de ejemplo que se muestran en el sistema 10, pero no se limitan a, un dispositivo o aparato electrónico 50, una combinación de asistente personal digital (PDA) y teléfono móvil 14, un PDA 16, un dispositivo de mensajería integrado (IMD) 18, un ordenador de sobremesa 20, un ordenador portátil 22. El aparato 50 puede ser estacionario o móvil cuando es transportado por un individuo que está en movimiento. El aparato 50 también puede estar situado en un medio de transporte, pero no se limita a, un coche, un camión, un taxi, un autobús, un tren, un barco, un avión, una bicicleta, una motocicleta o cualquier otro medio de transporte adecuado similar.

20 Las realizaciones también pueden implementarse en un decodificador; es decir, en un receptor de televisión digital, que puede o no tener una pantalla o capacidades inalámbricas, en tabletas u ordenadores personales (portátiles), que tienen hardware o software o una combinación de las implementaciones del codificador/decodificador, en varios sistemas operativos, y en conjuntos de chips, procesadores, DSP y/o sistemas integrados que ofrecen codificación basada en hardware/software.

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

30 Los dispositivos de comunicación pueden comunicarse utilizando diversas tecnologías de transmisión, pero no se limitan a, el acceso múltiple por división de código (CDMA), los sistemas globales de comunicaciones móviles (GSM), el sistema universal de telecomunicaciones móviles (UMTS), el acceso múltiple por división de tiempo (TDMA), el acceso múltiple por división de frecuencia (FDMA), el protocolo de control de la transmisión- protocolo de internet (TCP-IP), el servicio de mensajería corta (SMS), el servicio de mensajería multimedia (MMS), el correo electrónico, el servicio de mensajería instantánea (IMS), Bluetooth, IEEE 802.11 y cualquier otra tecnología de comunicación inalámbrica similar. Un dispositivo de comunicaciones involucrado en la implementación de varias realizaciones de la presente invención puede comunicarse utilizando varios medios, pero no se limita a, la radio, los infrarrojos, el láser, las conexiones por cable y cualquier conexión adecuada.

40 El códec de vídeo consta de un codificador que transforma el vídeo de entrada en una representación comprimida adecuada para su almacenamiento/transmisión y un decodificador que puede descomprimir la representación de vídeo comprimida de nuevo en una forma visible. Un codificador de vídeo y/o un decodificador de vídeo también pueden estar separados entre sí, es decir, no necesitan formar un códec. Normalmente, el codificador descarta parte de la información de la secuencia de vídeo original para representar el vídeo de una forma más compacta (es decir, a una velocidad de bits inferior).

45 Los codificadores de vídeo híbrido típicos, como por ejemplo muchas implementaciones de codificador de ITU-T H.263 y H.264, codifican la información de vídeo en dos fases. En primer lugar, los valores de los píxeles de una determinada zona de la imagen (o "bloque") se predicen, por ejemplo, por medios de compensación de movimiento (encontrando e indicando una zona en uno de los marcos de vídeo previamente codificados que se corresponda estrechamente con el bloque que se va a codificar) o por medios espaciales (utilizando los valores de los píxeles alrededor del bloque que se va a codificar de una manera determinada). En segundo lugar, se codifica el error de predicción, es decir, la diferencia entre el bloque predicho de píxeles y el bloque original de píxeles. Esto se hace típicamente transformando la diferencia de los valores de los píxeles mediante una transformación específica (por ejemplo, la transformada discreta del coseno (DCT) o una variante de esta), cuantificando los coeficientes y codificando con 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 los píxeles (calidad de la imagen) y el tamaño de la representación de vídeo codificada resultante (tamaño del archivo o de la velocidad de bits de transmisión).

60 La interpredicción, que también puede denominarse predicción temporal, compensación del movimiento o predicción compensada por el 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 los valores de la muestra o los coeficientes de la transformada. La intrapredicción se aprovecha típicamente en la intercodificación, donde no se aplica la interpredicción.

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

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

Dependiendo de qué modo de codificación se selecciona 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 de modo 310, 410. El selector de modo 410 también recibe una copia de la imagen de la capa de mejora 400. El primer dispositivo sumador puede restar la salida del predictor de píxeles 302, 402 de la imagen de la capa base 300/de la capa de mejora 400 para producir una primera señal de errores de predicción 320, 420 que se introduce en el codificador de errores de predicción 303, 403.

El predictor de píxeles 302, 402 recibe además de un reconstructor preliminar 339, 439 la combinación de la representación de predicción del bloque de imagen 312, 412 y la salida 338, 438 del decodificador de errores de predicción 304, 404. La imagen reconstruida preliminar 314, 414 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 reconstruida final 340, 440 que puede guardarse en una memoria de marco de referencia 318, 418. La memoria de marco de referencia 318 puede estar conectada al interpredictor 306 para ser utilizada como imagen de referencia con la que se compara una futura imagen de la capa base 300 en las operaciones de interpredicción. Si se selecciona la capa base y se indica que es la fuente para la predicción de la muestra entre capas y/o la predicción de la información de movimiento entre capas de la capa de mejora, según algunas realizaciones, la memoria de marco de referencia 318 puede conectarse también al interpredictor 406 para utilizarla como imagen de referencia con la que se comparan las futuras imágenes de la capa de mejora 400 en las operaciones de interpredicción. Además, la memoria de marco de referencia 418 puede estar conectada al interpredictor 406 para ser utilizada como imagen de referencia con la que se compara una futura imagen de la capa de mejora 400 en las operaciones de interpredicción.

Los parámetros de filtrado del filtro 316 de la primera sección del codificador 500 pueden proporcionarse a la segunda sección del codificador 502 sometida a la capa base seleccionada e indicada para ser fuente para predecir los parámetros de filtración de la capa de mejora según algunas realizaciones.

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

El decodificador de errores de predicción 304, 404 recibe la salida del codificador de errores de predicción 303, 403 y realiza los procesos opuestos del codificador de errores de predicción 303, 403 para producir una señal de errores de predicción decodificada 338, 438 que, cuando se combina con la representación de predicción del bloque de imágenes 312, 412 en el segundo dispositivo sumador 339, 439, produce la imagen reconstruida preliminar 314, 414. Se puede considerar que el decodificador de errores de predicción comprende un descuantificador 361, 461, que descuantificará los valores de los coeficientes cuantificados, por ejemplo, los coeficientes DCT, para reconstruir la señal de transformación y una unidad de transformación inversa 363, 463, que realizará la transformación inversa a la señal de transformación reconstruida, en donde la salida de la unidad de transformación inversa 363, 463 contiene bloque(s) reconstruido(s). El decodificador de errores de predicción también puede comprender un filtro de bloques que puede filtrar el bloque(s) reconstruido(s) según información decodificada adicional y parámetros de filtro.

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

La norma Codificación de Vídeo Avanzada (H.264/AVC, también conocida como AVC) fue desarrollada por el Equipo Conjunto de Vídeo (JVT) del Grupo de Expertos en Codificación de Vídeo (VCEG) del Sector de Normalización de las Telecomunicaciones de la Unión Internacional de Telecomunicaciones (UIT-T) y el Grupo de Expertos en Imágenes en Movimiento (MPEG) de la Organización Internacional de Normalización (ISO) / Comisión Electrotécnica Internacional (IEC). La norma H.264/AVC ha sido publicada por ambas organizaciones principales de normalización, y se denomina Recomendación UIT-T H.264 y Norma Internacional ISO/IEC 14496-10, también conocida como MPEG-4 Parte 10 Codificación de Vídeo Avanzada (AVC). Ha habido múltiples versiones de la norma H.264/AVC, integrando nuevas extensiones o características a la especificación. Estas extensiones incluyen la Codificación de Vídeo Escalable (SVC) y la Codificación de Vídeo Multivisión (MVC).

La versión 1 de la norma de codificación de vídeo de alta eficiencia (H.265/HEVC, también conocida como HEVC) fue desarrollada por el Equipo de Colaboración Conjunta - Codificación de Vídeo (JCT-VC) de VCEG y MPEG. La norma fue publicada por las dos organizaciones de normalización principales, y se denomina Recomendación UIT-T H.265 y Norma Internacional ISO/IEC 23008-2, también conocida como Codificación de Vídeo de Alta Eficiencia MPEG-H Parte 2 (HEVC). La versión 2 de H.265/HEVC incluyó extensiones escalables, multivista y de rango de fidelidad, que pueden abreviarse como SHVC, MV-HEVC y REXT, respectivamente. La versión 2 de H.265/HEVC fue prepublicada como Recomendación UIT-T H.265 (10/2014) y es probable que se publique como Edición 2 de ISO/IEC 23008-2 en 2015. Actualmente, hay proyectos de normalización en curso para desarrollar más extensiones de H.265/HEVC, incluyendo extensiones de codificación de contenidos tridimensionales y de pantalla, que pueden abreviarse como 3D-HEVC y SCC, respectivamente.

SHVC, MV-HEVC y 3D-HEVC utilizan una especificación base común, especificada en el Anexo F de la versión 2 de la norma HEVC. Esta base común comprende, por ejemplo, una sintaxis y una semántica de alto nivel que especifica, por ejemplo, algunas de las características de las capas de la secuencia de bits, tal como las dependencias entre capas, así como los procesos de decodificación, como la construcción de la lista de imágenes de referencia, incluidas las imágenes de referencia entre capas, y la derivación del recuento del orden de las imágenes para la secuencia de bits multicapa. El anexo F también puede utilizarse en posibles extensiones posteriores de HEVC en varias capas. Debe entenderse que, aunque un codificador de vídeo, un decodificador de vídeo, los métodos de codificación, los métodos de decodificación, las estructuras de secuencia de bits y/o las realizaciones pueden describirse en lo que sigue con referencia a extensiones específicas, tal como SHVC y/o MV-HEVC, son generalmente aplicables a cualquier extensión multicapa de HEVC, e incluso más generalmente a cualquier esquema de codificación de vídeo multicapa.

En esta sección se describen algunas definiciones clave, estructuras de secuencia de bits y de codificación, y conceptos de H.264/AVC y HEVC como ejemplo de un codificador de vídeo, un decodificador, un método de codificación, un método de decodificación y una estructura de secuencia de bits, en donde se pueden implementar las realizaciones. Algunas de las definiciones clave, las estructuras de secuencia de bits y de codificación y los conceptos de H.264/AVC son los mismos que en HEVC, por lo que se describen a continuación de forma conjunta. Los aspectos de la invención no se limitan a H.264/AVC o HEVC, sino que la descripción se hace sobre una posible base en la que la invención puede realizarse parcial o totalmente.

Al igual que muchas normas de codificación de vídeo anteriores, la sintaxis y la semántica de la secuencia de bits, así como el proceso de decodificación para los flujos de bits sin errores, están especificados 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 la secuencia de bits y el decodificador se pueden verificar con el decodificador de referencia hipotético (HRD). Las normas contienen herramientas de codificación que ayudan a hacer frente a los 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 los flujos de bits erróneos.

En la descripción de las normas existentes, así como en la descripción de las realizaciones de ejemplo, un elemento de sintaxis puede definirse como un elemento de datos representado en la secuencia de bits. Una estructura sintáctica puede definirse como cero o más elementos sintácticos presentes juntos en la secuencia de bits en un orden especificado. En la descripción de las normas existentes, así como en la descripción de realizaciones ilustrativas, se puede usar la frase “por medios externo” o “a través de medios externos”. Por ejemplo, una entidad, tal como una estructura sintáctica o un valor de una variable usada en el proceso de decodificación, puede proporcionarse “por medios externos” al proceso de decodificación. La frase “por medio externo” puede indicar que la entidad no se incluye en la secuencia de bits creado por el codificador, sino que se transmite externamente desde la secuencia de bits, por ejemplo, utilizando un protocolo de control. Puede significar, alternativa o adicionalmente, que la entidad no es creada por el codificador, sino que puede ser creada, por ejemplo, en el reproductor o en la lógica de control de decodificación o similar que está utilizando el decodificador. El decodificador puede tener una interfaz para introducir los medios externos, tales como valores variables.

Un perfil puede definirse como un subconjunto de toda la sintaxis de secuencia de bits que se especifica mediante una norma o especificación de decodificación/codificación. Dentro de los límites impuestos por la sintaxis de un perfil dado, aún es posible requerir una variación muy grande en el rendimiento de codificadores y decodificadores dependiendo de los valores tomados por elementos de sintaxis en la secuencia de bits, tal como el tamaño especificado de las imágenes decodificadas. En muchas aplicaciones, podría no ser práctico ni económico implementar un decodificador capaz de tratar con todos los usos hipotéticos de la sintaxis dentro de un perfil particular. Para solucionar este problema, se pueden utilizar niveles. Un nivel puede definirse como un conjunto especificado de restricciones impuestas sobre los valores de los elementos sintácticos en la secuencia de bits y las variables especificadas en una norma o especificación de decodificación/codificación. Estas restricciones pueden ser simples límites de valores. De forma alternativa o adicional, pueden adoptar la forma de restricciones sobre combinaciones aritméticas de valores (por ejemplo, la anchura de la imagen multiplicada por la altura de la imagen multiplicada por el número de imágenes decodificadas por segundo). También pueden utilizarse otros medios para especificar las restricciones de los niveles. Algunas de las restricciones especificadas en un nivel pueden referirse, por ejemplo, al tamaño máximo de la imagen, a la velocidad de bits máxima y a la velocidad de datos máxima en términos de unidades de codificación, como los macrobloques, por un período de tiempo, como un segundo. Se puede definir el mismo conjunto de niveles para todos los perfiles. Puede ser preferible, por ejemplo, para aumentar la interoperabilidad de los terminales que implementan diferentes perfiles, que la mayoría o todos los aspectos de la definición de cada nivel puedan ser comunes entre los diferentes perfiles. Un nivel puede definirse como una categoría específica de restricciones de nivel impuestas a los valores de los elementos de sintaxis en la secuencia de bits, donde las restricciones de nivel están anidadas dentro de un nivel y un decodificador conforme a un determinado nivel y nivel sería capaz de decodificar todas las secuencias de bits que se ajusten al mismo nivel o al nivel inferior de ese nivel o a cualquier nivel inferior.

En algunos casos, un punto de conformidad puede definirse como una combinación de un perfil particular y un nivel particular o una combinación de un perfil particular, un nivel particular y un nivel particular. Debe entenderse que un punto de conformidad puede definirse de formas alternativas, mientras que su intención de especificar las características y los límites de la secuencia de bits y/o las características y los recursos (máximos) de los decodificadores puede mantenerse sin cambios.

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 puede denominarse también imagen fuente, y una imagen decodificada por un decodificador puede denominarse imagen decodificada.

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

Solo Luma (Y) (monocromo).

Luma y dos cromas (YCbCr o YCgCo).

Verde, azul y rojo (GBR, también conocido como RGB).

Matrices que representan otros muestreos de colores monocromos o triestímulos no especificados (por ejemplo, YZX, también conocido como XYZ).

A continuación, estas matrices pueden denominarse luma (o L o Y) y croma, donde las dos matrices del croma pueden denominarse Cb y Cr; independientemente del método de representación del color que se utilice. El método de representación del color en uso puede indicarse, por ejemplo, en una secuencia de bits codificado, utilizando la sintaxis de la información de uso del vídeo (VUI) de H.264/AVC y/o HEVC. Un componente puede definirse como una matriz o una sola muestra de una de las tres matrices de muestras (luma y dos cromas) o la matriz o una sola muestra de la matriz que componen una imagen en formato monocromo.

En H.264/AVC y HEVC, una imagen puede ser un marco o un campo. Un marco comprende una matriz de muestras de luma y posiblemente las correspondientes muestras de croma. Un campo es un conjunto de filas de muestras alternas de un marco y puede utilizarse como entrada del codificador, cuando la señal de origen está entrelazada.

- 5 Las matrices de muestras de croma pueden estar ausentes (y, por lo tanto, puede utilizarse el muestreo monocromático) o las matrices de muestras de croma pueden estar submuestreadas en comparación con las matrices de muestras de luma. Los formatos de croma pueden resumirse como sigue:

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

- 10 En el muestreo 4:2:0, cada una de las dos matrices de croma tiene la mitad de la altura y la mitad de la anchura de la matriz de luma.

- 15 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 se utilizan planos de color separados, cada una de las dos matrices de croma tiene la misma altura y anchura que la matriz de luma.

- 20 En H.264/AVC y HEVC, es posible codificar matrices de muestras como planos de color separados en la secuencia de bits y, respectivamente, decodificar planos de color codificados por separado desde la secuencia de bits. Cuando se utilizan 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.

- 25 Cuando se utiliza el submuestreo de croma (por ejemplo, el muestreo de croma 4:2:0 o 4:2:2), la ubicación de las muestras de croma con respecto a las muestras de luma puede determinarse en el lado del codificador (por ejemplo, como paso de preprocesamiento o como parte de la codificación). Las posiciones de las muestras de croma con respecto a las posiciones de las muestras de luma pueden estar predefinidas, por ejemplo, en una norma de codificación, como H.264/AVC o HEVC, o pueden indicarse en la secuencia de bits, por ejemplo, como parte de la VUI de H.264/AVC o HEVC.

Una partición puede definirse como una división de un conjunto en subconjuntos tal que cada elemento del conjunto está exactamente en uno de los subconjuntos.

- 35 En H.264/AVC, un macrobloque es un bloque de 16x16 muestras de luma y los correspondientes bloques de muestras de croma. Por ejemplo, en el patrón de muestreo 4:2:0, un macrobloque contiene un bloque 8x8 de muestras de croma por cada componente de croma. En H.264/AVC, una imagen se divide en uno o más grupos de fragmentos, y un grupo de fragmentos contiene uno o más fragmentos. En H.264/AVC, un fragmento consiste en un número entero de macrobloques ordenados consecutivamente en el barrido de fotograma dentro de un grupo de fragmentos determinados.

Cuando se describe la operación de codificación y/o decodificación de HEVC, se pueden utilizar los siguientes términos. Un bloque de codificación puede definirse como un bloque de muestras NxN para algún valor de N, de manera 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 muestras NxN para algún valor de N tal que la división de un componente en bloques de árbol de codificación es una partición. Una unidad de árbol de codificación (UTC) puede definirse como un bloque de árbol de codificación de muestras de luma, dos bloques correspondientes de árbol de codificación 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 monocromática o una imagen que se codifica utilizando tres planos de color separados y estructuras de sintaxis utilizadas 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 monocromática o una imagen que se codifica utilizando tres planos de color separados y estructuras de sintaxis utilizadas para codificar las muestras.

- 55 En algunos códecs de vídeo, tales como el códec de Codificación de Vídeo de Alta Eficiencia (HEVC), las imágenes de vídeo se dividen en unidades de codificación (CU) que cubren el área de la imagen. Una CU consta de 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 transformación (TU) que definen el proceso de codificación de errores de predicción para las muestras en dicha CU. Normalmente, una CU consiste en un bloque cuadrado de muestras con un tamaño seleccionable entre un conjunto predefinido de posibles tamaños de CU. Una CU con el tamaño máximo permitido puede denominarse LCU (unidad de codificación más grande) o unidad de árbol de codificación (CTU) y la imagen de vídeo se divide en LCU no solapadas. Una LCU puede dividirse a su vez en una combinación de CU más pequeñas, por ejemplo, dividiendo recursivamente la LCU y las CU resultantes. Cada CU resultante tiene típicamente asociadas al menos una PU y al menos una TU. Cada PU y TU puede dividirse a su vez en varias PU y TU más pequeñas para aumentar la granularidad de los procesos de predicción y codificación de errores de predicción, respectivamente. Cada PU tiene información de predicción

asociada con la que define qué tipo de predicción se debe aplicar para los píxeles dentro de esa PU (por ejemplo, información del vector de movimiento para PU interpredichas e información de direccionalidad de intrapredicción para PU intrapredichas).

5 Cada TU puede asociarse con información que describa el proceso de decodificación del error de predicción para las muestras dentro de dicha TU (incluyendo, por ejemplo, información sobre el coeficiente DCT). Normalmente se señala a nivel de la CU si se aplica o no la codificación de errores de predicción para cada CU. En el caso de que no haya ningún residuo de error de predicción asociado a la CU, se puede considerar que no hay TU para dicha CU. La división de la imagen en CU, y la división de las CU en PU y TU se señala normalmente en la
10 secuencia de bits permitiendo al decodificador reproducir la estructura prevista de estas unidades.

En HEVC, una imagen puede dividirse en mosaicos, que son rectangulares y contienen un número entero de LCU. En HEVC, la partición en mosaicos forma una cuadrícula regular, las alturas y anchuras de los mosaicos difieren entre sí en una LCU como máximo. En HEVC, se define un fragmento que es un número entero de
15 unidades de árbol de codificación contenidas en un fragmento de segmento independiente y todos los fragmentos de segmento dependientes posteriores (si los hay) que anteceden al siguiente fragmento de segmento independiente (si lo hay) dentro de la misma unidad de acceso. En HEVC, un fragmento de segmento se define como un número entero de unidades de árbol de codificación ordenadas consecutivamente en la exploración de mosaico y contenida en una única unidad de NAL. La división de cada imagen en segmentos de fragmento es una
20 partición. En HEVC, un fragmento de segmento independiente se define como un fragmento de segmento para el cual los valores de los elementos de sintaxis de la cabecera de fragmento de segmento no se infieren de los valores para un segmento de segmento anterior, y un fragmento de segmento dependiente se define como un fragmento de segmento para el cual los valores de algunos elementos de sintaxis de la cabecera de fragmento de segmento se infieren de los valores para el fragmento de segmento independiente anterior en orden de
25 decodificación. En HEVC, una cabecera de fragmento se define como la cabecera de fragmento de segmento del segmento independiente que es un fragmento del segmento actual o es el fragmento de segmento independiente que antecede a un fragmento de segmento dependiente de corriente, y una cabecera de fragmento de segmento se define como una parte de un fragmento de segmento codificado que contiene los elementos de datos pertenecientes a la primera o todas las unidades de árbol de codificación representadas en el fragmento de
30 segmento. Las CU se escanean en el orden de barrido por trama de LCU dentro de los mosaicos o dentro de una imagen, si no se usan mosaicos. Dentro de una LCU, las CU tienen un orden de exploración específico. La Figura 5 muestra un ejemplo de una imagen que consiste en dos mosaicos divididos en unidades de codificación cuadrada (líneas continuas) que se han dividido adicionalmente en unidades de predicción rectangulares (líneas discontinuas).

35 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 información espacial creada por el codificador y almacenado en la representación comprimida) y la decodificación de errores de predicción (operación inversa de la codificación de errores de predicción recuperando la señal de errores de predicción cuantificada en el dominio de píxeles
40 espaciales). Después de aplicar la predicción y la decodificación de errores de predicción, el decodificador suma la predicción y las señales de errores de predicción (valores de píxel) para formar el marco del 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 las tramas vecinas en la secuencia de vídeo.

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

50 En códecs de vídeo típicos, la información de movimiento se indica con vectores de movimiento asociados con cada bloque de imágenes 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 decodificado (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
55 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, calcular la mediana de los vectores de movimiento codificados o decodificados de los bloques adyacentes. Otra forma de crear predicciones de los vectores de movimiento consiste en generar una lista de predicciones candidatas a partir de bloques adyacentes y/o de bloques cúbicados en imágenes de
60 referencia temporal y señalar el candidato elegido como predictor de los vectores de movimiento. Además de predecir los valores del vector de movimiento, puede predecirse qué 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 cúbicados en la imagen de referencia temporal.
65 Además, los códecs de vídeo típicos de alta eficiencia emplean un mecanismo adicional de codificación/decodificación de la información de movimiento, a menudo denominado modo de fusión/mezcla, en el

que toda la información del 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 utiliza sin ninguna modificación/corrección. Del mismo modo, la predicción de la información del campo de movimiento se lleva a cabo utilizando la información del campo de movimiento de los bloques adyacentes y/o de los bloques

5 coubicados en las imágenes de referencia temporal y la información del campo de movimiento utilizada se señala entre una lista de candidatos al campo de movimiento rellena con la información del campo de movimiento de los bloques adyacentes/coubicados disponibles.

Los códecs de vídeo típicos permiten utilizar la unipredicción, en la que se utiliza un único bloque de predicción para un

10 bloque que se está (des)codificando, y la bipredicción, en la que se combinan dos bloques de predicción para formar la predicción de un bloque que se está (des)codificando. Algunos códecs de vídeo permiten la predicción ponderada, en la que los valores de muestra de los bloques de predicción se ponderan antes de añadir la información residual. Por ejemplo, se puede aplicar un factor de ponderación multiplicativo y un desplazamiento aditivo. En la predicción ponderada explícita, habilitada por algunos códecs de vídeo, se puede codificar un factor de ponderación y un desplazamiento, por ejemplo, en

15 la cabecera de la porción para cada índice de imagen de referencia admisible. En la predicción ponderada implícita, habilitada por algunos códecs de vídeo, los factores de ponderación y/o los desplazamientos no se codifican, sino que se derivan, por ejemplo, en función de las distancias relativas de recuento de imágenes (POC) de las imágenes de referencia.

En los códecs de vídeo típicos, el residuo de predicción tras la compensación de movimiento se transforma primero con un kernel de transformada (como la DCT) y luego se codifica. La razón es que a menudo todavía existe cierta correlación entre los residuos y la transformación puede, en muchos casos, ayudar a reducir esta correlación y proporcionar una codificación más eficaz.

20

Los codificadores de vídeo típicos utilizan funciones de coste lagrangianas para encontrar los modos de codificación óptimos, por ejemplo, el modo de macrobloque deseado y los vectores de movimiento asociados. Este tipo de función de costes utiliza un factor de ponderación λ para unir la distorsión (exacta o estimada) de la imagen debida a los métodos de codificación con pérdidas y la cantidad (exacta o estimada) de información que se necesita para representar los valores de los píxeles en una zona de la imagen:

25

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

30

donde C es el coste lagrangiano que hay que minimizar, D es la distorsión de la imagen (por ejemplo, el error cuadrático medio) con el modo y los vectores de movimiento considerados, y R el número de bits necesarios para representar los datos requeridos para reconstruir el bloque de la imagen en el decodificador (incluyendo la

35 cantidad de datos para representar los vectores de movimiento candidatos).

Las normas y especificaciones de codificación de vídeo pueden permitir a los codificadores dividir una imagen codificada en fragmentos codificados o similares. La predicción en imagen está desactivada típicamente en los límites de los fragmentos. Así, los fragmentos pueden considerarse como una forma de dividir una imagen codificada en fragmentos

40 decodificables de forma independiente. En H.264/AVC y HEVC, la predicción en imagen puede desactivarse a través de los límites de fragmento. Por lo tanto, los fragmentos se pueden considerar como una forma de dividir una imagen codificada en fragmentos decodificables, y los fragmentos, por lo tanto, a menudo se consideran unidades elementales para la transmisión. En muchos casos, los codificadores pueden indicar en la secuencia de bits qué tipos de predicción en imagen se apagan a través de límites de fragmento, y la operación del decodificador toma esta información en cuenta, por

45 ejemplo, cuando se concluye qué fuentes de predicción están disponibles. Por ejemplo, las muestras de un macrobloque o CU adyacente pueden considerarse no disponibles para intrapredicción, si el macrobloque vecino o CU reside en un fragmento 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 por redes orientadas a paquetes o el almacenamiento en archivos estructurados, las unidades NAL pueden encapsularse en paquetes o estructuras similares. Se ha especificado un formato de secuencia de bytes en H.264/AVC y HEVC para entornos de transmisión o almacenamiento que no proporcionan estructuras de encuadre. El formato de secuencia de bytes separa las unidades NAL entre sí uniéndolas con un código de inicio delante de cada unidad NAL. Para evitar la falsa

50 detección de los límites de la 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 la unidad NAL si se hubiera producido un código de inicio de otro modo. Para permitir un funcionamiento directo de la pasarela entre los sistemas orientados a paquetes y a secuencias, la prevención de la emulación del código de inicio puede realizarse siempre, independientemente de que se utilice o no el formato de secuencia de bytes. Una unidad NAL puede definirse

55 como una estructura de sintaxis que contiene una indicación del tipo de datos a seguir y bytes que contienen esos datos en forma de RBSP intercalados, según sea necesario, con bits de prevención de emulación. Una carga útil de secuencia de bytes sin procesar (RBSP) puede definirse como una estructura sintáctica que contiene un número entero de bytes que está encapsulado 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

60 posteriores iguales a 0.

65

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

La cabecera de la unidad de NAL H.264/AVC incluye un elemento sintáctico `nal_ref_idc` de 2 bits, que cuando es igual a 0 indica que un fragmento codificado contenido en la unidad NAL es una parte de una imagen no de referencia y cuando es mayor que 0 indica que un fragmento codificado contenido en la unidad NAL es una parte de una imagen de referencia. La cabecera de las unidades de NAL SVC y MVC puede contener adicionalmente diversas indicaciones relacionadas con la jerarquía de escalabilidad y multivista.

En HEVC, se usa una cabecera de la 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 un nivel temporal (puede ser necesario que sea mayor o igual a 1) y un elemento sintáctico `nuh_layer_id` de seis bits. El elemento sintáctico `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: $\text{TemporalId} = \text{temporal_id_plus1} - 1$. `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. La secuencia de bits creada excluyendo todas las unidades de la VCL NAL que tengan un `TemporalId` mayor o igual que un valor seleccionado e incluyendo todas las demás unidades de la VCL NAL sigue siendo conforme. En consecuencia, una imagen que tiene `TemporalId` igual a TID no usa ninguna imagen que tenga un `TemporalId` mayor que TID como referencia de interpretación. Una subcapa o subcapa temporal puede definirse como una capa temporal escalable de una secuencia de bits temporal escalable, compuesta por unidades NAL de la VCL con un valor particular de la variable `TemporalId` y las unidades NAL no VCL asociadas. El elemento sintáctico `nuh_layer_id` de HEVC puede llevar información sobre la jerarquía de escalabilidad.

Las unidades NAL pueden clasificarse en unidades NAL de Capa de Codificación de Vídeo (VCL) y unidades NAL non-VCL. Las unidades VCL NAL son típicamente unidades NAL de fragmentos codificados. En H.264/AVC, las unidades NAL de fragmentos codificados contienen elementos sintácticos que representan uno o más macrobloques codificados, cada uno de los cuales corresponde a un bloque de muestras de la imagen sin comprimir. En HEVC, las unidades VCL NAL contienen elementos sintácticos que representan una o más CU.

En H.264/AVC, una unidad NAL de segmento codificado puede indicarse como un fragmento codificado en una imagen de renovación de decodificación instantánea (IDR) o un fragmento codificado en una imagen no IDR.

En HEVC, una unidad NAL de fragmento codificado se puede indicar como uno de los siguientes tipos:

nal_unit_type	Nombre de nal_unit_type	Contenido de la unidad NAL y estructura sintáctica de la RBSP
0, 1	TRAIL_N, TRAIL_R	Fragmento de segmento codificado de un no TSA, Imagen final no STSA <code>slice_segment_layer_rbsp()</code>
2, 3	TSA_N, TSA_R	Fragmento de segmento codificado de una imagen TSA <code>slice_segment_layer_rbsp()</code>
4, 5	STSA_N, STSA_R	Fragmento de segmento codificado de una imagen STSA <code>slice_layer_rbsp()</code>
6, 7	RADL_N, RADL_R	Fragmento de segmento codificado de una imagen RADL <code>slice_layer_rbsp()</code>
8, 9	RASL_N, RASL_R	Fragmento de segmento codificado de una imagen RASL <code>slice_layer_rbsp()</code>
10, 12, 14	RSV_VCL_N10 RSV_VCL_N12 RSV_VCL_N14	Reservado // no RAP no reservado tipos de unidad VCL NAL de referencia
11, 13, 15	RSV_VCL_R11 RSV_VCL_R13 RSV_VCL_R15	Reservado // no RAP reservado tipos de unidad VCL NAL de referencia
16, 17, 18	BLA_W_LP BLA_W_DLP (a.k.a.	Fragmento de segmento codificado de una imagen BLA <code>slice_segment_layer_rbsp()</code>

	IDR_W_RADL) BLA_N_LP	
19,	IDR_W_DLP (a.k.a.	Fragmento de segmento codificado de una imagen IDR
20	IDR_W_RADL) IDR_N_LP	slice_segment_layer_rbsp()
21	CRA_NUT	Fragmento de segmento codificado de una imagen CRA slice_segment_layer_rbsp()
22,	RSV_IRAP_VCL22..	Reservado // RAP VCL NAL reservado
23	RSV_IRAP_VCL23	tipos de unidades
24..31	RSV_VCL24.. RSV_VCL31	Reservado // no RAP VCL reservado tipos de unidades NAL

En el HEVC, las abreviaturas de los tipos de imagen pueden definirse de la siguiente manera: imagen de arrastre (TRAIL), acceso temporal a la subcapa (TSA), acceso temporal a la subcapa por pasos (STSA), imagen de acceso aleatorio decodificable (RADL), imagen de acceso aleatorio omitido (RASL), imagen de acceso a enlaces rotos (BLA), imagen de renovación de decodificación instantánea (IDR), imagen de acceso aleatorio limpio (CRA).

Una imagen de punto de acceso aleatorio (RAP), que también puede denominarse imagen de punto de acceso intraaleatorio (IRAP), es una imagen en la que cada fragmento o fragmento de segmento tiene nal_unit_type en el rango de 16 a 23, ambos inclusive. Una imagen IRAP en una capa independiente contiene solo fragmentos intracodificados. Una imagen IRAP perteneciente a una capa predicha con nuh_layer_id value currLayerId puede contener los fragmentos P, B e I, pero no puede utilizar la predicción entre capas de otras imágenes con nuh_layer_id equal to currLayerId, y puede utilizar la predicción entre capas 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 de la secuencia de bits que contiene una capa base es una imagen IRAP. Siempre que los conjuntos de parámetros necesarios estén disponibles cuando deban activarse, una imagen IRAP en una capa independiente y todas las imágenes posteriores que no sean RASL en la capa independiente en orden de decodificación pueden decodificarse correctamente sin realizar el proceso de decodificación de ninguna imagen que anteceda a la imagen IRAP en orden de decodificación. Puede haber imágenes en una secuencia de bits que contengan solo fragmentos intracodificados que no sean imágenes RAP. La imagen IRAP perteneciente a una capa prevista con nuh_layer_id value currLayerId y todas las imágenes posteriores que no sean IRAP con nuh_layer_id equal to currLayerId en el orden de decodificación pueden decodificarse correctamente sin realizar el proceso de decodificación de ninguna imagen con nuh_layer_id equal to currLayerId que anteceda a la imagen IRAP en el orden de decodificación, cuando los conjuntos de parámetros necesarios están disponibles cuando deben ser activados y cuando la decodificación de cada capa de referencia directa de la capa con nuh_layer_id equal to currLayerId ha sido inicializada (es decir, cuando LayerInitializedFlag[refLayerId] is equal to 1 for refLayerId equal to all nuh_layer_id values of the direct reference layers of the layer with nuh_layer_id equal to currLayerId).

En HEVC, una imagen CRA puede ser la primera imagen en la secuencia de bits en orden de decodificación, o puede aparecer más adelante en la secuencia de bits. Las imágenes CRA en HEVC permiten las denominadas imágenes principales que siguen a la imagen CRA en el orden de decodificación pero la anteceden en el orden de salida. Algunas de las imágenes principales, las denominadas imágenes RASL, pueden utilizar imágenes decodificadas antes de la imagen CRA como referencia. Las imágenes que siguen a una imagen CRA tanto en el orden de decodificación como en el de salida pueden ser decodificables si el acceso aleatorio se realiza en la imagen CRA y, por tanto, se puede conseguir un acceso aleatorio limpio de forma similar a la funcionalidad de acceso aleatorio limpio de una imagen IDR.

Una imagen CRA puede tener imágenes RAL o RASL asociadas. Cuando una imagen CRA es la primera imagen en la secuencia de bits en orden de decodificación, la imagen CRA es la primera imagen de una secuencia de vídeo codificada en orden de decodificación, y cualquier imagen RASL asociada no se emite por el decodificador y puede no ser decodificable, ya que puede contener referencias a imágenes que no están presentes en la secuencia de bits.

Una imagen principal es una imagen que antecede a la imagen RAP asociada en orden de salida. La imagen RAP asociada es la imagen RAP anterior en orden de decodificación (si está presente). Una imagen principal puede ser una imagen RAL o una imagen RASL.

Todas las imágenes RASL son imágenes principales de una imagen BLA o CRA asociada. Cuando la imagen RAP asociada es una imagen BLA o es la primera imagen codificada en la secuencia de bits, la imagen RASL no se emite y puede no ser decodificable correctamente, ya que la imagen RASL puede contener referencias a imágenes que no están presentes en la secuencia de bits. Sin embargo, una imagen RASL puede decodificarse correctamente si la decodificación ha comenzado desde una imagen RAP antes de la imagen RAP asociada de la imagen RASL. Las imágenes RASL no se usan como imágenes de referencia para el proceso de decodificación de imágenes no RASL.

Cuando está presente, todas las imágenes RASL anteceden, en orden de decodificación, todas las imágenes posteriores de la misma imagen RAP asociada. En algunas versiones preliminares de la norma HEVC, se hizo referencia desde una imagen RASL a una imagen Marcada para ser Eliminada (TFD).

5 Todas las imágenes RADL son imágenes principales. Las imágenes RADL no se usan como imágenes de referencia para el proceso de decodificación de las imágenes posteriores de la misma imagen RAP asociada. Cuando está presente, todas las imágenes RADL anteceden, en orden de decodificación, todas las imágenes posteriores de la misma imagen RAP asociada. Las imágenes RADL no se refieren a ninguna imagen que antecede a la imagen RAP asociada en orden de decodificación y, por lo tanto, pueden decodificarse
10 correctamente cuando la decodificación comienza a partir de la imagen RAP asociada. En algunas versiones preliminares de la norma HEVC, se hizo referencia a una imagen RADL a una Imagen Principal Decodificable (DLP).

15 Cuando una parte de una secuencia de bits que comienza desde una imagen CRA se incluye en otra secuencia de bits, las imágenes RASL asociadas a la imagen CRA podrían no decodificarse correctamente, porque algunas de sus imágenes de referencia podrían no estar presentes en la secuencia de bits combinado. Para simplificar dicha operación de empalme, el tipo de unidad NAL de la imagen CRA puede cambiarse para indicar que es una imagen BLA. Las imágenes RASL asociadas con una imagen BLA pueden no decodificarse correctamente por lo tanto no se emiten/muestran. Además, las imágenes RASL asociadas con una imagen BLA pueden omitirse de la
20 decodificación.

Una imagen BLA puede ser la primera imagen en la secuencia de bits en orden de decodificación, o puede aparecer más adelante en la secuencia de bits. Cada imagen BLA comienza una nueva secuencia de vídeo codificada, y tiene un efecto similar en el proceso de decodificación como una imagen IDR. Sin embargo, una imagen BLA puede contener
25 elementos de sintaxis que especifican un conjunto de imágenes de referencia no vacío. Cuando una imagen BLA tiene `nal_unit_type` igual a `BLA_W_LP`, puede tener imágenes RASL asociadas, que no son emitidas por el decodificador y pueden no ser decodificables, ya que pueden contener referencias a imágenes que no están presentes en la secuencia de bits. Cuando una imagen BLA tiene `nal_unit_type` igual a `BLA_W_LP`, también puede tener imágenes RADL asociadas, que se especifican para decodificarse. Cuando una imagen BLA tiene `nal_unit_type` igual a `BLA_W_DLP`,
30 no tiene imágenes RASL asociadas pero puede tener imágenes RADL asociadas, que se especifican para decodificarse. Cuando una imagen BLA tiene `nal_unit_type` igual a `BLA_N_LP`, no tiene ninguna imagen principal asociada.

Una imagen IDR que tiene `nal_unit_type` igual a `IDR_N_LP` no tiene imágenes principales asociadas presentes en la secuencia de bits. Una imagen IDR que tiene `nal_unit_type` igual a `IDR_W_LP` no tiene imágenes RASL asociadas presentes en la secuencia de bits, pero puede tener imágenes RADL asociadas en la secuencia de bits.

40 Cuando el valor de `nal_unit_type` es igual a `TRAIL_N`, `tSA_N`, `sTSA_N`, `rADL_N`, `rASL_N`, `rSV_VCL_N` 10, `rSV_VCL_N` 12, o `RSV_VCL_N` 14, la imagen decodificada no se usa como referencia para ninguna otra imagen de la misma subcapa temporal. Es decir, en HEVC, cuando el valor de `nal_unit_type` es igual a `TRAIL_N`, `TSA_N`, `STSA_N`, `RADL_N`, `RASL_N`, `RSV_VCL_N` 10, `RSV_VCL_N` 12, or `RSV_VCL_N` 14, la imagen decodificada no se incluye en ninguno de `RefPicSetStCurrBefore`, `RefPicSetStCurrAfter` y `RefPicSetLtCurr` de ninguna imagen con el mismo valor de `TemporalId`. Se puede descartar una imagen codificada con `nal_unit_type` igual a `TRAIL_N`,
45 `TSA_N`, `STSA_N`, `RADL_N`, `RASL_N`, `RSV_VCL_N` 10, `RSV_VCL_N` 12, or `RSV_VCL_N` 14 sin afectar la capacidad de decodificación de otras imágenes con el mismo valor de `TemporalId`.

Una imagen posterior puede definirse como una imagen que sigue a la imagen RAP asociada en orden de salida. Cualquier imagen que sea una imagen posterior no tiene `nal_unit_type` igual a `RADL_N`, `RADL_R`, `RASL_N` o `RASL_R`.
50 Cualquier imagen que sea una imagen principal puede estar restringida a anteceder, en orden de decodificación, a todas las imágenes posteriores que están asociadas con la misma imagen RAP. No hay imágenes RASL presentes en la secuencia de bits que están asociadas a una imagen BLA que tiene `nal_unit_type` igual a `BLA_W_DLP` o `BLA_N_LP`. No hay imágenes RADL presentes en la secuencia de bits que están asociadas a una imagen BLA que tiene `nal_unit_type` igual a `BLA_N_LP` o que se asocian con una imagen IDR que tiene `nal_unit_type` igual a `IDR_N_LP`. Cualquier imagen
55 RASL asociada a una imagen CRA o BLA puede estar restringida para anteceder a cualquier imagen de RAL asociada con la imagen CRA o BLA en orden de salida. Cualquier imagen RASL asociada a una imagen CRA puede estar restringida a seguir, en orden de salida, cualquier otra imagen RAP que antecede a la imagen CRA en orden de decodificación.

60 En HEVC hay dos tipos de imágenes, los tipos de imágenes de TSA y STSA que pueden usarse para indicar puntos de conmutación de subcapa temporal. Si las subcapas temporales con `TemporalId` de hasta N habían sido decodificadas hasta la imagen de TSA o STSA (exclusiva) y la imagen de TSA o STSA tienen `TemporalId` igual a N+1, la imagen TSA o STSA permite decodificar todas las imágenes posteriores (en orden de decodificación) que tienen `TemporalId` igual a N+1. El tipo de imagen TSA puede imponer restricciones en la propia imagen de TSA y todas las imágenes en la misma subcapa que siguen a la imagen de TSA en orden de decodificación. No se permite que ninguna de estas imágenes utilicen interpredicción a partir de cualquier imagen en la misma subcapa que antecede a la imagen de TSA en orden de
65

decodificación. La definición de TSA puede imponer además restricciones sobre las imágenes en subcapas superiores que siguen a la imagen de TSA en orden de decodificación. No se permite ninguna de estas imágenes para referirse a una imagen que antecede a la imagen de TSA en orden de decodificación si esa imagen pertenece a la misma subcapa o una superior como la imagen de TSA. Las imágenes de TSA tienen TemporalId mayor que 0. El STSA es similar a la imagen de TSA pero no impone restricciones en las imágenes en subcapas superiores que siguen a la imagen de STSA en orden de decodificación y, por lo tanto, permiten conmutación ascendente sólo sobre la subcapa donde reside la imagen de STSA.

Una unidad no VCL NAL 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, un extremo de una unidad NAL de secuencia, un extremo de la unidad NAL de secuencia 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 no VCL NAL no son necesarias para la reconstrucción de valores de muestra decodificados.

Los parámetros que permanecen sin cambios a través de una secuencia de vídeo codificada 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 renderización y la reserva de recursos. Existen tres unidades NAL especificadas en H.264/AVC para transportar conjuntos de parámetros de secuencia: la unidad NAL de conjunto de parámetros de secuencia que contiene todos los datos para las unidades de H.264/AVC VCL NAL en la secuencia, la unidad de NAL de extensión del conjunto de parámetros de secuencia que contiene los datos para imágenes codificadas auxiliares, y el conjunto de parámetros de secuencia del subconjunto para las unidades MVC y SVC VCL NAL. En HEVC, un RBSP de conjunto de parámetros de secuencia incluye parámetros a los que pueden hacer referencia uno o más RBSP del conjunto de parámetros de imagen o una o más unidades SEI NAL 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 a los que pueden hacer referencia las unidades NAL del segmento codificado de una o más imágenes codificadas.

En HEVC, un conjunto de parámetros de vídeo (VPS) puede definirse como una estructura sintáctica que contiene elementos de sintaxis que se aplican a cero o más secuencias de vídeo codificadas completas según lo determinado por el contenido de un elemento sintáctico que se encuentra en el SPS al que se hace referencia por un elemento sintáctico que se encuentra en el PPS al que se hace referencia por un elemento sintáctico encontrado en cada cabecera de fragmento del segmento.

Un RBSP de conjunto de parámetros de vídeo puede incluir parámetros a los que pueden hacer referencia uno o más RBSP de conjunto de parámetros de secuencia.

La relación y jerarquía entre el conjunto de parámetros de vídeo (VPS), conjunto de parámetros de secuencia (SPS) y conjunto de parámetros de imagen (PPS) puede describirse como se indica a continuación. El 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 segmentos en todas las capas (escalabilidad o vista) en toda la secuencia de vídeo codificada. El SPS incluye los parámetros que son comunes para todos los segmentos en una capa particular (escalabilidad o vista) en toda la secuencia de vídeo codificada, y pueden compartirse por capas múltiples (escalabilidad o vista). El PPS incluye los parámetros que son comunes para todos los segmentos en una representación de capa particular (la representación de una capa de escalabilidad o visión en una unidad de acceso) y es probable que se compartan por todos los segmentos en múltiples representaciones de capa.

El VPS puede proporcionar información acerca de las relaciones de dependencia de las capas en una secuencia de bits, así como mucha otra información que es aplicable a todos los segmentos en todas las capas (escalabilidad o vista) en toda la secuencia de vídeo codificada. Se puede considerar que VPS comprende tres partes, la base VPS, la extensión VPS y la VUI VPS, donde la extensión VPS y la VUI VPS pueden estar opcionalmente presentes. En HEVC, se puede considerar que la base VPS comprende la estructura sintáctica de `video_parameter_set_rbsp()` sin la estructura sintáctica `vps_extension()`. La estructura sintáctica de `video_parameter_set_rbsp()` se especificó principalmente ya para la versión 1 de HEVC e incluye elementos de sintaxis que pueden ser de uso para la decodificación de capa base. En HEVC, la extensión VPS puede considerarse que comprende la estructura sintáctica `vps_extension()`. La estructura sintáctica `vps_extension()` se especificó en la versión 2 de HEVC principalmente para extensiones de múltiples capas y comprende elementos de sintaxis que pueden ser de uso para decodificar una o más capas no base, tales como elementos de sintaxis que indican relaciones de dependencia de la capa. El VPS comprende elementos de sintaxis que pueden ser útiles para la decodificación u otros propósitos, pero no son necesarios para ser utilizados en el proceso de decodificación de HEVC.

H.264/AVC y la sintaxis HEVC permite que muchas instancias de conjuntos de parámetros, y cada instancia se identifique con un identificador único. Para limitar el uso de memoria necesario para conjuntos de parámetros, el rango de valores para identificadores del conjunto de parámetros se ha limitado. En H.264/AVC y HEVC, cada encabezado de

segmento incluye el identificador del conjunto de parámetros de imagen que está activo para la decodificación de la imagen que contiene el segmento, y cada conjunto de parámetros de imagen contiene el identificador del conjunto de parámetros de secuencia activa. En consecuencia, la transmisión de conjuntos de imágenes y parámetros de secuencia no tiene que sincronizarse con precisión con la transmisión de segmentos. En cambio, es suficiente que la secuencia activa y los conjuntos de parámetros de imagen se reciban en cualquier momento antes de que sean referenciados, lo que permite la transmisión de conjuntos de parámetros “fuera de banda” usando un mecanismo de transmisión más fiable en comparación con los protocolos utilizados para los datos de segmento. Por ejemplo, los conjuntos de parámetros pueden incluirse como un parámetro en la descripción de la sesión para sesiones del Protocolo de Transporte en Tiempo Real (RTP). Si los conjuntos de parámetros se transmiten en banda, se pueden repetir para mejorar la robustez de errores.

La transmisión, señalización o almacenamiento fuera de banda puede usarse adicionalmente o alternativamente para otros propósitos que la tolerancia frente a errores de transmisión, tales como facilidad de acceso o negociación de la sesión. Por ejemplo, una entrada de muestra de una pista en un archivo conforme al Formato de Archivo de Medios de Base ISO puede comprender conjuntos de parámetros, mientras que los datos codificados en la secuencia de bits se almacenan en otra parte del archivo o en otro archivo. La frase a lo largo de la secuencia de bits (por ejemplo, que indica a lo largo de la secuencia de bits) puede usarse en las reivindicaciones y las realizaciones descritas para referirse a la transmisión, señalización o almacenamiento fuera de banda de una manera que los datos fuera de banda están asociados con la secuencia de bits. La expresión decodificación a lo largo de la secuencia de bits o similar puede referirse a la decodificación de los datos de fuera de banda referidos (que pueden obtenerse de la transmisión, señalización o almacenamiento fuera de banda) que está asociada con la secuencia de bits.

Un conjunto de parámetros puede activarse mediante una referencia de un segmento o de otro conjunto de parámetros activos o en algunos casos de otra estructura sintáctica tal como un mensaje SEI de período de almacenamiento en memoria intermedia.

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 renderización, 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. En consecuencia, se requieren codificadores para seguir la norma H.264/AVC o la norma HEVC cuando crean mensajes SEI, y decodificadores conformes a la norma H.264/AVC o la norma 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, la interoperación. 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 adicionalmente se puede especificar el proceso para manejar mensajes SEI particulares en el receptor.

Varios mensajes SEI anidados se han especificado en las normas AVC y HEVC o se proponen de otro modo. La idea de los mensajes SEI anidados es contener uno o más mensajes SEI dentro de un mensaje SEI de anidamiento y proporcionar un mecanismo para asociar los mensajes SEI contenidos con un subconjunto de la secuencia de bits y/o un subconjunto de datos decodificados. Puede ser necesario que un mensaje SEI anidado contenga uno o más mensajes SEI que no sean los propios mensajes SEI. Se puede hacer referencia a un mensaje SEI contenido en un mensaje SEI anidado como un mensaje SEI anidado. Se puede hacer referencia a un mensaje SEI no contenido en un mensaje SEI anidado como un mensaje SEI no anidado. El mensaje SEI anidado escalable de HEVC permite identificar un subconjunto de secuencia de bits (resultante de un proceso de extracción de subsecuencia de bits) o un conjunto de capas a las que se aplican los mensajes SEI anidados. Un subconjunto de secuencia de bits también puede denominarse como una subsecuencia de bits.

Una imagen codificada es una representación codificada de una imagen. Una imagen codificada en H.264/AVC comprende las unidades NAL VCL que se requieren para la decodificación de la imagen. En H.264/AVC, una imagen codificada puede ser una imagen codificada primaria o una imagen codificada redundante. Una imagen codificada primaria se usa en el proceso de decodificación de flujos de bits válidos, mientras que una imagen codificada redundante es una representación redundante que solo debería decodificarse cuando la imagen codificada primaria no puede decodificarse con éxito. En HEVC, no se ha especificado una imagen codificada redundante.

En H.264/AVC, una unidad de acceso (AU) comprende una imagen codificada primaria y esas unidades NAL que están asociadas con ella. En H.264/AVC, el orden de aparición de las unidades NAL dentro de una unidad de acceso se restringe como se indica a continuación. Una unidad NAL de delimitador de unidad de acceso opcional puede indicar el inicio de una unidad de acceso. Es seguido por cero o más unidades NAL SEI. Los segmentos codificados de la imagen codificada primaria aparecen a continuación. En H.264/AVC, el segmento codificado de la imagen codificada primaria puede ser seguido por segmentos codificados para cero o más imágenes codificadas

redundantes. Una imagen codificada redundante es una representación codificada de una imagen o una parte de una imagen. Una imagen codificada redundante puede decodificarse si la imagen codificada primaria no es recibida por el decodificador, por ejemplo, debido a una pérdida de transmisión o una corrupción en el medio de almacenamiento físico.

En H.264/AVC, una unidad de acceso también puede incluir una imagen codificada auxiliar, que es una imagen que complementa la imagen codificada primaria y puede usarse, por ejemplo, en el proceso de visualización. Una imagen codificada auxiliar puede usarse, por ejemplo, como un canal alfa o un plano alfa que especifica el nivel de transparencia de las muestras en las imágenes decodificadas. Un canal o plano alfa puede usarse en una composición en capas o sistema de representación, donde la imagen de salida se forma superponiendo imágenes que son al menos parcialmente transparentes una encima de la otra. Una imagen codificada auxiliar tiene las mismas restricciones sintácticas y semánticas que una imagen codificada redundante monocromática. En H.264/AVC, una imagen codificada auxiliar contiene el mismo número de macrobloques que la imagen codificada primaria.

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) puede definirse como un conjunto de unidades NAL que están asociadas entre sí según una regla de clasificación especificada, son consecutivas en 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.

Una secuencia de bits puede definirse como una secuencia de bits, en forma de una corriente de unidad NAL o una secuencia de bytes, que forma la representación de imágenes codificadas y datos asociados que forman una o más secuencias de vídeo codificadas. Una primera secuencia de bits puede ser seguido por una segunda secuencia 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 extremo de la primera secuencia de bits puede indicarse mediante una unidad NAL específica, que puede denominarse unidad NAL de extremo de secuencia de bits (EOB) y que es la última unidad NAL de la secuencia de bits. En HEVC y sus extensiones preliminares actuales, se requiere que la unidad NAL EOB tenga `nuh_layer_id` igual a 0.

En H.264/AVC, una secuencia de vídeo codificada se define como una secuencia de acceso consecutivo

unidades en orden de decodificación desde una unidad de acceso IDR, incluyendo, a la siguiente unidad de acceso IDR, exclusiva, o al final de la secuencia de bits, lo que aparece anteriormente.

En HEVC, una secuencia de vídeo codificada (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 `NoRaslOutput` igual a 1, seguida de cero o más unidades de acceso que no son unidades de acceso de IRAP con `NoRaslOutputFlag` igual a 1, incluyendo todas las unidades de acceso posteriores hasta pero sin incluir ninguna unidad de acceso posterior que sea una unidad de acceso de IRAP con `NoRaslOutputFlag` igual a 1. Una unidad de acceso IRAP puede definirse como una unidad de acceso en la que la imagen de la capa base es una imagen IRAP. El valor de `NoRaslOutputFlag` es igual a 1 para cada imagen IDR, cada imagen BLA y cada imagen IRAP que es la primera imagen en esa capa particular en la secuencia de bits en orden de decodificación, es la primera imagen IRAP que sigue un extremo de la unidad NAF de secuencia que tiene el mismo valor de `nuh_layer_id` en orden de decodificación. En la HEVC de múltiples capas, el valor de `NoRaslOutputFlag` es igual a 1 para cada imagen IRAP cuando su `nuh_layer_id` es tal que `FayerinicializedFlag[nuh_layer_id]` es igual a 0 y `FayerInitalizedFlag[refLayerId]` es igual a 1 para todos los valores de `refLayerId` equal to `IdDirectRefLayer[nuh_layer_id][j]`, donde `j` está en el rango de 0 a `NumDirectRefLayers[nuh_layer_id] - 1`, inclusive. De otro modo, el valor de `NoRaslOutputFlag` es igual a `HandleCraAsBlaFlag`. `NoRaslOutputFlag` igual a 1 tiene un impacto que las imágenes RASL asociadas con la imagen IRAP para la que se establece el indicador `NoRaslOutput` no son emitidas por el decodificador. Puede haber medios para proporcionar el valor de `HandleCraAsBlaFlag` al decodificador desde una entidad externa, tal como un jugador o un receptor, que puede controlar el decodificador. `HandleCraAsBlaFlag` puede establecerse en 1 por ejemplo por un jugador que busca una nueva posición en una secuencia de bits o sintoniza en una difusión y comienza a decodificar y luego comienza a decodificar desde una imagen CRA. Cuando `HandleCraAsBlaFlag` es igual a 1 para una imagen CRA, la imagen CRA se gestiona y decodifica como si fuera una imagen BLA.

En HEVC, una secuencia de vídeo codificada puede especificarse adicional o alternativamente (a la especificación anterior) para finalizar, cuando una unidad de NAL específica, a la cual puede hacerse referencia como una unidad NAL de extremo de secuencia (EOS), aparece en la secuencia de bits y tiene `nuh_layer_id` igual a 0.

En HEVC, un grupo de secuencia de vídeo codificado (CVSG) puede definirse, por ejemplo, como una o más CVS consecutivas en orden de decodificación que consisten colectivamente en una unidad de acceso de IRAP que activa un RBSP VPS `firstVpsRbsp` que no estaba ya activo seguido de todas las unidades de acceso posteriores, en orden de

decodificación, para las que firstVpsRbsp es el RBSP VPS activo hasta el final de la secuencia de bits o hasta que excluye la unidad de acceso que activa un RBSP VPS diferente a firstVpsRbsp, lo que sea primero en orden de decodificación.

Un grupo de imágenes (GOP) y sus características pueden definirse como se indica a continuación. Un GOP puede decodificarse independientemente de si se decodificaron cualesquiera imágenes anteriores. Un GOP abierto es un grupo de imágenes en las que las imágenes que anteceden 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 interpredicción) a imágenes que pertenecen a un GOP anterior. Un decodificador H.264/AVC puede reconocer una imagen intra que inicia un GOP abierto desde el mensaje SEI del punto de recuperación en una secuencia de bits H.264/AVC. Un decodificador HEVC puede reconocer una imagen intra que inicia un GOP abierto, porque un tipo de unidad de NAL específico, un tipo de unidad de NAL de CRA, puede usarse para sus segmentos 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 abierta 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 Estructura de Imágenes (SOP) puede definirse como una o más imágenes codificadas consecutivas en orden de decodificación, en la que la primera imagen codificada en orden de decodificación es una imagen de referencia en la subcapa temporal más baja y sin imagen codificada, excepto potencialmente la primera imagen codificada en orden de decodificación es una imagen RAP. Todas las imágenes en la SOP anterior anteceden en el orden de decodificación todas las imágenes en la actual SOP y todas las imágenes en la siguiente SOP tienen éxito en el orden de decodificación de todas las imágenes en la actual SOP. Una SOP puede representar una estructura de interpredicción jerárquica y repetitiva. El término grupo de imágenes (GOP) a veces se puede usar indistintamente con el término SOP y tiene la misma semántica que la semántica de la SOP.

La sintaxis de secuencia de bits de H.264/AVC y HEVC indica si una imagen particular es una imagen de referencia para la interpredicción de cualquier otra imagen. Las imágenes de cualquier tipo de codificación (I, P, B) pueden ser imágenes de referencia o imágenes no referenciales en H.264/AVC y HEVC.

H.264/AVC especifica el proceso para la marca de imagen de referencia decodificada para controlar el consumo de memoria en el decodificador. El número máximo de imágenes de referencia utilizado para la interpredicción, denominado como M, se determina en el conjunto de parámetros de secuencia. Cuando se decodifica una imagen de referencia, se marca como “usada para la referencia”. Si la decodificación de la imagen de referencia causó más de M imágenes marcadas como “usadas para la referencia”, al menos una imagen se marca como “no usada para referencia”. Hay dos tipos de operación para el marcado de imagen de referencia decodificado: control de memoria adaptativa y ventana deslizante. El modo de operación para el marcado de imagen de referencia decodificada se selecciona en base de imágenes. El control de memoria adaptativa permite la señalización explícita de que las imágenes se marcan como “no usadas para referencia” y también pueden asignar índices a largo plazo a imágenes de referencia a corto plazo. El control de memoria adaptativa puede requerir la presencia de parámetros de operación de control de gestión de memoria (MMCO) en la secuencia de bits. Los parámetros MMCO pueden incluirse en una estructura sintáctica de marcado de imágenes de referencia decodificadas. Si el modo de operación de la ventana deslizante está en uso y hay M imágenes marcadas como “usadas para la referencia”, la imagen de referencia a corto plazo que era la primera imagen decodificada entre esas imágenes de referencia a corto plazo que se marcan como “usadas para la referencia” se marca como “no usada para referencia”. En otras palabras, el modo de operación de ventana deslizante da como resultado una operación de almacenamiento en memoria intermedia de primera entrada primera salida entre imágenes de referencia a corto plazo.

Una de las operaciones de control de gestión de memoria en H.264/AVC hace que todas las imágenes de referencia excepto la imagen actual se marquen como “no usada para referencia”. Una imagen de renovación de decodificación instantánea (IDR) contiene solo segmentos intracodificados y provoca un “reinicio” similar de imágenes de referencia.

En HEVC, no se usan estructuras de sintaxis de marcado de imágenes de referencia y procesos de decodificación relacionados, pero en cambio se usan una estructura sintáctica del conjunto de imágenes de referencia (RPS) y el proceso de decodificación para un propósito similar. Un conjunto de imágenes de referencia válido o activo para una imagen incluye todas las imágenes de referencia usadas como referencia para la imagen y todas las imágenes de referencia que se mantienen marcadas como “usadas para la referencia” para cualquier imagen posterior en orden de decodificación. Hay seis subconjuntos del conjunto de imágenes de referencia, que se denominan como, por ejemplo, RefPicSetStCurr0 (a.k.a. RefPicSetStCurrBefore), RefPicSetStCurr1 (a.k.a. RefPicSetStCurrAfter), RefPicSetStFoll0, RefPicSetStFoll1, RefPicSetFtCurr, and RefPicSetLtFol1. RefPicSetStFoll0 y RefPicSetStFoll1 también se puede considerar que forma conjuntamente un subconjunto RefPicSetStFol1. La notación de los seis subconjuntos es la siguiente. “Curr” se refiere a imágenes de referencia que se incluyen en las listas de imágenes de referencia de la imagen actual y, por lo tanto, pueden usarse como referencia de interpredicción para la imagen actual. “Fol” se refiere a imágenes de referencia que no están

incluidas en las listas de imágenes de referencia de la imagen actual, pero pueden usarse en imágenes posteriores en orden de decodificación como imágenes de referencia. “St” se refiere a imágenes de referencia a corto plazo, que generalmente pueden identificarse a través de un cierto número de bits menos significativos de su valor POC. “Lt” se refiere a imágenes de referencia a largo plazo, que se identifican específicamente y generalmente tienen una mayor diferencia de valores de POC en relación con la imagen actual que lo que puede representarse por el determinado número de bits menos significativos mencionado. “0” se refiere a aquellas imágenes de referencia que tienen un valor POC más pequeño que el de la imagen actual. “1” se refiere a aquellas imágenes de referencia que tienen un valor POC mayor que el de la imagen actual. RefPicSetStCurr0, RefPicSetStCurr1, RefPicSetStFoll0 y RefPicSetStFoll1, se refieren colectivamente como el subconjunto a corto plazo del conjunto de imágenes de referencia. Se hace referencia a RefPicSetLtCurr y RefPicSetFtFoll colectivamente como el subconjunto a largo plazo del conjunto de imágenes de referencia.

En HEVC, un conjunto de imágenes de referencia puede especificarse en un conjunto de parámetros de secuencia y tomarse en uso en la cabecera del segmento a través de un índice al conjunto de imágenes de referencia. También se puede especificar un conjunto de imágenes de referencia en una cabecera del segmento. Un conjunto de imágenes de referencia puede codificarse independientemente o puede predecirse a partir de otro conjunto de imágenes de referencia (conocido como predicción inter-RPS). En ambos tipos de codificación de conjunto de imágenes de referencia, se envía adicionalmente una marca (used_by_curr_pic_X_flag) para cada imagen de referencia que indica si la imagen de referencia se usa para referencia mediante la imagen actual (incluida en una lista de *Curr) o no (incluida en una lista de *Foll). Las imágenes que se incluyen en el conjunto de imágenes de referencia usado por el segmento actual se marcan como “usadas para la referencia”, y las imágenes que no están en el conjunto de imágenes de referencia usado por el segmento actual se marcan como “no usadas para referencia”. Si la imagen actual es una imagen IDR, RefPicSetStCurr0, RefPicSetStCurr1, RefPicSetStFoll0, RefPicSetStFoll1, RefPicSetFtCurr, y RefPicSetFtFoll se establecen en vacío.

Una Memoria intermedia de imágenes decodificadas (DPB) puede usarse en el codificador y/o en el decodificador. Hay dos razones para almacenar temporalmente 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 para el marcado de imágenes de referencia y reordenación de salida, separar las memorias intermedias para el almacenamiento en memoria intermedia de la imagen de referencia y el almacenamiento en memoria intermedia de la 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 unificadas 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, dos listas de imágenes de referencia (lista 0 de imágenes de referencia y lista 1 de imágenes de referencia) se generan para cada segmento bipredictivo (B), y una lista de imágenes de referencia (lista 0 de imágenes de referencia) se forma para cada segmento intercodificado (P).

Una lista de imágenes de referencia, tal como la lista de imágenes de referencia 0 y la lista de imágenes de referencia 1, se construye típicamente en dos pasos: Primero, se genera una lista inicial de imágenes de referencia. La lista de imágenes de referencia inicial puede generarse, por ejemplo, basándose en frame_num, POC, temporal_id (o TemporalId o similar), o información sobre la jerarquía de predicción tal como la estructura de GOP, o cualquier combinación de los mismos. En segundo lugar, la lista de imágenes de referencia inicial puede reordenarse mediante comandos de reordenación de lista de imágenes de referencia (RPLR), también conocidos como estructura sintáctica de modificación de la lista de imágenes de referencia, que puede estar contenida en cabeceras del segmento. En H.264/AVC, los comandos RPLR indican las imágenes que se ordenan al comienzo de la lista de imágenes de referencia respectiva. Se pueden hacer referencia a esta segundo paso también como el proceso de modificación de la lista de imágenes de referencia, y los comandos RPLR pueden incluirse en una estructura sintáctica de modificación de la lista de imágenes de referencia. Si se usan conjuntos de imágenes de referencia, la lista 0 de imágenes de referencia puede inicializarse para contener RefPicSetStCurr0 primero, seguido de RefPicSetStCurr1, seguido de RefPicSetLtCurr. La lista de imágenes de referencia 1 puede inicializarse para contener RefPicSetStCurr1 primero, seguido de RefPicSetStCurr0. En HEVC, las listas de imágenes de referencia iniciales pueden modificarse a través de la estructura sintáctica de modificación de la lista de imágenes de referencia, donde las imágenes en las listas de imágenes de referencia iniciales pueden identificarse a través de un índice de entrada a la lista. En otras palabras, en HEVC, la modificación de la lista de imágenes de referencia se codifica en una estructura sintáctica que comprende un bucle sobre cada entrada en la lista de imágenes de referencia final, donde cada entrada de bucle es un índice codificado de longitud fija a la lista de imágenes de referencia inicial e indica la imagen en orden de posición ascendente en la lista de imágenes de referencia final.

Muchas normas de codificación, incluyendo H.264/AVC y HEVC, pueden tener un proceso de decodificación para obtener 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 la secuencia de bits es algunos modos de intercodificación o puede derivarse (por un codificador y un decodificador), por ejemplo, usando bloques circundantes en algunos otros modos de intercodificación.

Para representar vectores de movimiento eficazmente en flujos de bits, los vectores de movimiento pueden codificarse diferencialmente con respecto a un vector de movimiento predicho específico de bloque. En muchos códecs de vídeo, 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 del vector de movimiento, a veces denominadas predicción de vector de movimiento avanzado (AMVP), es generar una lista de predicciones candidatas de bloques adyacentes y/o bloques cúbicados en imágenes de referencia temporal y señalizando el candidato elegido como el predictor del vector de movimiento. Además de predecir los valores del vector de movimiento, se puede predecir el índice de referencia de la imagen previamente codificada/decodificada. El índice de referencia se predice típicamente desde bloques adyacentes y/o bloques cúbicados en la imagen de referencia temporal. La codificación diferencial de vectores de movimiento está típicamente deshabilitada a través de los límites del segmento.

Muchos códecs de vídeo de alta eficiencia, tales como el códec de HEVC, emplean un mecanismo de codificación/decodificación de información de movimiento adicional, a menudo llamado modo/ proceso/ mecanismo fusionado/ de unión, donde toda la información de movimiento de un bloque/PU se predice y usa sin ninguna modificación/corrección.

La codificación de vídeo escalable puede referirse a la estructura de codificación donde una secuencia de bits puede contener múltiples representaciones del contenido, por ejemplo, a diferentes velocidades de bits, de resoluciones o de marcos. En estos casos, el receptor puede extraer la representación deseada dependiendo de sus características (por ejemplo, resolución que coincida mejor con el dispositivo de visualización). Alternativamente, un servidor o un elemento de la red puede extraer las porciones de la secuencia de bits que deben transmitirse al receptor dependiendo de, por ejemplo, las características de la red o de las capacidades de procesamiento del receptor. Una secuencia 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 potencian la calidad de vídeo cuando se reciben y decodifican junto con las capas inferiores. Para mejorar la eficacia de la codificación de 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. De manera similar, los datos de píxeles de las capas inferiores pueden usarse para crear predicción en la capa de mejora.

En algunos esquemas de codificación de vídeo escalables, 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 velocidad de marco), 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 una secuencia de bits escalable correspondiente a una representación de capa escalable puede extraerse y decodificarse para producir una representación de la señal original en cierta fidelidad.

Los modos de escalable o las dimensiones de escalabilidad pueden incluir, pero no se limitan a, los siguientes:

- Escala de calidad: Las imágenes de capa base se codifican a una calidad más baja que las imágenes de la capa de mejora, que se pueden lograr, por ejemplo, utilizando un valor de parámetro de cuantificación mayor (es decir, un tamaño de paso de cuantificación mayor para la cuantificación del coeficiente de transformación) en la capa base que en la capa de mejora. La escalabilidad de calidad se puede categorizar adicionalmente en una escalabilidad de grano fino o de granularidad fina (FGS), escalabilidad media de grano o de media granularidad (MGS), y/o escalabilidad de grano grueso o de escala 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 (por ejemplo, 8 bits) que las imágenes de la capa de mejora (por ejemplo, 10 o 12 bits).

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

– Escalabilidad de gama de colores: 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 UHDTV (BT.2020) la gama de colores y la capa base pueden tener la gama de colores ITU-R BT.709.

5 – Ver escalabilidad, a la que también puede referirse como codificación de múltiples visualizaciones. La capa base representa una primera vista, mientras que una capa de mejora representa una segunda vista.

10 – La escalabilidad de profundidad, a la que también puede referirse como codificación de profundidad mejorada. Una capa o algunas capas de una secuencia de bits pueden representar la(s) vista(s) de textura, mientras que otra capa o capas pueden representar vistas en profundidad.

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

15 – Escalabilidad entrelazada a progresiva (también conocida como escalabilidad de campo a marco): el material de contenido de origen entrelazado codificado de la capa base se mejora con una capa de mejora para representar el contenido de fuente progresiva. El contenido de origen entrelazado codificado en la capa base puede comprender campos codificados, marcos codificados que representan pares de campo, o una mezcla de ellos. En la escalabilidad entrelazada a progresiva, 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.

20 – Escalabilidad de códec híbrido (también conocida como escalabilidad estándar de codificación): En la escalabilidad de códec híbrido, la sintaxis de secuencia de bits, la semántica y el proceso de decodificación de la capa base y la capa de mejora se especifican en diferentes normas de codificación de vídeo. Por lo tanto, las imágenes de la capa base se codifican según una norma o formato de codificación diferente que las imágenes de la capa de mejora.

25 Por ejemplo, la capa base puede codificarse con H.264/AVC y una capa de mejora puede codificarse con una extensión de múltiples capas de HEVC. De forma más general, en la escalabilidad de códec híbrido una o más capas puede codificarse según una norma o especificación de codificación y otras una o más capas pueden codificarse según otra norma de codificación o especificación. Por ejemplo, puede haber dos capas codificadas según la extensión de la MVC de H.264/AVC (de la que una es una capa base codificada según H.264/AVC), y una o más capas adicionales codificadas según MV-HEVC. Además, el número de norma de codificación o especificaciones según la cual se codifican diferentes capas del mismo secuencia de bits puede no limitarse a dos en la escalabilidad de códec híbrido.

35 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.

40 El SHVC permite el uso de predicción ponderada o un proceso de mapeo de color basado en una tabla de consulta 3D (LUT) para la escalabilidad de gama de colores. El enfoque LUT 3D puede describirse como sigue. El rango de valores de muestra de cada componente de color puede dividirse primero en dos rangos, formando hasta 2x2x2 octantes, y luego los intervalos de luma pueden dividirse adicionalmente hasta cuatro partes, dando como resultado hasta 8x2x2 octantes. Dentro de cada octante, se aplica un modelo lineal de componente de color transversal para realizar el mapeo de color. Para cada octante, cuatro vértices se codifican y/o decodifican a partir de la secuencia de bits para representar un modelo lineal dentro del octante. La tabla de mapeo de color se codifica y/o decodifica a partir de la secuencia de bits por separado para cada componente de color. Se puede considerar que el mapeo de color implica tres pasos: Primero, se determina el octante al que pertenece un triplete de muestra de capa de referencia dado (Y, Cb, Cr). En segundo lugar, las ubicaciones de muestras de luma y croma pueden alinearse mediante la aplicación de un proceso de ajuste del componente de color. En tercer lugar, se aplica el mapeo lineal especificado para el octante determinado. El mapeo puede tener naturaleza de componentes cruzados, es decir, un valor de entrada de un componente de color puede afectar el valor mapeado de otro componente de color. Adicionalmente, si también se requiere remuestreo entre capas, la entrada al proceso de remuestreo es la imagen que se ha mapeado en color. El mapeo de color puede (pero no necesita) mapear muestras de una primera profundidad de bit a muestras de otra profundidad de bit.

55 La correspondencia espacial de una imagen de capa de referencia y una imagen de capa de mejora puede inferirse o se puede indicar con uno o más tipos de los denominados desplazamientos de ubicación de capa de referencia. En HEVC, los desplazamientos de ubicación de la capa de referencia pueden incluirse en el PPS por el codificador y decodificarse a partir del PPS por el decodificador. Los desplazamientos de ubicación de la capa de referencia pueden usarse, pero no se limitan a, lograr la escalabilidad de ROI. Los desplazamientos de ubicación de la capa de referencia pueden comprender uno o más desplazamientos de capa de referencia escalados, desplazamientos de región de referencia y conjuntos de fases de remuestreo. Los desplazamientos de capa de referencia escalados pueden considerarse que especifican los desplazamientos horizontales y verticales entre la muestra en la imagen actual que está ubicada con la muestra de luma superior izquierda de la región de referencia en una imagen decodificada en una capa de referencia y las compensaciones horizontal y vertical entre la muestra en la imagen actual que está ubicada con la muestra de luma inferior derecha de la región de referencia en una imagen decodificada en una capa de referencia. Otra forma es considerar desplazamientos de capa de referencia escalados para especificar las posiciones de las muestras de esquina de la región de referencia muestreada en un nivel superior en relación con las muestras de

esquina respectivas de la imagen de capa de mejora. Los valores de desplazamiento de la capa de referencia escalados pueden ser firmados. Pueden considerarse desplazamientos de la región de referencia para especificar las compensaciones horizontales y verticales entre la muestra de luma superior izquierda de la región de referencia en la imagen decodificada en una capa de referencia y la muestra de luma superior izquierda de la misma imagen decodificada, así como las compensaciones horizontales y verticales entre la muestra de luma inferior derecha de la región de referencia en la imagen decodificada en una capa de referencia y la muestra de luma inferior derecha de la misma imagen decodificada. Los valores de desplazamiento de la región de referencia pueden ser firmados. Se puede considerar que un conjunto de fases de remuestreo especifica los desplazamientos de fase utilizados en el proceso de remuestreo de una imagen de capa de referencia directa. Pueden proporcionarse diferentes desplazamientos de fase para componentes de luma y croma.

La escalabilidad de códec híbrido puede usarse junto con cualquier tipo de escalabilidad, tal como temporal, calidad, espacial, vista múltiple, profundidad mejorada, imagen auxiliar, profundidad de bits, gama de colores, formato de croma y/o escalabilidad de ROI. Dado que se puede usar la escalabilidad de códec híbrido junto con otros tipos de escalabilidades, se puede considerar que forma una categorización diferente de tipos de escalabilidad.

El uso de escalabilidad de códec híbrido puede indicarse, por ejemplo, en una secuencia de bits de capa de mejora. Por ejemplo, en la HEVC multicapa, el uso de escalabilidad de códec híbrido puede indicarse en el VPS, por ejemplo, usando el elemento de sintaxis `vps_base_layer_internal_flag`.

El término capa puede usarse en contexto de cualquier tipo de escalabilidad, incluyendo la escalabilidad de vista y mejoras en profundidad. Una capa de mejora puede referirse a cualquier tipo de mejora, tal como SNR, espacial, vista múltiple, profundidad, profundidad de bit, formato de croma y/o mejora de la 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 SNR/espacial, o una vista de base de textura para la codificación de vídeo de profundidad mejorada.

Algunos esquemas de codificación de vídeo escalables pueden requerir que las imágenes IRAP se alineen a través de capas de una manera que todas las imágenes en una unidad de acceso sean imágenes IRAP o ninguna imagen en una unidad de acceso sea una imagen IRAP. Otros esquemas de codificación de vídeo escalables, tales como las extensiones de múltiples capas de HEVC, pueden permitir que las 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 imágenes de una unidad de acceso no sean imágenes IRAP. Los flujos de bits escalables con imágenes IRAP o similares que no están alineados a través de capas pueden usarse, por ejemplo, para proporcionar imágenes IRAP más frecuentes en la capa base, donde pueden tener un tamaño codificado más pequeño debido a, por ejemplo, una resolución espacial más pequeña. Un proceso o mecanismo para el arranque por capas de la decodificación puede incluirse en un esquema de decodificación de vídeo. Por lo tanto, los decodificadores pueden iniciar la decodificación de una secuencia de bits cuando una capa base contiene una imagen IRAP y empieza a decodificar por pasos otras capas cuando contienen imágenes IRAP. En otras palabras, en un arranque 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, las vistas, los componentes adicionales tales como la profundidad, o una combinación) dado que imágenes posteriores de las capas de mejora adicionales se decodifican en el proceso de decodificación. El aumento progresivo del número de capas decodificadas puede ser percibido por ejemplo como una mejora progresiva de la calidad de la imagen (en el caso de escalabilidad de calidad y espacial).

Un mecanismo de arranque por capa puede generar imágenes no disponibles para las imágenes de referencia de la primera imagen en orden de decodificación en una capa de mejora particular. Alternativamente, un decodificador puede omitir la decodificación de imágenes anteriores, en orden de decodificación, la imagen IRAP a partir de la cual puede iniciarse la decodificación de una capa. Estas imágenes que pueden omitirse pueden ser marcadas específicamente por el codificador u otra entidad dentro de la secuencia de bits. Por ejemplo, pueden usarse uno o más tipos de unidad NAL específicos para ellos. A estas imágenes, independientemente de si se marcan específicamente con un tipo de unidad NAL o se infieren, por ejemplo, por el decodificador, se les puede hacer referencia como imágenes de omisión de acceso aleatorio (CL-RAS) de capa transversal. El decodificador puede omitir la salida de las imágenes no disponibles generadas y las imágenes CL-RAS decodificadas.

Un mecanismo de arranque por capas puede comenzar la salida de imágenes de capa de mejora de una imagen IRAP en esa capa de mejora, cuando todas las capas de referencia de esa capa de mejora se han inicializado de manera similar con una imagen IRAP en las capas de referencia. En otras palabras, cualquier imagen (dentro de la misma capa) que antecede dicha imagen IRAP en orden de salida podría no emitirse del decodificador y/o no mostrarse. En algunos casos, las imágenes principales decodificables asociadas con dicha imagen IRAP pueden ser emitidas por el decodificador, mientras que otras imágenes anteriores a tal imagen IRAP, tales como las imágenes RASL asociadas con la imagen IRAP, podrían no ser emitidas por el decodificador y/o podrían no visualizarse.

La concatenación de datos de vídeo codificados, que también puede denominarse empalme, puede ocurrir, por ejemplo, se concatenan secuencias de vídeo codificadas en una secuencia de bits que se difunde o se transmite o

almacena en una memoria de masa. Por ejemplo, las secuencias de vídeo codificadas que representan comerciales o anuncios pueden concatenarse con películas u otro contenido “primario”.

Los flujos de bits de vídeo escalable podrían contener imágenes IRAP que no están alineadas a través de las capas. Sin embargo, puede ser conveniente permitir la concatenación de una secuencia de vídeo codificada que contenga una imagen IRAP en la capa base en su primera unidad de acceso pero no necesariamente en todas las capas. Una segunda secuencia de vídeo codificada que se empalma después de una primera secuencia de vídeo codificada debería desencadenar un proceso de arranque de decodificación por capa. Esto es porque la primera unidad de acceso de dicha segunda secuencia de vídeo codificada podría no contener una imagen IRAP en todas sus capas y, por lo tanto, algunas imágenes de referencia para las imágenes no IRAP en esa unidad de acceso pueden no estar disponibles (en la secuencia de bits concatenado) y, por lo tanto, no pueden decodificarse. La entidad que concatena las secuencias de vídeo codificadas, a la que se hace referencia a continuación como el empalmador, debería, por lo tanto, modificar la primera unidad de acceso de la segunda secuencia de vídeo codificada de modo que desencadene un proceso de arranque por capa en el (los) decodificador(es).

Pueden existir indicaciones en la sintaxis de secuencia de bits para indicar el desencadenamiento de un proceso de arranque por capa. Estas indicaciones pueden generarse por codificadores o empalmadores y pueden ser obedecidas por los decodificadores. Esta(s) indicación(es) pueden usarse para tipos de imágenes particulares o tipos de unidad NAL solamente, tales como solo para imágenes IDR, o pueden usarse para cualquier tipo de imagen. Sin pérdida de generalidad, se hace referencia a continuación a una indicación llamada `cross_layer_bla_flag` que se considera incluida en una cabecera de segmento del segmento. Debe entenderse que una indicación similar con cualquier otro nombre o incluida en cualquier otra estructura sintáctica podría usarse adicional o alternativamente.

Independientemente de la(s) indicación(es) que desencadenan un proceso de arranque por capa, cierto(s) tipo(s) de unidades NAL y/o tipo(s) de imagen pueden desencadenar un proceso de arranque por capa. Por ejemplo, una imagen BLA de capa base puede desencadenar un proceso de arranque por capa.

Un mecanismo de arranque por capas puede iniciarse en uno o más de los siguientes casos:

Al comienzo de una secuencia de bits.

Al comienzo de una secuencia de vídeo codificada, cuando se controla específicamente, por ejemplo cuando se inicia o reinicia un proceso de decodificación, por ejemplo como respuesta a la sintonización en una difusión o búsqueda a una posición en un archivo o secuencia. El proceso de decodificación puede ingresar una variable, por ejemplo, a la cual se hace referencia como `NoCllasOutputFlag`, que puede controlarse por medios externos, como el reproductor de vídeo o similares.

Una imagen BLA de capa base.

Una imagen IDR de capa base con `cross_layer_bla_flag` igual a 1. (O una imagen IRAP de capa base con `cross_layer_bla_flag` igual a 1.)

Una imagen CRA de capa base con `HandleCraAsBlaFlag` igual a 1, donde el proceso de decodificación aporta la variable `HandleCraAsBlaFlag` que puede estar controlada por medios externos.

Cuando se inicia un mecanismo de arranque por capas, el proceso de decodificación puede establecer variables en consecuencia. Por ejemplo, la variable `NoCllasOutputFlag` puede establecerse igual a 1.

Un proceso de decodificación puede especificarse de manera que una cierta variable controle si se usa o no un proceso de arranque por capas. Por ejemplo, se puede usar una variable `NoCllasOutputFlag`, que, cuando es igual a 0, indica una operación de decodificación normal, y cuando es igual a 1, indica una operación de arranque por capas, o, cuando la unidad de acceso donde `NoCllasOutputFlag` se establece igual a 1 contiene imágenes IRAP en todas las capas, indica un inicio de decodificación donde todas las capas se decodifican convencionalmente.

Cuando se inicia un mecanismo de arranque por capas, todas las imágenes de la DPB pueden marcarse como “no usadas para referencia”. En otras palabras, todas las imágenes en todas las capas pueden marcarse como “no usadas para referencia” y no se utilizarán como una referencia para la predicción para la imagen que inicia el mecanismo de arranque por capas o cualquier imagen posterior en orden de decodificación.

Un proceso de decodificación para arranque por capas puede ser controlado por ejemplo por dos variables de matriz `LayerInitializedFlag[ i ]` y `FirstPicInLayerDecodedFlag[ i ]` que puede tener entradas para cada capa (posiblemente excluyendo la capa base y posiblemente otras capas independientes también). Cuando se invoca el proceso de arranque por capas, por ejemplo, como respuesta a `NoCllasOutputFlag` que es igual a 1, estas variables de matriz pueden reiniciarse a sus valores por defecto. Por ejemplo, cuando se habilitan 63 capas (por

ejemplo, con 6-bit `nuh_layer_id`), las variables pueden reiniciarse de la siguiente manera: la variable `LayerInitializedFlag[ i ]` se establece igual a 0 para todos los valores de `i` de 0 a 62, inclusive, y la variable `FirstPicInLayerDecodedFlag[ i ]` se establece igual a 0 para todos los valores de `i` de 1 a 62, inclusive.

5 El proceso de decodificación puede incluir lo siguiente o similar para controlar la salida de imágenes RASL. Cuando la imagen actual es una imagen IRAP, se aplica lo siguiente:

- Si `LayerInitializedFlag[ nuh_layer_id ]` es igual a 0, la variable `NoRasOutputFlag` se establece igual a 1.

10 - De cualquier otra manera, si hay algunos medios externos disponibles para establecer la variable `HandleCraAsBlaFlag` en un valor para la imagen actual, la variable `HandleCraAsBlaFlag` se establece igual al valor proporcionado por el medio externo y la variable `NoRasOutputFlag` se establece igual a `HandleCraAsBlaFlag`.

15 - De cualquier otra manera, la variable `HandleCraAsBlaFlag` se establece igual a 0 y la variable `NoRasOutputFlag` se establece igual a 0.

El proceso de decodificación puede incluir lo siguiente para actualizar `LayerInitializedFlag` para una capa. Cuando la imagen actual es una imagen IRAP y una de las siguientes opciones es verdadera, `LayerInitializedFlag[ nuh_layer_id ]` se establece igual a 1.

20 - `nuh_layer_id` es igual a 0.

25 - `LayerInitializedFlag[ nuh_layer_id ]` es igual a 0 y `LayerInitializedFlag[ refLayerId ]` es igual a 1 para todos los valores de `refLayerId` iguales a `RefLayerId [ nuh_layer_id ][ j ]`, donde `j` está en el intervalo de 0 a `NumDirectRefLayers[ nuh_layer_id ] - 1`, inclusive.

30 Cuando `FirstPicInLayerDecodedFlag[ nuh_layer_id ]` es igual a 0, el proceso de decodificación para generar imágenes de referencia no disponibles puede invocarse antes de decodificar la imagen actual. El proceso de decodificación para generar imágenes de referencia no disponibles puede generar imágenes para cada imagen en una imagen de referencia establecida con valores predeterminados. El proceso de generación de imágenes de referencia no disponibles puede especificarse principalmente solo para la especificación de restricciones de sintaxis para las imágenes CL-RAS, donde una imagen CL-RAS puede definirse como una imagen con `nuh_layer_id` igual a `layerId` y `LayerInitializedFlag[ layerId ]` es igual a 0. En las operaciones HRD, es posible que se necesite tener en cuenta a las imágenes CL-RAS en la obtención de los tiempos de llegada y eliminación del CPB. Los decodificadores pueden ignorar cualquier imagen CL-RAS, ya que estas imágenes no se especifican para la salida y no tienen efecto en el proceso de decodificación de ninguna otra imagen que se especifique para la salida.

40 Las imágenes omitidas de acceso aleatorio entre capas (CL-RAS) pueden tener la propiedad de que cuando se invoca un mecanismo de arranque por capa (p. ej., cuando `NoCrasOutputFlag` es igual a 1), las imágenes CL-RAS no se emiten y pueden no decodificarse correctamente, ya que la imagen CL-RAS puede contener referencias a imágenes que no están presentes en la secuencia de bits. Se puede especificar que las imágenes CL-RAS no se usan como imágenes de referencia para el proceso de decodificación de imágenes que no son CL-RAS.

45 Las imágenes CL-RAS se pueden inferir. Por ejemplo, una imagen con `nuh_layer_id` igual a `layerId` se puede inferir que es una imagen CL-RAS cuando la capa `LayerInitializedFlag[ layerId ]` es igual a 0. Una imagen CL-RAS puede definirse como una imagen con `nuh_layer_id` igual a `layerId` de tal manera que `LayerInitializedFlag[ layerId ]` es igual a 0 cuando se inicia la decodificación de una imagen codificada con `nuh_layer_id` superior a 0.

50 Un árbol de capas puede definirse como un conjunto de capas de tal manera que cada capa del conjunto de capas es una capa predicha directa o indirecta o una capa de referencia directa o indirecta de al menos otra capa del conjunto de capas y ninguna capa fuera del conjunto de capas es una capa predicha directa o indirecta o una capa de referencia directa o indirecta de cualquier capa del conjunto de capas. Una capa predicha directa puede definirse como una capa para la cual otra capa es una capa de referencia directa. Una capa de referencia directa puede definirse como una capa que puede usarse para la predicción entre capas de otra capa para la cual la capa es la capa de referencia directa. Una capa predicha indirecta puede definirse como una capa para la cual otra capa es una capa de referencia indirecta. Una capa de referencia indirecta puede definirse como una capa que no es una capa de referencia directa de una segunda capa sino que 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 independiente puede definirse como una capa que no tiene ninguna capa de referencia directa. Una capa base puede definirse como una capa que tiene el valor de identificador de capa más bajo posible. Por ejemplo, en HEVC una capa base puede definirse como una capa con `nuh_layer_id` igual a 0. Una capa no base independiente puede definirse como una capa que es una capa independiente, pero no es la capa base.

Alternativamente, un árbol de capas puede definirse como un conjunto de capas donde cada capa tiene una relación de predicción entre capas con al menos otra capa del árbol de capas y ninguna capa fuera del árbol de capas tiene una relación de predicción entre capas con cualquier capa del árbol de capas.

- 5 Un subárbol de capas puede definirse como un subconjunto de las capas de un árbol de capas que incluye todas las capas de referencia de las capas dentro del subconjunto.

Se ha propuesto que una secuencia de bits no necesita necesariamente tener una capa base (es decir, una capa con `nuh_layer_id` igual a 0 en las extensiones de HEVC multicapa) incluida en la secuencia de bits o proporcionada externamente (en caso de escalabilidad híbrida de códec), pero la capa más baja puede ser una capa no base independiente. Debe entenderse que cuando se hace referencia a la capa base o `nuh_layer_id` igual a 0 en los ejemplos y realizaciones, los ejemplos y realizaciones pueden aplicarse o se pueden realizar similarmente para una capa no base independiente o `nuh_layer_id` igual a la de una capa no base independiente. La capa con el `nuh_layer_id` más bajo presente en la secuencia de bits puede considerarse como la capa base de la secuencia de bits.

En HEVC, los indicadores VPS `vps_base_layer_internal_flag` y `vps_base_layer_available_flag` pueden usarse para indicar la presencia y disponibilidad de la capa base de la siguiente manera: Si `vps_base_layer_internal_flag` es igual a 1 y `vps_base_layer_available_flag` es igual a 1, la capa base está presente en la secuencia de bits. De cualquier otra manera, si `vps_base_layer_internal_flag` es igual a 0 y `vps_base_layer_available_flag` es igual a 1, la capa base se proporciona por medios externos al proceso de decodificación multicapa de HEVC, es decir, las imágenes de capa base decodificadas, así como determinadas variables y elementos sintácticos para las imágenes de la capa base decodificada, se proporcionan al proceso de decodificación multicapa de HEVC. De cualquier otra manera, si `vps_base_layer_internal_flag` es igual a 1 y `vps_base_layer_available_flag` es igual a 0, la capa base no está disponible (ni está presente en la secuencia de bits ni se proporciona por medios externos) sino que el VPS incluye información de la capa base como si estuviera presente en la secuencia de bits. De cualquier otra manera (`vps_base_layer_internal_flag` es igual a 0 y `vps_base_layer_available_flag` es igual a 0), la capa base no está disponible (ni está presente en la secuencia de bits ni se proporciona por medios externos) sino que el VPS incluye información de la capa base como si se proporcionara por medios externos.

Una norma de codificación puede incluir un proceso de extracción de subsecuencia de bits, y tal se especifica, por ejemplo, en SVC, MVC y HEVC. El proceso de extracción de subsecuencia de bits se relaciona con la conversión de una secuencia de bits, típicamente mediante la eliminación de unidades NAL, en una subsecuencia de bits, al cual también puede hacerse referencia como subconjunto de secuencia de bits. La subsecuencia de bits aún permanece conforme a la norma. Por ejemplo, en HEVC, la secuencia de bits se crea mediante la exclusión de todas las unidades NAL VCL que tienen un valor de `TemporalId` mayor que un valor seleccionado y la inclusión de todas las otras unidades NAL VCL permanece conforme.

La norma HEVC (versión 2) incluye tres procesos de extracción de subsecuencia de bits. El proceso de extracción de subsecuencia de bits en la cláusula 10 de la norma HEVC es idéntico al de la cláusula F. 10.1 excepto que los requisitos de conformidad de la secuencia de bits para la subsecuencia de bits resultante se relajan en la cláusula F. 10.1, de modo que se puede usar también para la secuencia de bits en el que la capa base es externa (en cuyo caso `vps_base_layer_internal_flag` es igual a 0) o no disponible (en cuyo caso `vps_base_layer_available_flag` es igual a 0). Las unidades VCL NAL con `nuh_layer_id` igual a 0 no están presentes en una secuencia de bits con `vps_base_layer_available_flag` igual a 0, y la capa con el valor `nuh_layer_id` más bajo puede tratarse como una capa base en el proceso de semántica y/o decodificación. La cláusula F.10.3 de la norma HEVC (versión 2) especifica un proceso de extracción de subsecuencia de bits para conjuntos de capas adicionales, lo que da como resultado una subsecuencia de bits que no contiene la capa base. Los tres procesos de extracción de subsecuencia de bits operan de forma similar: el proceso de extracción de subsecuencia de bits toma un `TemporalId` y/o una lista de valores de `nuh_layer_id` como entrada y deriva una subsecuencia de bits (también conocido como subconjunto de secuencia de bits) eliminando de la secuencia de bits todas las unidades NAL con `TemporalId` mayor que el valor de `TemporalId` de entrada o el valor de `nuh_layer_id` que no está entre los valores de la lista de entrada de valores de `nuh_layer_id`. La cláusula F.10.3 gestiona las unidades NAL de ciertos tipos y con `nuh_layer_id` igual a 0 de forma especial y también establece el valor de `vps_base_layer_available_flag` en VPSs. El proceso de la cláusula F.10.3 puede especificarse de la siguiente manera:

- 55 – La secuencia de bits de salida de `outBitstream` se configura para que sea idéntico al secuencia de bits de entrada de `inBitstream`.
- Las unidades NAL con `nal_unit_type` distinto de `VPS_NUT`, `SPS_NUT`, `PPS_NUT`, `EOS_NUT` y `EOB_NUT` y con `nuh_layer_id` distinto de cualquier valor en la lista de entrada de valores de `nuh_layer_id`, `layerIdListTarget`, se eliminan de `outBitstream`.
- 60 – Las unidades NAL con `nal_unit_type` igual a `VPS_NUT`, `SPS_NUT`, `PPS_NUT` o `EOS_NUT` con `nuh_layer_id` distinto de 0 o de cualquier valor de `layerIdListTarget` se eliminan de `outBitstream`.
- 65 – Todas las unidades NAL con `TemporalId` mayor que el `TemporalId` de entrada, `tIdTarget`, se eliminan de `outBitstream`.

– vps_base_layer_available_flag en cada VPS se establece igual a 0.

5 Una norma o sistema de codificación puede hacer referencia a un punto operativo de término o similar, que puede indicar las capas y/o subcapas escalables bajo las cuales opera la decodificación y/o puede asociarse con una subsecuencia de bits que incluye las capas escalables y/o subcapas que se están decodificando. A continuación, se proporcionan algunas definiciones no limitantes de un punto operativo.

10 En HEVC, un punto operativo se define como una secuencia de bits creado a partir de otra secuencia de bits mediante la operación del proceso de extracción de subsecuencia de bits con el otra secuencia de bits, un TemporalId máximo objetivo y una lista de identificadores de capa objetivo como entradas.

15 El VPS de HEVC especifica conjuntos de capas y parámetros HRD para estos conjuntos de capas. Se puede usar un conjunto de capas como la lista de identificador de capas objetivo en el proceso de extracción de subsecuencia de bits. En HEVC, un conjunto de capas puede definirse como las capas representadas dentro de una secuencia de bits creado a partir de otra secuencia de bits mediante la operación del proceso de extracción de subsecuencia de bits con el otra secuencia de bits, un TemporalId máximo objetivo igual a 6 y la lista de identificadores de capa objetivo igual a la lista de identificadores de capa asociada con el conjunto de capas como entradas.

20 Un conjunto de capas adicionales puede definirse como un conjunto de capas de una secuencia de bits con un conjunto de capas de uno o más subárboles de capa no base. Un conjunto de capas adicionales puede entenderse como un conjunto de capas que no dependen de las capas excluidas del conjunto de capas adicionales. En HEVC, la sintaxis VPS especificada en la versión 1 de HEVC no permite la indicación de un conjunto de capas adicionales, es decir, se requiere que la capa base esté presente en los conjuntos de capas indicados en el VPS especificado en la versión 1 de HEVC. La sintaxis de extensión de VPS especificada en la versión 2 de HEVC permite indicar conjuntos de capas adicionales.

30 Los conjuntos de capas y conjuntos de capas adicionales pueden tratarse de manera equivalente cuando se usan, por ejemplo, en la derivación e indicación de conjuntos de capas de salida. A menos que el contexto indique lo contrario, el término “conjunto de capas” puede entenderse a continuación como un término colectivo que abarca tanto el término “conjunto de capas” como el término “conjunto de capas adicionales”, como se definió anteriormente.

35 Una capa de salida puede definirse como una capa cuyas imágenes decodificadas emite el proceso de decodificación. Las capas de salida pueden depender de qué subconjunto de la secuencia de bits multicapa se decodifica. Las imágenes emitidas por el proceso de decodificación pueden procesarse aún más, por ejemplo, puede realizarse una conversión de espacio de color del espacio de color YUV a RGB, y se pueden mostrar. Sin embargo, el procesamiento y/o la visualización adicionales pueden considerarse procesos externos del decodificador y/o el proceso de decodificación y podrían no ocurrir.

40 En flujos de bits de vídeo multicapa, una definición de punto operativo puede incluir una consideración de un conjunto de capas de salida objetivo. Por ejemplo, un punto operativo puede definirse como una secuencia de bits que se crea a partir de otra secuencia de bits mediante la operación del proceso de extracción de subsecuencia de bits con el otra secuencia de bits, una subcapa temporal máxima objetivo (p. ej., un TemporalId máximo objetivo) y una lista de identificadores de capa objetivo como entradas, y que está asociado con un conjunto de capas de salida. Alternativamente, puede usarse otro término, tal como un punto operativo de salida, cuando se hace referencia a un punto operativo y al conjunto asociado de capas de salida. Por ejemplo, en MV-HEVC/SHVC, un punto operativo de salida puede definirse como una secuencia de bits que se crea a partir de una secuencia de bits de entrada mediante la operación del proceso de extracción de subsecuencia de bits con la secuencia de bits de entrada, un TemporalId máximo objetivo y una lista de identificadores de capa objetivo como entradas, y que está asociado con un conjunto de capas de salida.

55 Un conjunto de capas de salida (OLS) puede definirse como un conjunto de capas que consisten en las capas de uno de los conjuntos de capas especificados, donde una o más capas en el conjunto de capas están marcadas como capas de salida. Una capa de salida puede definirse como una capa de un conjunto de capas de salida que se emite cuando el decodificador y/o el HRD operan usando el conjunto de capas de salida como el conjunto de capas de salida objetivo. En MV-HEVC/SHVC, la variable TargetOlsIdx puede especificar qué conjunto de capas de salida es el conjunto de capas de salida objetivo estableciendo TargetOlsIdx igual al índice del conjunto de capas de salida que es el conjunto de capas de salida objetivo. Un conjunto de capas de salida objetivo puede definirse como el conjunto de capas de salida para el cual el índice es igual a TargetOlsIdx. TargetOlsIdx puede establecerse, por ejemplo, mediante el HRD y/o puede establecerse por medios externos, por ejemplo, por un reproductor o similar a través de una interfaz proporcionada por el decodificador. En MV-HEVC/SHVC, una capa de salida puede definirse como una capa de un conjunto de capas de salida que se emite cuando TargetOlsIdx es igual al índice del conjunto de capas de salida.

Una capa necesaria puede definirse como una capa en un punto operativo de salida asociado con un conjunto de capas de salida, donde la capa es una capa de salida del conjunto de capas de salida o una capa de referencia de una capa de salida del conjunto de capas de salida.

5 MV-HEVC/SHVC habilitan la derivación de un conjunto de capas de salida “predeterminado” para cada conjunto de capas especificado en el VPS usando un mecanismo específico o indicando las capas de salida explícitamente. Se han especificado dos mecanismos específicos: puede especificarse en el VPS que cada capa es una capa de salida o que solo la capa más alta es una capa de salida en un conjunto de capas de salida “predeterminado”. Podrían no considerarse las capas de imagen auxiliares cuando se determina si una capa es una capa de salida usando los
10 mecanismos específicos mencionados. Además de los conjuntos de capas de salida “predeterminados”, la extensión VPS permite especificar conjuntos de capas de salida adicionales con capas seleccionadas marcadas como capas de salida.

15 Puede hacerse referencia a un conjunto de capas particular, conjunto de capas adicionales o conjunto de capas de salida a través de un índice, y los índices pueden asignarse de acuerdo con los conjuntos de capas de orden, conjuntos de capas adicionales y conjuntos de capas de salida que se indican en el VPS. Los conjuntos de capas de salida pueden usar el mismo espacio de índices que los conjuntos de capas y capas adicionales de modo que los conjuntos de capas de salida “predeterminados” para cada conjunto de capas y conjunto de capas adicionales tengan el índice del respectivo conjunto de capas o capa adicional, y los conjuntos de capas de salida adicionales
20 usan valores de índice más allá de que usan los conjuntos de capas o conjuntos de capas adicionales.

Un transmisor, una puerta de enlace o similares pueden seleccionar las capas y/o subcapas transmitidas de una secuencia de bits de vídeo escalable. Los términos “extracción de capas” o “conmutación descendente de capas”
25 pueden hacer referencia a la transmisión de menos capas que lo que está disponible en la secuencia de bits recibido por el transmisor, la puerta de enlace o similares. La conmutación ascendente de capas puede hacer referencia a la transmisión de capas adicionales en comparación con las transmitidas antes de la conmutación ascendente de capas por parte del transmisor, la puerta de enlace o similares, es decir, el reinicio de la transmisión de una o más capas cuya transmisión se detuvo antes en la conmutación descendente de capas. De manera similar a la conmutación descendente y/o ascendente de capas, el transmisor, la puerta de enlace o similares pueden realizar una conmutación descendente y/o ascendente de las subcapas temporales. El transmisor, la puerta de enlace o similares también
30 pueden realizar la conmutación descendente y/o ascendente de capas y subcapas. La conmutación ascendente y/o descendente de capas y subcapas se puede llevar a cabo en la misma unidad de acceso o similar (es decir, prácticamente en simultáneo) o se puede llevar a cabo en diferentes unidades de acceso o similares (es decir, prácticamente en distintos momentos).

35 Mientras que un conjunto constante de capas de salida se adapta bien a casos de uso y flujos de bits donde la capa más alta permanece sin cambios en cada unidad de acceso, pueden no admitir casos de uso donde la capa más alta cambia de una unidad de acceso a otra. Por lo tanto, se ha propuesto que los codificadores puedan especificar el uso de capas de salida alternativas dentro de la secuencia de bits y, en respuesta al uso especificado de decodificadores de capas de
40 salida alternativas, emitir una imagen decodificada a partir de una capa de salida alternativa en ausencia de una imagen en una capa de salida dentro de la misma unidad de acceso. Existen varias posibilidades para marcar capas de salida alternativas. Por ejemplo, cada capa de salida en un conjunto de capas de salida puede asociarse con una capa de salida alternativa mínima, y se pueden usar elementos de sintaxis orientados a capas de salida para especificar las capas de salida alternativas para cada capa de salida. Alternativamente, el mecanismo de conjunto de capas de salida alternativas puede estar limitado a usarse solo para conjuntos de capas de salida que contienen solo una capa de salida, y se pueden
45 usar elementos de sintaxis orientados a capas de salida para especificar las capas de salida alternativas para la capa de salida del conjunto de capas de salida. Alternativamente, como se especifica en HEVC, el mecanismo de conjunto de capas de salida alternativo puede estar limitado a usarse solo para conjuntos de capas de salida que contengan solo una capa de salida, y una marca orientada al conjunto de capas de salida (`alt_output_layer_flag[olsidx]` en HEVC) puede usarse para especificar que cualquier capa de referencia directa o indirecta de la capa de salida puede servir como una
50 capa de salida alternativa para la capa de salida del conjunto de capas de salida. Alternativamente, el mecanismo de conjunto de capas de salida alternativo puede estar limitado a usarse solo para flujos de bits o CVS en los que todos los conjuntos de capas de salida especificados contienen solo una capa de salida, y las capas de salida alternativas pueden indicarse mediante elementos de sintaxis orientados a secuencia de bits o CVS. Las capas de salida alternativas pueden especificarse, por ejemplo, enumerando, p. ej., dentro de VPS las capas de salida alternativas (p. ej., usando sus
55 identificadores de capa o índices de la lista de capas de referencia directas o indirectas), indicando una capa de salida alternativa mínima (p. ej., usando su identificador de capa o su índice dentro de la lista de capas de referencia directas o indirectas) o una marca que especifique que cualquier capa de referencia directa o indirecta es una capa de salida alternativa. Cuando se habilita más de una capa de salida alternativa para usarse, puede especificarse que la primera imagen de referencia entre capas directas o indirectas presente en la unidad de acceso ordenada de mayor a menor por
60 identificador de capa hasta la capa de salida alternativa mínima es salida.

La salida de imágenes en la codificación escalable puede controlarse, por ejemplo, de esta manera: Para cada imagen, `PicOutputFlag` se deriva primero en el proceso de decodificación de manera similar a la de una secuencia
65 de bits de una sola capa. Por ejemplo, la `pic_output_flag` incluida en la secuencia de bits para la imagen puede tenerse en cuenta en la derivación de `PicOutputFlag`. Cuando se ha decodificado una unidad de acceso, las capas

de salida y las posibles capas de salida alternativas se usan a fin de actualizar la PicOutputFlag para cada imagen de la unidad de acceso.

Cuando una secuencia de bits especifica el uso de un mecanismo de capas de salida alternativas, el proceso de decodificación puede operar de la siguiente manera cuando se debe controlar la salida de imagen decodificada del proceso de decodificación. Aquí, se supone que la decodificación de HEVC está en uso y alt_output_layer_flag[TargetOlsIdx] es igual a 1, pero el proceso de decodificación podría realizarse de manera similar con otros códecs. Cuando se completa la decodificación de una imagen, la variable PicOutputFlag de la imagen puede establecerse de la siguiente manera:

Si LayerInitializedFlag[nuh_layer_id] es igual a 0, PicOutputFlag se establece igual a 0.

De otro modo, si la imagen actual es una imagen RASL y la NoRaslOutputFlag de la imagen IRAP asociada es igual a 1, PicOutputFlag se establece igual a 0.

De otro modo, PicOutputFlag se establece igual a pic_output_flag, donde pic_output_flag es un elemento de sintaxis asociado con la imagen, por ejemplo, portado en la cabecera de fragmento de los segmentos codificados de la imagen. Además, cuando se completa la decodificación de la última imagen de una unidad de acceso, se puede actualizar la PicOutputFlag de cada imagen decodificada de la unidad de acceso de la siguiente manera (antes de la decodificación de la siguiente imagen):

Si alt_output_layer_flag[TargetOlsIdx] es igual a 1 y la unidad de acceso actual no contiene una imagen en la capa de salida o contiene una imagen en la capa de salida que tiene una PicOutputFlag igual a 0, se aplican los siguientes pasos ordenados:

La lista nonOutputLayerPictures se establece como la lista de las imágenes de la unidad de acceso con PicOutputFlag igual a 1 y con valores nuh_layer_id entre los valores nuh_layer_id de las capas de referencia de la capa de salida.

Cuando la lista nonOutputLayerPictures no está vacía, la imagen con el valor nuh_layer_id más alto en la lista nonOutputLayerPictures se elimina de la lista nonOutputLayerPictures.

La PicOutputFlag para cada imagen que se incluye en la lista nonOutputLayerPictures se establece igual a 0.

De otro modo, PicOutputFlag para imágenes que no están incluidas en una capa de salida se establece igual a 0.

Como se describe en el párrafo anterior, cuando el mecanismo de capa de salida alternativo está en uso, es posible que necesite completarse la decodificación de una unidad de acceso antes de que se pueda determinar qué imágenes decodificadas de la unidad de acceso emite el proceso de decodificación.

Un identificador de recursos uniforme (URI) puede definirse como una cadena de caracteres utilizados para identificar un nombre de un recurso. Dicha identificación permite la interacción con representaciones del recurso a través de una red, utilizando protocolos específicos. Un URI se define a través de un esquema que especifica una sintaxis concreta y un protocolo asociado para el URI. El localizador de recursos uniforme (URL) y el nombre de recurso uniforme (URN) son formas de URI. Un URL puede definirse como un URI que identifica un recurso web y especifica los medios para actuar sobre el recurso u obtener una representación de este, especificando tanto su mecanismo de acceso primario como la ubicación de red. Un URN puede definirse como un URI que identifica un recurso por nombre en un espacio de nombres particular. Se puede usar un URN para identificar un recurso sin implicar su ubicación o cómo acceder a él.

Las normas disponibles de formato de archivo de medios incluyen el formato de archivo de medios de base ISO (ISO/IEC 14496-12, que puede abreviarse como ISOBMFF), el formato de archivo MPEG-4 (ISO/IEC 14496-14, también conocido como formato MP4), el formato de archivo para vídeo estructurado de unidad NAL (ISO/IEC 14496-15) y el formato de archivo 3GPP (3GPP TS 26.244, también conocido como formato 3GP). ISO/IEC 14496-15 especifica el almacenamiento de flujos de bits de H.264/AVC y/o HEVC y/o sus extensiones en archivos compatibles con ISOBMFF. El formato de archivo ISO es la base para la derivación de todos los formatos de archivo mencionados anteriormente (sin incluir el formato de archivo ISO en sí). Estos formatos de archivo (incluido el formato de archivo ISO en sí) se denominan generalmente como la familia ISO de formatos de archivos.

Algunos conceptos, estructuras y especificaciones de ISOBMFF se describen a continuación como un ejemplo de un formato de archivo de contenedor, en cuya base se pueden implementar las realizaciones. Los aspectos de la invención no se limitan a ISOBMFF, sino que se proporciona la descripción para una base posible sobre la que la invención puede estar parcialmente o totalmente desarrollada.

Un bloque de construcción en el formato de archivo de medios de base ISO se denomina una caja. Cada caja puede tener una cabecera y una carga útil. El encabezado 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. Adicionalmente, para algunos tipos de caja, puede permitirse tener más de una caja presente en un archivo. Por lo tanto, puede considerarse que el formato de archivo de medios de base ISO especifica una estructura jerárquica de cajas. Cada caja del archivo de medios de base ISO puede identificarse mediante un código de cuatro caracteres (4CCs). El encabezado puede proporcionar información sobre el tipo y tamaño de la caja.

Según la familia ISO de formatos de archivo, un archivo puede incluir datos de medios y metadatos que pueden estar encerrados en cajas separadas. En una realización de ejemplo, los datos de medios pueden proporcionarse en una caja de datos de medios (mdat) y la caja de película (moov) puede usarse para encerrar los metadatos. En algunos casos, para que un archivo sea operable, deben estar presentes ambas cajas, mdat y moov. La caja de película (moov) puede incluir una o más pistas, y cada pista puede residir en una caja de pista (trak) correspondiente. Cada pista está asociada con un controlador, identificado por un código de cuatro caracteres, que especifica el tipo de pista. Las pistas de secuencia de vídeo, audio e imagen pueden denominarse colectivamente pistas de medios, y contienen un flujo de medios elemental. Otros tipos de vía comprenden pistas de rastro y pistas de metadatos programados. Las pistas comprenden muestras, tales como cuadros de vídeo o audio. Una pista de audio hace referencia a muestras (que también pueden denominarse muestras de medios) formateadas de acuerdo con un formato de compresión de medios (y su encapsulación en el formato de archivo de medios de base ISO). Una pista de rastro hace referencia a muestras de rastro, que contienen instrucciones de recetario para construir paquetes para su transmisión a través de un protocolo de comunicación indicado. Las instrucciones de recetario pueden incluir una guía para la construcción de encabezados de paquete y puede incluir la construcción de carga útil de paquetes. En la construcción de carga útil de paquetes, se pueden hacer referencia a los datos que residen en otras pistas o elementos. Como tal, por ejemplo, los datos que residen en otras pistas o elementos pueden indicarse mediante una referencia acerca de qué dato en una pista o elemento particular debe copiarse en un paquete durante el proceso de construcción de paquete. Una pista de metadatos programados puede hacer referencia a muestras que describen muestras de medios y/o de rastro referenciadas. Para la presentación de un tipo de medio, se puede seleccionar una pista de medios. Las muestras de una pista pueden asociarse implícitamente con números de muestra que pueden incrementarse, p. ej., en 1 en el orden de decodificación indicado de las muestras. La primera muestra en una pista puede estar asociada con el número de muestra 1.

La caja “trak” contiene una caja de tabla de muestras. La caja de tabla de muestras comprende, p. ej., todo el tiempo y la indexación de datos de las muestras de medios en una pista. Se requiere que el cuadro de tabla de muestras contenga una caja de descripción de muestras. La caja de descripción de muestras incluye un campo de recuento de entradas, que especifica el número de entradas de muestra incluidas en la caja. Se requiere que el cuadro de descripción de muestras contenga al menos una entrada de muestra. El formato de entrada de muestra depende del tipo de controlador para la pista. Las entradas de muestra dan información detallada sobre el tipo de codificación usado y cualquier información de inicialización necesaria para esa codificación.

El formato de archivo de medios de base ISO no limita una presentación a que esté contenida en un archivo. Como tal, una presentación puede estar contenida en varios archivos. Como ejemplo, un archivo puede incluir los metadatos para toda la presentación y, por lo tanto, puede incluir todos los datos de medios para hacer que la presentación sea autónoma. Los demás archivos, si los hay, no necesariamente tendrán que formatearse al formato de archivo multimedia de base ISO, y pueden usarse para incluir datos de medios, y también pueden incluir datos de medios no utilizados u otra información. El formato de archivo de medios de base ISO hace referencia solamente a la estructura del archivo de la presentación. El formato de los archivos de datos de medios puede estar limitado por el formato del archivo de medios de base ISO o sus formatos derivados solo en que los datos de medios en los archivos de medios se formatean como se especifica en el formato de archivo de medios de base ISO o sus formatos derivados.

La capacidad de hacer referencia a archivos externos se puede lograr a través de referencias de datos. En algunos ejemplos, una caja de descripción de muestra incluida en cada pista puede proporcionar una lista de entradas de muestra, cada una de las cuales proporciona información detallada sobre el tipo de codificación usado y cualquier información de inicialización necesaria para esa codificación. Todas las muestras de un segmento y todas las muestras de un fragmento de pista pueden usar la misma entrada de muestra. Un fragmento puede definirse como un conjunto contiguo de muestras para una pista. La caja de referencia de datos (dref), que también puede incluirse en cada pista, puede definir una lista indexada de localizadores de recursos uniformes (URL), nombres de recursos uniformes (URN) y/o autorreferencias al archivo que contiene los metadatos. Una entrada de muestra puede apuntar a un índice de la caja de referencia de datos, indicando así el archivo que contiene las muestras del segmento o fragmento de pista respectivo.

Los fragmentos de película pueden usarse, p. ej., cuando se graba contenido en archivos ISO, p. ej., para evitar perder datos si una aplicación de grabación falla, se queda sin espacio en la memoria o si se produce algún otro incidente. Sin fragmentos de película, la pérdida de datos puede ocurrir porque el formato de archivo puede requerir que todos los metadatos, por ejemplo, la caja de película, se graben en un área contigua del archivo. Además, cuando se graba un archivo, puede no haber suficiente cantidad de espacio en la memoria (p. ej., memoria de acceso aleatorio RAM para

almacenar una caja de película para el tamaño del almacenamiento disponible, y volver a calcular el contenido de una caja de película cuando la película está cerrada puede ser demasiado lento. Además, los fragmentos de película pueden permitir la grabación y la reproducción simultáneas de un archivo usando un analizador de archivos ISO regular. Además, podría necesitarse una duración más pequeña del almacenamiento en memoria intermedia inicial para la descarga progresiva, p. ej., la recepción y la reproducción de un archivo simultáneamente cuando se usan fragmentos de película y la caja de película inicial es más pequeña en comparación con un archivo con el mismo contenido medios, pero estructurado sin fragmentos de película.

La característica de fragmento de película puede permitir dividir en múltiples piezas los metadatos que de otro modo podrían residir en la caja de película. Cada pieza puede corresponder a un cierto período de una pista. En otras palabras, la característica de fragmento de película puede permitir la intercalación de metadatos de archivo y datos de medios. En consecuencia, el tamaño de la caja de película puede ser limitado y pueden ocurrir los casos de uso mencionados anteriormente.

En algunos ejemplos, las muestras de medios de los fragmentos de película pueden encontrarse 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 previamente habría estado 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 seguirán los fragmentos de película en el mismo archivo. Los fragmentos de película pueden extender a su tiempo la presentación que está asociada a la caja moov.

Dentro del fragmento de película puede haber un conjunto de fragmentos de pista, de cero a una pluralidad por pista. A su vez, los fragmentos de pista pueden incluir de cero a una pluralidad de ejecuciones de pista, cuyos documentos son, cada uno, una seguidilla contigua de muestras para esa pista. Dentro de estas estructuras, muchos campos son opcionales y pueden usar valores 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 relativos 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 se puede definir como 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 (cuyos metadatos proporciona la caja moof) y no contiene muestras de cualquier otro fragmento de película (es decir, cualquier otra caja moof).

El formato de archivo de medios de base ISO contiene tres mecanismos para metadatos programados que pueden asociarse con muestras particulares: grupos de muestras, pistas de metadatos programados e información auxiliar de muestra. La especificación derivada puede proporcionar una funcionalidad similar con uno o más de estos tres mecanismos.

Una agrupación de muestras en el formato de archivo de medios de base ISO y sus derivados, tal como el formato de archivo AVC y el formato de archivo SVC, pueden definirse como una asignación de que cada muestra en una pista sea un miembro de un grupo de muestras, con base en un criterio de agrupamiento. Un grupo de muestras en una agrupación de muestras no se limitan a ser muestras contiguas y puede contener muestras no adyacentes. Como puede haber más de un agrupamiento de muestras para las muestras de una pista, cada agrupamiento de muestras puede tener un campo de tipo para indicar el tipo de agrupación. Los agrupamientos de muestras pueden representarse mediante dos estructuras de datos enlazadas: (1) una caja SampleToGroup (caja sbgp) representa la asignación de muestras a grupos de muestra; y (2) una caja de SampleGroupDescription (caja sgpd) contiene una entrada de grupo de muestras para cada grupo de muestras que describe las propiedades del grupo. Puede haber múltiples instancias de cajas de SampleToGroup y SampleGroupDescription con base en diferentes criterios de agrupamiento. Estos pueden distinguirse por un campo de tipo usado para indicar el tipo de agrupamiento.

El formato de archivo Matroska es capaz de almacenar, pero no se limita a, pistas de vídeo, audio, imagen o subtítulo en un solo archivo. Las extensiones de archivo Matroska incluyen .mkv para vídeo (con subtítulos y audio), mk3d para vídeo estereoscópico, .mka para archivos de solo audio y .mks para solo subtítulos. Matroska puede usarse como formato de base para formatos de archivo derivados, tales como WebM.

Matroska utiliza el metalenguaje binario extensible (EBMLinter-layer) como base. EBML especifica un formato alineado de binario y octeto (byte) inspirado por el principio de XML. El propio EBML es una descripción generalizada de la técnica de marcado binario. Un archivo Matroska consiste en elementos que forman un "documento" EBML. Los elementos incorporan un ID de elemento, un descriptor para el tamaño del elemento y los datos binarios en sí. Los elementos pueden estar anidados.

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 compuesto por) un segmento. Los datos de medios en archivos Matroska se organizan en clústeres (o elementos de clúster), cada uno de los cuales contiene típicamente unos pocos segundos de datos de medios. Un clúster comprende elementos de BlockGroup que, a su

vez, comprenden elementos de bloque. Un elemento de señales comprende metadatos que pueden ayudar en el acceso aleatorio o la búsqueda y pueden incluir punteros de archivo o marcas de tiempo respectivas para puntos de búsqueda.

Los formatos de archivo de transporte o formatos de segmento que pueden emplearse pueden clasificarse a grandes rasgos en diferentes clases. En una clase de ejemplo, los archivos transmitidos pueden cumplir con un formato de archivo existente que puede usarse para la reproducción de archivos en directo. Por ejemplo, los archivos transmitidos pueden cumplir con el formato de archivo de medios de base ISO o el perfil de descarga progresiva del formato de archivo del Proyecto de Asociación de Tercera Generación (3GPP). En otra clase de ejemplo, los archivos transmitidos pueden ser similares a los archivos formateados de acuerdo con un formato de archivo existente usado para la reproducción de archivos en directo. Por ejemplo, los archivos transmitidos pueden ser fragmentos de un archivo de servidor, que puede no ser autónomo para la reproducción individualmente. En otro enfoque, los archivos a transmitir pueden cumplir con un formato de archivo existente que puede usarse para la reproducción de archivos en directo, pero los archivos pueden transmitirse únicamente parcialmente y, por lo tanto, la reproducción de tales archivos puede requerir reconocimiento y capacidad de administrar archivos parciales.

Una extensión de correo de Internet multipropósito (MIME) es una extensión a un protocolo de correo electrónico que hace posible transmitir y recibir diferentes tipos de archivos de datos en Internet, por ejemplo, vídeo y audio, imágenes, software, etc. Un tipo de medios de Internet es un identificador usado en Internet para indicar el tipo de datos que contiene un archivo. Dichos tipos de medios de Internet también se pueden denominar tipos de contenido. Existen varias combinaciones de tipo/subtipo MIME que pueden contener diferentes formatos de medios. Una entidad transmisora puede incluir la información del tipo de contenido en un encabezado MIME al comienzo de una transmisión de medios. Por lo tanto, una entidad receptora puede tener que examinar los detalles de tal contenido de medios para determinar si los elementos específicos pueden reproducirse dado un conjunto disponible de códecs. Especialmente cuando el sistema final tiene recursos limitados, o la conexión al sistema final tiene un ancho de banda limitado, puede ser útil conocer el tipo de contenido para saber si se puede representar el contenido.

RFC 6381 especifica dos parámetros, 'codecs' y 'profiles', que se usan con varios tipos MIME o combinaciones de tipo/subtipo para permitir la especificación inequívoca de los códecs que usan los formatos de medios contenidos dentro, o los perfiles del formato general del contenedor.

Cuando se marca el contenido con los códecs específicos indicados para reproducir el medio contenido, los sistemas receptores puedan determinar si el sistema final admite los códecs, y si no, pueden realizar acciones apropiadas (tales como rechazar el contenido, enviar notificación de la situación, transcódecificar el contenido a un tipo admitido, obtener e instalar los códecs requeridos, una inspección adicional para determinar si será suficiente admitir un subconjunto de los códecs indicados, etc.).

De manera similar, los perfiles pueden proporcionar una indicación general, al receptor, de las especificaciones con las que cumple el contenido. Esta es una indicación de la compatibilidad del formato del contenedor y su contenido con alguna especificación. El receptor puede ser capaz de trabajar en el grado en que puede manejar y reproducir el contenido examinando cuáles de los perfiles declarados admite, y lo que significan.

Aunque una motivación para MIME es la capacidad de identificar el tipo de medio específico de una parte de mensaje, puede no siempre ser posible saber con solo ver el tipo y subtipo de MIME qué formatos de medios específicos están contenidos en la parte de cuerpo o qué códecs se indican para reproducir el contenido.

Existen varios tipos/subtipos de medios (actualmente registrados o implementado con registro pendiente) que contienen códecs elegidos de un conjunto. En ausencia de los parámetros 'codecs' y/o 'profiles', puede ser necesario examinar cada elemento de medios para determinar los códecs u otras características requeridas para reproducir el contenido.

El valor del parámetro de códecs puede especificarse de la siguiente manera: Un valor único, o una lista de valores separados por comas que identifican los códecs indicados para reproducir el contenido en la parte de cuerpo. Cada valor puede comprender uno o más elementos separados por puntos. El espacio de nombres para el primer elemento se determina mediante el tipo MIME. El espacio de nombres para cada elemento posterior se determina según el elemento anterior.

En las extensiones de HEVC multicapa, se indica una combinación de perfil-estrato-nivel para cada capa necesaria en cada conjunto de capas de salida. Para las extensiones de HEVC multicapa, un punto de conformidad puede definirse como una lista de combinaciones de perfil-estrato-nivel, donde la lista incluye la combinación de perfil-estrato-nivel para cada capa necesaria de un conjunto de capas de salida particular. Se puede indicar un diferente valor de perfil-estrato-nivel (particularmente diferente valor de nivel) para la misma capa en conjuntos de capa de salida diferentes, ya que los requisitos de memoria intermedia de imágenes decodificadas difieren dependiendo de qué capas son capas necesarias y qué capas son capas de salida, y los requisitos de memoria intermedia de imágenes decodificadas son parte de las especificaciones de nivel. Las capacidades de decodificación se pueden indicar como una lista de combinaciones de perfil-estrato-nivel. Por estas razones mencionadas, un único valor de perfil-estrato-nivel dentro de un parámetro MIME de

códecs podría no ser suficiente para describir una secuencia de bits HEVC multicapa almacenado, p. ej., en un archivo compatible con ISOBMFF.

El parámetro de códecs del tipo MIME para archivos basados en ISOBMFF puede especificarse de esta manera para tales códecs en capas (p. ej., extensiones de HEVC multicapa) para las cuales la señalización de nivel de perfil está orientada a la capa (u orientada a la partición de secuencia de bits) en lugar de orientada al secuencia de bits o flujo de subbits. Debe entenderse que, aunque los siguientes párrafos hacen referencia al parámetro de códecs, podría nombrarse un parámetro de otro nombre alternativa o adicionalmente.

Cuando se usa el parámetro 'codecs' de un tipo MIME, como se define en RFC 6381, los siguientes párrafos documentan los subparámetros cuando el tipo MIME identifica un formato de archivo de la familia ISOBMFF y el parámetro 'codecs' comienza con un código de entrada de muestra de ISO/IEC 14496-15. Sin embargo, debe entenderse que el parámetro de códecs podría especificarse de manera similar para contenedores o formatos distintos de ISOBMFF y/o códecs distintos de AVC o HEVC.

El parámetro de códecs puede tener la siguiente estructura (como se especifica en RFC 6381): *ListItem* (, *ListItem*N)*

donde un asterisco (**) indica una repetición de 0 o más veces.

Cada *ListItem* puede tener la siguiente estructura:

SampleEntryType.*ProfileTierLevel*(.*SampleEntryType*N.*ProfileTierLevel*N)*

Debe entenderse que el carácter separador '.' puede ser igualmente cualquier otro carácter separador, o que pueden usarse varios diferentes caracteres separadores, posiblemente para diferente ubicación o propósito. Por ejemplo, se puede usar un carácter separador '!' en lugar de '.' frente a *SampleEntryType*N.

Un par de *SampleEntryType* y *ProfileTierLevel* se puede denominar como una subcadena de perfil-estrato-nivel.

El tipo de entrada de la muestra (*SampleEntryType*) puede ser el código de cuatro caracteres de la entrada de muestra de la pista que contiene los datos codificados. La especificación de subcadena de perfil-estrato-nivel puede conservarse sin cambios en comparación con la especificación de tipo MIME de códecs para HEVC de una sola capa.

Cuando el tipo de entrada de la muestra es un código que indica un códec a partir de la especificación de codificación de vídeo avanzada (AVC, recomendación ITU-T H.264 o ISO/IEC 14496-10), tal como 'avc1', 'avc2', 'avc3', 'avc4', 'svc1', 'svc2', 'mvc1', 'mvc2', 'mvc3' y 'mvc4', lo que indica AVC (H.264), codificación de vídeo escalable (SVC) o codificación de vídeo multivista (MVC), el elemento *ProfileTierLevel* es la representación hexadecimal de los siguientes tres bytes en la unidad NAL de conjunto de parámetros de secuencia (subconjunto) especificada en H.264/AVC:

profile_idc;

el byte que contiene las marcas constraint_set (actualmente desde constraint_set0_flag hasta constraint_set5_flag y reserved_zero_2bits); y

level_idc.

Debe observarse que las entradas de muestra 'avc1', 'avc2', 'avc3' y 'avc4' no necesariamente indican que los medios solo contienen unidades NAL de AVC. De hecho, los medios pueden codificarse como un perfil SVC o MVC y, por lo tanto, contener unidades de NAL de SVC o MVC. Para poder determinar qué códec se usa, puede hacer falta más información (profile_idc). También puede ser necesario que reserved_zero_2bits sea igual a 0 en H.264/AVC, pero ITU-T ISO/IEC puede especificar otros valores para ello.

Cuando el contenido SVC o MVC se codifica en un modo compatible con AVC, la descripción de la muestra puede incluir tanto un registro de configuración AVC como un registro de configuración SVC o MVC. En esas circunstancias, puede recomendarse que se reporten los dos registros de configuración porque pueden contener diferentes valores de perfil de AVC, nivel e indicador de compatibilidad. Por lo tanto, los códecs reportados incluirían la descripción de la muestra (por ejemplo, 'avc1') dos veces, y los valores de uno de los registros de configuración formarían la información 'avcoti' en cada uno.

Cuando el tipo de entrada de la muestra es un código que indica un códec de la especificación de codificación de vídeo de alta eficacia (HEVC, recomendación ITU-T H.265 o ISO/IEC 23008-2), tal como 'hev1', 'hvc1', '1hv1' o '1he1', la subcadena perfil-estrato-nivel se especifica como una serie del código y los valores de la lista siguiente, separados, p. ej., por caracteres de punto ("."). Los valores coinciden con los de la estructura sintáctica

mejora se ajusta al Scalable Main Profile, Main Tier, Level 4, la pista que contiene la capa base usa la entrada de muestra de tipo 'hev1' y la capa base se ajusta al Main Profile, Main Tier, Level 3.1. Cada capa es una pista separada. Hay un conjunto de capas de salida que contiene la capa de mejora y la capa base como capas necesarias, y un conjunto de capas de salida que contiene solo la capa base. codecs=1hv1.A7.1.L120.B0, avc1.64081F puede indicar que la pista que contiene la capa de mejora usa el tipo de entrada de muestra 'lhv1', la capa de mejora se ajusta al Scalable Main Profile, Main Tier, Level 4, la capa de mejora contiene imágenes progresivas y no empaquetadas y la capa de mejora realiza predicciones a parte de una capa de base con codificación AVC. La capa base codificada de AVC utiliza el tipo de entrada de muestra 'avc1', y la capa base se ajusta al Progressive High Profile, Level 3.1.

La norma internacional ISO/IEC 23009-1 especifica el flujo dinámico adaptable través de HTTP (DASH). Algunos conceptos, formatos y operaciones de MPEG-DASH se describen a continuación como un ejemplo de un sistema de flujo de vídeo, en donde pueden implementarse las realizaciones. Los aspectos de la invención no se limitan a MPEG-DASH, sino que se proporciona la descripción para una base posible sobre la que la invención puede estar parcialmente o totalmente desarrollada.

En el flujo dinámico adaptable dinámica a través de HTTP (DASH), el contenido de medios puede capturarse y almacenarse en un servidor HTTP y puede entregarse utilizando HTTP. El contenido puede almacenarse en el servidor en dos partes: Descripción de presentación de medios (MPD), que describe un manifiesto del contenido disponible, sus diversas alternativas, sus direcciones de URL y otras características; y segmentos, que contienen los flujos de bits de medios reales en forma de segmentos, en un único archivo o en varios archivos. Para reproducir el contenido, el cliente DASH puede obtener la MPD, p. ej., usando HTTP, correo electrónico, unidad portátil, difusión u otros métodos de transporte. Mediante el análisis de la MPD, el cliente DASH puede aprender sobre la sincronización del programa, disponibilidad del contenido de medios, los tipos de medios, las resoluciones, los anchos de banda mínimos y máximos, y la existencia de diversas alternativas codificadas de componentes de medios, características de accesibilidad y gestión de derechos digitales requeridos (DRM), ubicaciones de componentes de medios en la red y otras características de contenido. Usando esta información, el cliente DASH puede seleccionar la alternativa codificada apropiada e iniciar el flujo del contenido mediante la extracción de los segmentos usando, p. ej., solicitudes GET de HTTP. Después del almacenamiento adecuado en memoria intermedia en caso de variaciones de rendimiento de red, el cliente puede continuar recuperando los segmentos posteriores y también supervisar las fluctuaciones de ancho de banda de la red. El cliente puede decidir cómo adaptarse al ancho de banda disponible mediante la extracción de segmentos de diferentes alternativas (con velocidades de bits más bajas o más altas) para mantener una memoria intermedia adecuada.

La descripción de presentación de medios (MPD) puede proporcionar información para los clientes a fin de establecer un flujo dinámico adaptable a través de HTTP. La MPD puede contener información que describe la presentación de medios, tal como un localizador de recursos uniforme (URL) para HTTP de cada segmento a fin de hacer la solicitud de GET Segment. En DASH, el modelo de datos jerárquico puede usarse para estructurar la presentación de medios como se muestra en la Figura 6. Una presentación de medios puede comprender una secuencia de uno o más períodos, cada período puede contener uno o más grupos, cada grupo puede contener uno o más conjuntos de adaptación, cada conjunto de adaptación puede contener una o más representaciones y cada representación puede comprender uno o más segmentos. Una representación es una de las opciones alternativas del contenido de medios o un subconjunto de estas que puede diferir en la elección de codificación, p. ej., por velocidad de bits, resolución, idioma, códec, etc. El segmento puede contener cierta duración de datos de medios, además de metadatos para decodificar y presentar el contenido de medios incluido. Un segmento puede identificarse mediante un indicador de recursos uniforme (URI) y puede solicitarse mediante una solicitud GET de HTTP. Un segmento puede definirse como una unidad de datos asociados con un URL de HTTP y, opcionalmente, un rango de bytes especificado mediante una MPD.

La MPD de DASH cumple con el Lenguaje de Marcado Extensible (XML) y, por lo tanto, se especifica a través de elementos y atributo como se define en XML. La MPD puede especificarse usando las siguientes convenciones: Los elementos en un documento XML pueden identificarse mediante una primera letra mayúscula y pueden aparecer en negrita como **Element**. Para expresar que un elemento **Element1** está contenido en otro elemento **Element2**, uno puede escribir **Element2.Element1**. Si el nombre del elemento consiste en dos o más palabras combinadas, se pueden usar las mayúsculas mediales, p.ej., **ImportantElement**. Los elementos pueden estar presentes exactamente una vez, o la ocurrencia mínima y máxima se puede definir mediante <minOccurs> ... <maxOccurs>. Los atributos en un documento XML pueden ser identificados por una primera letra minúscula, así como pueden estar precedidos por un signo '@', p. ej., @attribute. Para señalar un atributo específico @attribute contenido en un elemento **Element**, uno puede escribir **Element**@attribute. Si el nombre de un atributo consiste en dos o más palabras combinadas, pueden usarse mayúsculas mediales después de la primera palabra, p. ej., @veryImportantAttribute. Los atributos pueden tener asignado un estado en el XML como obligatorio (M), opcional (O), opcional con el valor predeterminado (OD) y condicionalmente obligatorio (CM).

En DASH, todos los elementos descriptores están estructurados de la misma manera, es decir, contienen un atributo @schemeldUri que proporciona un URI para identificar el esquema y un atributo opcional @value y un atributo opcional @id. La semántica del elemento es específica para el esquema empleado. El URI que identifica el esquema puede ser un URN o un URL. Algunos descriptores se especifican en MPEG-DASH (ISO/IEC 23009-1), mientras que los descriptores pueden especificarse adicional o alternativamente en otras especificaciones. Cuando se

especifica en especificaciones distintas de MPEG-DASH, la MPD no proporciona ninguna información específica sobre cómo usar elementos descriptores. Le corresponde a la aplicación o especificación que emplea formatos DASH ejemplificar los elementos de descripción con información de esquema apropiada. Las aplicaciones o especificaciones que usan uno de estos elementos definen un Identificador de Esquema en la forma de un URI y el espacio de valor para el elemento cuando se usa ese Identificador de Esquema. El Identificador de Esquema aparece en el atributo @schemeldUri. En el caso de que se requiera un conjunto simple de valores enumerados, se puede definir una cadena de texto para cada valor y esta cadena puede incluirse en el atributo de valor @. Si se requieren datos estructurados, entonces cualquier elemento de extensión o atributo puede definirse en un espacio de nombres separado. El valor @id puede usarse para referirse a un descriptor único o a un grupo de descriptores. En el segundo caso, se puede requerir descriptores con valores idénticos para el atributo @id para que sean sinónimos, es decir, el procesamiento de uno de los descriptores con un valor idéntico para @id es suficiente. Dos elementos del tipo DescriptorType son *equivalentes*, si el nombre del elemento, el valor del @schemeldUri y el atributo del valor @ son equivalentes. Si el @schemeldUri es un URN, entonces la equivalencia puede referirse a una equivalencia léxica como se define en la cláusula 5 del RLC 2141. Si el @schemeldUri es un URL, entonces la equivalencia puede referirse a la igualdad sobre una base de un carácter por carácter como se define en la cláusula 6.2.1 del RFC3986. Si el atributo de valor @ no está presente, la equivalencia puede determinarse mediante la equivalencia para @schemeldUri solamente. Los atributos y elementos en los espacios de nombres de extensión no podrían usarse para determinar equivalencia. El atributo @id puede ignorarse para la determinación de equivalencia.

MPEG-DASH especifica los descriptores EssentialProperty y SupplementalProperty. Para el elemento EssentialProperty el autor de la Presentación Multimedia expresa que el procesamiento exitoso del descriptor es esencial para usar apropiadamente la información en el elemento padre que contiene este descriptor a menos que el elemento comparta el mismo @id con otro elemento EssentialProperty. Si los elementos EssentialProperty comparten el mismo @id, entonces el procesamiento de uno de los elementos EssentialProperty con el mismo valor para @id es suficiente. Se espera que al menos se procese un elemento EssentialProperty de cada valor @id distinto. Si el esquema o el valor para un descriptor EssentialProperty no se reconoce, se espera que el cliente DASH ignore el elemento padre que contiene el descriptor. En una MPD pueden estar presentes múltiples elementos EssentialProperty con el mismo valor para @id y con diferentes valores para @id.

Para el elemento SupplementalProperty el autor de la Presentación Multimedia expresa que el descriptor contiene información complementaria que puede usar el cliente DASH para un procesamiento optimizado. Si el esquema o el valor para un descriptor SupplementalProperty no se reconoce, se espera que el cliente DASH ignore el descriptor. Múltiples elementos SupplementalProperty pueden estar presentes en una MPD.

Un servicio DASH puede proporcionarse como un servicio según demanda o un servicio en vivo. En el primero, la MPD es un estático y todos los Segmentos de una Presentación Multimedia ya están disponibles cuando un proveedor de contenidos publica una MPD. En el segundo, sin embargo, la MPD puede ser estático o dinámico dependiendo del método de construcción de los URL de Segmentos empleado mediante una MPD y los Segmentos pueden crearse continuamente a medida que el contenido se produce y se publica a los clientes DASH mediante un proveedor de contenidos. El método de construcción de los URL de Segmentos puede ser un método de construcción de los URL de Segmento basado en plantillas o el método de generación de listado de Segmentos. En el primero, un cliente DASH puede ser capaz de construir los URL de Segmentos sin actualizar una MPD antes de solicitar un Segmento. En el segundo, un cliente DASH puede necesitar descargar periódicamente las MPD actualizadas para obtener los URL de Segmentos. Para un servicio en vivo, por lo tanto, el método de construcción de los URL de Segmentos basado en plantillas puede ser superior al método de generación de listado de Segmentos.

En el contexto de DASH, pueden usarse las siguientes definiciones: Un componente de contenido multimedia o un componente multimedia puede definirse como un componente continuo del contenido multimedia con un tipo de componente multimedia asignado que puede codificarse individualmente en un flujo multimedia. El contenido multimedia puede definirse como un período de contenido multimedia o una secuencia contigua de períodos de contenido multimedia. El tipo de componente del contenido multimedia puede definirse como un único tipo de contenido multimedia, tal como audio, vídeo, o texto. Un flujo multimedia puede definirse como una versión codificada de un componente de contenido multimedia.

Un Segmento de Inicialización puede definirse como un Segmento que contiene metadatos que son necesarios para presentar los flujos multimedia encapsulados en Segmentos Multimedia. En formatos de segmento basados en ISOBMFF, un Segmento de Inicialización puede comprender la Caja de Película ('caja moov') que puede no incluir metadatos para ninguna muestra, es decir, cualquier metadato para muestras se proporciona en las cajas 'moof'.

Un Segmento Multimedia puede definirse como un Segmento que cumple con el formato de archivo de contenedor y/o el formato o formatos multimedia en uso y que permite la reproducción cuando se combina con cero o más segmentos anteriores, y un Segmento de Inicialización (si lo hubiera). Un Segmento Multimedia puede contener cierta duración de datos multimedia para reproducción a una velocidad normal, tal duración puede denominarse duración del Segmento Multimedia o duración del Segmento. El productor de contenido o proveedor de servicio puede seleccionar la duración del Segmento según las características deseadas del servicio. Por ejemplo, una duración del Segmento relativamente

corta puede usarse en un servicio en vivo para lograr una latencia corta de extremo a extremo. La razón es que la duración del Segmento puede ser un límite menor en la latencia de extremo a extremo percibida por un cliente DASH ya que un Segmento es una unidad discreta de generación de datos multimedia para DASH. La generación de contenido puede realizarse de tal manera que un Segmento completo de datos multimedia esté disponible para un servidor. Además, muchas implementaciones de cliente pueden usar un Segmento como la unidad para solicitudes GET. Por lo tanto, en algunas disposiciones para servicios en vivo, un cliente DASH puede solicitar un segmento solo cuando toda la duración del Segmento Multimedia está disponible así como codificada y encapsulada en un Segmento. Para el servicio según demanda, pueden usarse diferentes estrategias para seleccionar la duración del segmento.

Un Segmento puede dividirse adicionalmente en Subsegmentos, cada uno de los cuales puede contener unidades de acceso completo. Los subsegmentos pueden indexarse por índice de segmento, el cual contiene información para mapear el rango de tiempo de presentación y el rango de bytes para cada subsegmento, y puede usarse para realizar una solicitud GET en HTTP para un subsegmento específico usando una solicitud en HTTP de rangos de bytes. Si se usa una duración del segmento relativamente larga, entonces se pueden usar subsegmentos para mantener el tamaño de las respuestas en HTTP razonables y flexibles para la adaptación de la velocidad de bits. En formatos de segmento basados en ISO/BMFF, un subsegmento puede definirse como un conjunto autónomo de uno o más fragmentos de película consecutivos, donde el conjunto autónomo contiene una o más cajas de Fragmentos de Película con la(s) correspondiente(s) caja(s) de Datos Multimedia, y una Caja de Datos Multimedia que contiene datos referenciados por una Caja de Fragmento de Película debe seguir esa caja de Fragmento de Película y anteceder a la siguiente caja de Fragmento de Película que contiene información sobre la misma pista.

A cada segmento multimedia se le puede asignar una URF única (posiblemente con rango de bytes), un índice, y tiempo y duración de inicio explícitos o implícitos. Cada segmento multimedia puede contener al menos un punto de acceso de flujo, que es un acceso aleatorio o conmutador a punto en el flujo multimedia donde la decodificación puede comenzar usando solo datos de ese punto hacia adelante.

Para permitir la descarga de segmentos en partes múltiples, se puede usar un método para señalar subsegmentos mediante el uso de una caja de índice de segmentos. Esta caja describe subsegmentos y puntos de acceso de flujo en el segmento señalizando sus duraciones y desplazamientos de bytes. El cliente DASH puede usar la información de indexación para solicitar subsegmentos usando solicitudes GET en HTTP parciales. La información de indexación de un segmento puede ponerse en la caja única al comienzo de ese segmento, o extenderse entre muchas cajas de indexación en el segmento. Son posibles diferentes métodos de extensión, tales como jerárquico, cadena de margarita, e híbrido. Esta técnica puede evitar añadir una caja grande al comienzo del segmento y, por lo tanto, puede evitar un posible retardo de descarga inicial.

Los Segmentos (o Subsegmentos respectivamente) pueden definirse para no superponerse de la siguiente manera: Dejar que $T_e(S,i)$ sea el tiempo de presentación más temprano de cualquier unidad de acceso en el flujo i de un Segmento o Subsegmento S , y *dejar que* $T_L(S,i)$ sea el último tiempo de presentación de cualquier unidad de acceso en el flujo i de un Segmento o Subsegmento S . Dos segmentos (Subsegmentos respectivamente), A y B , que pueden o no ser de diferentes Representaciones, pueden definirse para no superponerse, cuando $T_L(A,i) < T_e(B,i)$ para todos los flujos multimedia i en A y B o si $T_L(B,i) < T_e(A,i)$ para todos los flujos i en A y B donde i se refiere al mismo componente multimedia.

MPEG-DASH define los formatos del segmento-contenedor para el Formato de Archivos Multimedia de Base ISO y para los Flujos de Transporte de MPEG-2. Otras especificaciones pueden especificar formatos de segmento basados en otros formatos de contenedores. Por ejemplo, se ha propuesto un formato de segmento basado en el formato contenedor de archivos Matroska y se puede resumir como sigue. Cuando los archivos Matroska se transportan como segmentos DASH o similares, la asociación de unidades DASH y unidades Matroska puede especificarse como sigue. Un subsegmento (de DASH) puede definirse como una o más Agrupaciones consecutivas de contenido encapsulado en Matroska. Puede requerirse un Segmento de Inicialización de DASH para comprender el encabezado EBML, la cabecera de Segmento (de Matroska), la Información de Segmento (de Matroska) y Pistas, y opcionalmente puede comprender otros elementos de nivel 1 y relleno. Un Índice de Segmento de DASH puede comprender un Elemento de Localización de Matroska.

DASH especifica diferentes cronogramas que incluyen un cronograma de Presentación Multimedia y tiempos de disponibilidad de Segmentos. El primero indica el tiempo de presentación de una unidad de acceso con un contenido multimedia que se mapea con el cronograma de presentación común global. El cronograma de Presentación Multimedia puede permitir que DASH sincronice sin problemas diferentes componentes multimedia que se codifican con diferentes técnicas de codificación y comparten un cronograma común. El segundo indica un tiempo de reloj de pared y se usa para indicar a los clientes el tiempo de disponibilidad de los Segmentos que pueden ser identificados por los URL en HTTP. Un cliente DASH puede ser capaz de identificar un tiempo de disponibilidad de un cierto Segmento al comparar el tiempo del reloj con el tiempo de disponibilidad del Segmento asignado a ese Segmento. El tiempo de disponibilidad de Segmentos puede usarse en la entrega en vivo de Segmentos multimedia, denominado servicio en vivo. Para el servicio en vivo, el tiempo de disponibilidad de Segmentos es diferente al Segmento a Segmento y un tiempo de disponibilidad de un Segmento determinado puede depender de la posición del Segmento en el cronograma de Presentación Multimedia. Para el servicio según demanda, el tiempo de disponibilidad de Segmentos puede ser el mismo para todos los Segmentos.

DASH respalda la adaptación de velocidad al solicitar dinámicamente segmentos y/o subsegmentos multimedia de diferentes representaciones dentro de un conjunto de adaptación para coincidir con un ancho de banda de red variable. Cuando un cliente DASH conmuta la Representación arriba/abajo, puede ser necesario tener en cuenta las dependencias de codificación dentro de la Representación. En la decodificación multimedia, un interruptor de Representación puede solo ocurrir en un punto de acceso aleatorio (RAP), que puede usarse en técnicas de codificación de vídeo tales como H.264/AVC. Para evitar la solicitud y la transmisión de datos multimedia que no se decodificarán, los RAP pueden alinearse al comienzo de Segmentos y/o Subsegmentos Multimedia, y la MPD y/o la caja de índice de segmentos pueden usarse para indicar la alineación de los RAP al comienzo de los Segmentos y/o Subsegmentos Multimedia. Por consiguiente, los clientes DASH pueden ser capaces de concluir cuáles Segmentos y/o Subsegmentos solicitar, de tal manera que cuando se realiza la conmutación de representaciones, el primer Segmento y/o Subsegmento de una Representación de destino comienza con un RAP y los Segmentos y/o Subsegmentos de la Representación de origen y destino están alineados (por tiempos). En DASH, se introduce un concepto más general denominado Punto de Acceso de Flujo (SAP) para proporcionar una solución independiente del códec para acceder a una Representación y conmutación entre Representaciones. En DASH, un SAP se especifica como una posición en una Representación que permite que la reproducción de un flujo multimedia se inicie usando solo la información contenida en los datos de Representación comenzando desde esa posición en adelante (antecedida por la inicialización de datos en el segmento de inicialización, si lo hubiera). Por lo tanto, la conmutación de Representaciones se puede realizar en SAP.

Se han especificado varios tipos de SAP, incluido lo siguiente. El SAP Tipo 1 corresponde a lo que se conoce en algunos esquemas de codificación como un “punto de acceso aleatorio de GOP cerrado” (en el que todas las imágenes, en orden de decodificación, pueden decodificarse correctamente, dando como resultado una secuencia de tiempo continua de imágenes decodificadas correctamente sin vacíos) y, además, la primera imagen en orden de decodificación también es la primera imagen en orden de presentación. El SAP Tipo 2 corresponde a lo que se conoce en algunos esquemas de codificación como un “punto de acceso aleatorio de GOP cerrado” (en el que todas las imágenes, en orden de decodificación, pueden decodificarse correctamente, dando como resultado una secuencia de tiempo continua de imágenes decodificadas correctamente sin vacíos), para lo cual la primera imagen en orden de decodificación puede no ser la primera imagen en orden de presentación. El SAP Tipo 3 corresponde a lo que se conoce en algunos esquemas de codificación como un “punto de acceso aleatorio de GOP abierto”, en el que puede haber algunas imágenes en orden de decodificación que no pueden decodificarse correctamente y tienen tiempos de presentación menores que la imagen intracodificada asociada al SAP.

Un proveedor de contenido puede crear Segmentos y Subsegmentos de múltiples Representaciones de una manera que pueda hacer la conmutación más sencilla. En un caso sencillo, cada segmento y subsegmento comienza con un SAP y los límites de segmento y subsegmento están alineados a través de la representación de un conjunto de adaptación. En un caso así, un cliente DASH puede ser capaz de conmutar Representaciones sin desviación de error al solicitar Segmentos o Subsegmentos de una Representación original a una nueva Representación. En DASH, las restricciones para desarrollar Segmento y Subsegmento se especifican en la MPD y en el Índice de Segmentos para facilitar a un cliente DASH que conmute Representaciones sin introducir una desviación de error. Uno de los usos de perfil especificado en DASH es proporcionar diferentes niveles de restricciones para desarrollar Segmento y Subsegmento, etc.

Como se describió anteriormente, el cliente o jugador puede solicitar que se transmitan Segmentos o Subsegmentos desde diferentes representaciones de manera similar a como puedan determinarse las capas y/o subcapas de una secuencia de bits de vídeo escalable. Los términos conmutación descendente de representaciones o conmutación descendente de secuencia de bits pueden referirse a la solicitud o transmisión de una representación de velocidad de bits más baja que la solicitada o transmitida (respectivamente) previamente. Los términos conmutación ascendente de representaciones o conmutación ascendente de secuencia de bits pueden referirse a la solicitud o transmisión de una representación de velocidad de bits más alta que la solicitada o transmitida (respectivamente) previamente. Los términos conmutación de representaciones o conmutación de secuencia de bits pueden referirse colectivamente a la conmutación ascendente y descendente de representaciones o flujos de bits.

Los sistemas de transmisión continua similares a MPEG-DASH incluyen, por ejemplo, Transmisión continua en vivo en HTTP (conocido como HLS), especificado en la transmisión continua en vivo en http 13 borrador de Pantos de Borrador Internet del IETF (y otras versiones del mismo Borrador Internet). Como un formato de manifiesto correspondiente a la MPD, HLS usa un formato M3U extendido. M3U es un formato de archivo para listas de reproducción multimedia, desarrolladas originalmente para archivos de audio. Una lista de reproducción M3U es un archivo de texto que consiste en líneas individuales, y cada línea es un URI, en blanco, o comienza con el carácter '#' que indica una etiqueta o un comentario. Una línea de URI identifica un segmento multimedia o un archivo de lista de reproducción. Las etiquetas comienzan con #EXT. La especificación HLS especifica varias etiquetas, que pueden considerarse como pares de valores clave. La parte de valor de las etiquetas puede comprender una lista de atributos, que es una lista separada por comas de pares de valores de atributo, donde se puede considerar que un par de valores-atributo tiene la sintaxis AttributeName=AttributeValue. Por lo tanto, las etiquetas de los archivos M3U8 de HLS pueden considerarse similares a los Elementos en MPD o XML, y los atributos de los archivos M3U8

de HLS pueden considerarse similares a los atributos en MPD o XML. Los segmentos multimedia en HLS se formatean según el Flujo de Transporte MPEG-2 y contienen un Programa MPEG-2 único. Se recomienda que cada segmento multimedia comience con una Tabla de Asociación de Programas (PAT) y una Tabla de Mapas de Programas (PMT).

La conmutación entre diferentes secuencias de bits del mismo contenido puede usarse para la adaptación de la velocidad de bits. La conmutación a una secuencia de bits diferente puede realizarse, p. ej., en cualquier intraimagen que comience un GOP cerrado (tal como una imagen IDR).

Para responder a una necesidad de ajustar la velocidad de bits más rápida y evitar la penalización por compresión de intraimágenes frecuentes, una opción posible sería comenzar la conmutación del flujo desde no intraimágenes. En una realización, los marcos S que son marcos intercodificados se usan solo cuando se conmutan de un primer flujo a un segundo flujo. Tales marcos S pueden codificarse con un pequeño paso de cuantificación y hacer que el marco S decodificado sea cercano pero típicamente no idéntico a la imagen decodificada correspondiente del segundo flujo. H.264/AVC incluye la característica conocida como imágenes SI/SP. Los marcos SP y Si pueden usarse de manera similar a los marcos S, pero pueden proporcionar una imagen decodificada idéntica después de la conmutación en comparación con la decodificación del flujo desde el principio. Las imágenes decodificadas idénticas pueden obtenerse con el coste de los pasos de transformación y cuantificación adicionales en el proceso de decodificación para imágenes SI/SP en los flujos primarios y en las imágenes SI/SP usadas solo para conmutar. Sin embargo, la característica SI/SP no se incluye en el valor inicial ni en el perfil alto y, por lo tanto, no se usa comúnmente.

La Figura 7 muestra un ejemplo de cómo pueden usarse imágenes SP para conmutar entre dos flujos de bits diferentes. En el formato de archivo, las imágenes de conmutación (SP3 en el ejemplo de la Figura 7) se almacenan en pistas de imágenes de conmutación, que son pistas separadas de la pista desde la que se está conmutando y la pista a la que se conmuta. Las pistas de imágenes de conmutación pueden identificarse por la existencia de una referencia de pista requerida específica en esa pista. Una imagen de conmutación es una alternativa a la muestra en la pista de destino que tiene exactamente el mismo tiempo de decodificación. Si todas las imágenes de conmutación son imágenes SI, entonces puede no ser necesaria información adicional.

Si cualquiera de las imágenes de la pista de conmutación son imágenes SP, entonces pueden necesitarse dos piezas de información adicionales. Primero, la pista de origen desde la que se está conmutando puede identificarse usando una referencia de pista (la pista de origen puede ser la misma pista que la pista de destino). En segundo lugar, puede ser necesaria la dependencia de la imagen de conmutación en las muestras de la pista de origen, de tal manera que una imagen de conmutación solo se use cuando las imágenes de las que depende se hayan suministrado al decodificador.

Esta dependencia puede representarse por medio de una tabla de muestras adicional opcional. Hay una entrada por muestra en la pista de conmutación. Cada entrada registra los números de muestras relativos en la pista de origen de la que depende la imagen de conmutación. Si esta matriz está vacía para una muestra dada, entonces esa muestra de conmutación contiene una imagen SI. Si la caja de dependencia no está presente, entonces solo los marcos SI deberán estar presentes en la pista.

Una muestra de conmutación puede tener múltiples representaciones codificadas con diferentes dependencias. Para vídeos AVC, las múltiples representaciones de una muestra de conmutación pueden almacenarse en diferentes pistas de conmutación (es decir, unidades de acceso). Por ejemplo, una pista de conmutación podría contener una representación de imagen SP dependiente de algunas muestras anteriores, usadas para la conmutación del flujo, mientras que otra pista de conmutación puede contener otra representación como una imagen Si, usada para acceso aleatorio.

Un Cambio de Resolución Adaptativa (ARC) se refiere a cambiar dinámicamente la resolución dentro de la secuencia de vídeo, por ejemplo, en casos de uso en videoconferencias. El cambio de resolución adaptativa puede usarse p. ej., para una mejor adaptación de la red y una resistencia al error. Para una mejor adaptación a los requisitos de red cambiantes para diferente contenido, puede desearse poder cambiar la resolución temporal/espacial además de la calidad. El Cambio de Resolución Adaptativa también puede permitir un inicio rápido, en donde el tiempo de arranque de una sesión puede ser capaz de aumentarse enviando primero un marco de baja resolución y después aumentando la resolución. El Cambio de Resolución Adaptativa puede usarse además para componer una conferencia. Por ejemplo, cuando una persona comienza a hablar, su resolución correspondiente puede aumentarse. Hacer esto con un marco IDR puede provocar un "incidencia" en la calidad ya que los marcos IDR deben codificarse a una calidad relativamente baja de tal manera que el retardo no aumente significativamente.

Se puede usar una señalización de nivel de secuencias para indicarle al decodificador si hay un cambio de resolución en la secuencia de bits. Esto puede realizarse, p. ej., usando una bandera `single_layer_for_non_irap_flag`. Se puede usar `single_layer_for_non_irap_flag` para indicar que una secuencia de vídeo codificada está obligada a realizar la operación de cambio de resolución adaptativa. La bandera `single_capl_for_non_irap_flag` indica específicamente que, excepto para imágenes de conmutación, cada AU en

la secuencia contiene una imagen única de una capa única (que puede o no ser imagen BL); y que las unidades de acceso donde ocurre la conmutación incluyen imágenes provenientes de dos capas, la imagen EL en tal unidad de acceso es una imagen IRAP, y puede usarse una predicción entre capas para la imagen EL IRAP.

- 5 Indicar que la bandera `single_layer_for_non_irap_flag` en VPS puede permitir que el decodificador sepa que no se usa escalabilidad excepto para lograr cambio de resolución, de tal manera que los recursos del decodificador pueden asignarse en consecuencia al iniciarse una sesión.

10 `single_layer_for_non_irap_flag` puede incluirse en la sintaxis VUI VPS, ya que puede no provocar ningún cambio en el proceso de decodificación normativo.

La semántica de `single_layer_for_non_irap_flag` puede especificarse de la siguiente manera:

15 `single_layer_for_non_irap_flag` igual a 1 indica lo siguiente:

- Si `vps_base_layer_internal_flag` es igual a 1, `single_layer_for_non_irap_flag` igual a 1 indica que uno de los siguientes es verdadero para cada unidad de acceso para la que este VPS es el VPS activo:

- Todas las unidades NAL VCL de una unidad de acceso tienen el mismo valor `nuh_layer_id`,
- Dos valores `nuh_layer_id` son usados por las unidades NAL VCL de una unidad de acceso y la imagen con el mayor valor `nuh_layer_id` es una imagen IRAP.

- A menos que se indique lo contrario (`vps_base_layer_internal_flag` es igual a 0), `single_layer_for_non_irap_flag` igual a 1 indica que cualquiera de los siguientes es verdadero para cada unidad de acceso para la que este VPS es el VPS activo:

- La imagen decodificada con `nuh_layer_id` igual a 0 no se proporciona para la unidad de acceso por medios externos y la unidad de acceso contiene una imagen codificada,
- La imagen decodificada con `nuh_layer_id` igual a 0 no se proporciona para la unidad de acceso por medios externos, la unidad de acceso contiene dos imágenes codificadas, y la imagen con el mayor valor `nuh_layer_id` es una imagen IRAP,
- La imagen decodificada con `nuh_layer_id` igual a 0 se proporciona para una unidad de acceso por medios externos y la unidad de acceso contiene una imagen codificada que es una imagen IRAP.

40 `single_layer_for_non_irap_flag` igual a 0 indica que las restricciones anteriores pueden o no aplicarse. Cuando no está presente, se infiere que el valor de `single_layer_for_non_irap_flag` es igual a 0.

`single_layer_for_non_irap_flag` puede ir acompañado de `higher_layer_irap_skip_flag`, para la que la semántica puede especificarse de la siguiente manera:

45 `higher_layer_irap_skip_flag` igual a 1 indica que cada imagen IRAP `currIrapPic` está restringida como se especifica a continuación. `currIrapPic` se obtiene como sigue para cada unidad de acceso `currAu` para la que este VPS es el VPS activo:

- Si `vps_base_layer_internal_flag` es igual a 1, `currAu` contiene dos imágenes codificadas, y la imagen con el valor `nuh_layer_id` mayor es una imagen IRAP, permite que `currIrapPic` sea esa imagen IRAP.

- A menos que se indique lo contrario, si `vps_base_layer_internal_flag` es igual a 0, no se proporciona una imagen decodificada con `nuh_layer_id` igual a 0 para `currAu` por medios externos, `currAu` contiene dos imágenes codificadas, y la imagen con el mayor valor `nuh_layer_id` es una imagen IRAP, permite que `currIrapPic` sea esa imagen IRAP.

- A menos que se especifique lo contrario, si `vps_base_layer_internal_flag` es igual a 0, la imagen decodificada con `nuh_layer_id` igual a 0 se proporciona para `currAu` por medios externos y la unidad de acceso contiene una imagen codificada que es una imagen IRAP, permite que `currIrapPic` sea esa imagen IRAP.

- A menos que se especifique lo contrario, `currIrapPic` no se obtiene para `currAu`.

- Las siguientes restricciones pueden aplicarse para cada imagen `currIrapPic`:

- Para todos los fragmentos de la imagen IRAP::

- `slice_type` será igual a P.

- slice_sao_luma_flag y slice_sao_chroma_flag serán igual a 0.
- five_minus_max_num_merge_cand será igual a 4.
- weighted_pred_flag será igual a 0 en el PPS que se menciona por los fragmentos,
 - Para todas las unidades de codificación de la imagen IRAP:
- cu_skip_flag[i] [j] será igual a 1.

Cuando single_layer_for_non_irap_flag es igual a 0, higher_layer_irap_skip_flag debería ser igual a 0. Cuando higher_layer_irap_skip_flag no está presente se puede inferir que es igual a 0.

Cuando vps_base_layer_internal_flag es igual a 1, un codificador puede establecer single_layer_for_non_irap_flag igual a 1 como una indicación a un decodificador de que como máximo dos imágenes están presentes en cualquier unidad de acceso y siempre que haya dos imágenes en la misma unidad de acceso, aquella con el valor más alto de nuh_layer_id es una imagen IRAP. El codificador puede establecer adicionalmente higher_layer_irap_skip_flag igual a 1 como una indicación a un decodificador de que siempre que haya dos imágenes en la misma unidad de acceso, aquella con el valor más alto de nuh_layer_id es una imagen IRAP para la que las muestras decodificadas pueden obtenerse aplicando el proceso de obtención de imágenes de referencia entre capas con la otra imagen con el valor más bajo de nuh_layer_id como entrada.

Cuando single_layer_for_non_irap_flag es igual a 1 y higher_layer_irap_skip_flag es igual a 1, la imagen con el valor más alto de nuh_layer_id en una unidad de acceso que contiene dos imágenes puede denominarse imagen IRAP codificada por omisión o una imagen de omisión.

Las imágenes por omisión pueden ser pequeñas, ya que el modo de fusión puede usarse para codificar el movimiento entre capas para cada unidad de predicción, las unidades de árbol de codificación no necesitan dividirse en las unidades de codificación, y solo podría requerirse una bandera codificada por CABAC, cu_skip_flag, para omitir la codificación residual para cada unidad de codificación.

Capas independientes sin base y capacidad independiente de decodificación de capa sin base (INBLD)

Una capa independiente puede definirse como una capa que no tiene capas de referencia directas. En otras palabras, no se predice una capa independiente mediante el uso de predicción entre capas.

Una capa sin base puede definirse como cualquier otra capa que la capa base, y la capa base puede definirse como la capa más baja en la secuencia de bits.

Una capa independiente sin base puede definirse como una capa que es una capa independiente y una capa sin base.

La capacidad independiente de decodificación de capas sin base (INBLD) se relaciona con la capacidad de decodificación de uno o más de los perfiles de una sola capa, tal como el perfil principal. Cuando se expresan las capacidades de un decodificador para uno o más perfiles de una sola capa, también se debería expresar si la capacidad independiente de decodificación de capa sin base es compatible con esos perfiles.

La capacidad independiente de decodificación de capas sin base, cuando es compatible, puede indicar la capacidad de un decodificador para decodificar una capa independiente sin base que se indica, p. ej., en los VPS y SPS activos para conformar un perfil de una sola capa y es la capa con el valor nuh_layer_id más bajo en un conjunto de capas adicionales.

Cuando la estructura profile_tier_level() de HEVC se usa para indicar la capacidad de un decodificador en sistemas, la capacidad independiente de decodificación de capas sin base puede indicarse al establecer la general_inbld_flag igual a 1 en la estructura sintáctica profile_tier_level() usada para expresar el perfil, la clasificación, y el nivel a los que se ajusta el decodificador.

general_inbld_flag puede establecerse igual a 1 en las estructuras sintácticas profile_tier_level() de HEVC en las que se indica un perfil de una sola capa y que se especifican en el VPS que son aplicables para una capa sin base o se incluyen en un SPS activado para una capa independiente sin base.

Los decodificadores HEVC que tienen la capacidad independiente de decodificación de capa sin base y que se ajustan a un perfil específico de una sola capa a un nivel específico de un nivel específico pueden ser capaces de decodificar cualquier capa independiente sin base o una representación de subcapa con TemporalId igual a i de la capa independiente sin base para la que todas las siguientes condiciones se aplican para cada VPS:

– Hay una OLS que consiste en la capa independiente sin base y para la que la estructura sintáctica asociada `profile_tier_level( )` está restringida de la siguiente manera:

- 5 ○ `ptlStruct` indica que la capa independiente sin base o la representación de subcapa se ajusta a un perfil especificado en el Anexo A.
- `ptlStruct` indica que la capa independiente sin base o la representación de subcapa se ajusta a un nivel menor que o igual al nivel especificado,
- 10 ○ `ptlStruct` indica que la capa independiente sin base o la representación de subcapa se ajusta a un nivel menor que o igual a la clasificación especificada,
- `general_inbld_flag` o `sub_layer_inbld_flag[ i ]` en `ptlStruct` es igual a 1.

15 Una norma de codificación puede incluir un proceso de reescritura de la capa independiente sin base, y tal se especifica, por ejemplo, en HEVC. Dicho proceso de reescritura puede tomar una secuencia de bits que incluye una capa independiente sin base como entrada y convertir la capa independiente sin base para convertirse en la capa base de una secuencia de bits de salida del proceso. Dejar que una secuencia de bits obtenido de `outBitstream` sea una secuencia de bits que se obtiene al invocar el proceso de reescritura de la capa independiente sin base. Los decodificadores con capacidad independiente de decodificación de capa sin base pueden aplicar el proceso de reescritura de la capa independiente sin base para obtener `outBitstream` y después aplicar un proceso de decodificación para un perfil de una sola capa especificado con `outBitstream` como entrada.

20 Las imágenes de punto de acceso aleatorio (RAP) se usan en transmisión continua para proporcionar funcionalidad tal como búsqueda y avance rápido. En DASH, las imágenes RAP también se usan para permitir la conmutación de representaciones, lo que puede llevarse a cabo como una respuesta a la lógica de adaptación de la velocidad del jugador para evitar el bajo flujo del regulador de recepción y usar el rendimiento de la red tan eficazmente como sea posible.

25 Para respaldar la conmutación del cliente entre diferentes calidades y resoluciones durante la sesión de transmisión continua de representaciones DASH, las imágenes de punto de acceso aleatorio pueden codificarse en los límites del segmento. Convencionalmente, solo se han usado imágenes RAP instantáneas, como la imagen de renovación de decodificación instantánea (IDR), que inicia una estructura de predicción denominada grupo cerrado de imágenes (GOP), en límites de segmento de representaciones DASH. El uso de intraimágenes que comienzan los GOP abiertos, es decir, las imágenes CRA en H.265/HEVC, se ha mejorado en H.265/HEVC cuando se compara con las normas más antiguas, ya que un proceso de decodificación que comienza desde una imagen CRA se ha especificado normalmente. Cuando la decodificación se inicia a partir de una imagen CRA, algunas imágenes, denominadas de acceso aleatorio omitido (RASL), que siguen a la imagen CRA en el orden de decodificación pero que anteceden a la imagen CRA en el orden de salida pueden no ser decodificables. En consecuencia, si se usaron GOP abiertos en límites de segmento en DASH, la conmutación de representaciones daría como resultado la incapacidad de decodificar las imágenes RASL y, por lo tanto, un error de velocidad de imágenes en la reproducción. Por ejemplo, si se utilizara una jerarquía de predicción de 8 imágenes y la velocidad de las imágenes fuera de 25 Hz, el vídeo se congelaría durante aproximadamente un tercio de segundo.

30 La conmutación de representaciones sin interrupciones puede habilitarse cuando las representaciones usan estructuras de GOP abierto y comparten la misma resolución y otras características, es decir, cuando una imagen decodificada de la representación de origen puede usarse como tal como una imagen de referencia para predecir imágenes de una representación objetivo. Sin embargo, las representaciones pueden no compartir las mismas características, p. ej., pueden ser de diferente resolución espacial, en donde la conmutación de representaciones sin interrupciones puede necesitar algunas consideraciones adicionales.

A continuación, se describen con más detalle algunos desafíos con respecto a la conmutación de representaciones sin interrupciones.

35 Un manejo adecuado de las imágenes principales puede no ser obvio. Como ilustración de los desafíos de la manipulación de las imágenes principales, se describe a continuación un ejemplo con referencia a las Figuras 8a— 8e. En este ejemplo, el orden de salida de las imágenes pasa de izquierda a derecha. Se observa que el orden de decodificación difiere del orden de salida. En el ejemplo de la Figura 8a hay dos flujos de bits, BS1 y BS2. Las imágenes se indican con rectángulos y el tipo de imagen se indica dentro del rectángulo:

40 IDR, intra (I), inter (P) o biprevista (B). Un valor de `temporal_id` mayor que 0 se indica como un número después del tipo de imagen. Las flechas indican la relación de interpredicción: la fuente de una flecha se usa como una imagen de referencia para la imagen a la que apunta la flecha.

En el ejemplo, se realiza un conmutador desde una secuencia de bits BS1 a un segundo secuencia de bits BS2 en la imagen I. Se debería prestar atención al manejo de las imágenes principales (B2, B1, B2) inmediatamente antes de la imagen I en el orden de salida.

- 5 La omisión de la decodificación de imágenes principales (“no decodificables”) puede provocar un vacío en la reproducción, que no se prefiere. La Figura 8b ilustra este enfoque.

Un enfoque podría ser recibir y decodificar la imagen I de ambos flujos de bits. Después, se puede realizar una elección entre dos opciones. Según la primera opción, las imágenes principales se reciben y se decodifican desde la primera (conmutación de) secuencia de bits BS1. Se observa que generalmente las imágenes principales de BS2 también se reciben, ya que siguen la imagen I en orden de decodificación. Otra opción es recibir y decodificar las imágenes principales del segundo (conmutación a) secuencia de bits BS2. Las imágenes principales del primer secuencia de bits BS1 no necesitan recibirse o decodificarse. Se observa que las imágenes principales del segundo secuencia de bits BS2 pueden no ser perfectamente reconstruidas como algunas de sus imágenes de referencia en la decodificación que se originan a partir del primer secuencia de bits BS1, mientras que las imágenes principales del segundo secuencia de bits BS2 se codificaron usando imágenes de referencia provenientes de BS2. Además, esta operación supone que las imágenes decodificadas de BS1 son tales que pueden usarse como imágenes de referencia para decodificar imágenes de BS2, es decir, generalmente que al menos la resolución espacial, el formato de croma, y profundidad de bits de BS1 y BS2 son los mismos. Cuando las imágenes decodificadas de BS1 son tales que no son adecuadas para usarse como imágenes de referencia para la decodificación de imágenes de BS2, no es posible usar este enfoque.

En ambas opciones mencionadas anteriormente, se reciben y se decodifican dos imágenes I, lo que consume ancho de banda de transmisión y puede provocar una pequeña pausa en la reproducción debido a la decodificación más lenta que en tiempo real. La Figura 8c ilustra este enfoque. Además, los servicios DASH y similares funcionan típicamente sobre la base de Segmento o Subsegmento, donde se solicita y recibe una secuencia de imágenes en lugar de imágenes individuales. Por lo tanto, la recepción de dos imágenes I provenientes de diferentes representaciones puede no ser práctica.

En general, el patrón GOP de los flujos de bits no necesita ser idéntico. Por consiguiente, puede no ser conocido si las imágenes decodificadas provenientes de una secuencia de bits pueden usarse como imágenes de referencia para el otra secuencia de bits. Por lo tanto, las imágenes principales del primer secuencia de bits pueden decodificarse pero no del segundo secuencia de bits. La Figura 8d proporciona un ejemplo de tal situación. Las imágenes principales B1 del segundo secuencia de bits BS2 no pueden decodificarse ya que no hay imagen de referencia en el primer secuencia de bits BS1 equivalente a la imagen P del segundo secuencia de bits BS2.

Una estrategia de decodificación ejemplar según una realización se describe a continuación con referencia a la Figura 8e. En este ejemplo, las estructuras GOP de dos flujos de bits son las mismas. Este enfoque de decodificación puede requerir la decodificación de solo una de las intrainágenes en el punto de conmutación, por consiguiente, posiblemente mejorando el funcionamiento en tiempo real sin pausas. Las imágenes principales del segundo secuencia de bits BS2 pueden no ser perfectamente reconstruidas, pero tal degradación temporal de calidad de imagen generalmente no puede percibirse en absoluto o no se considera molesta. La imagen I y las imágenes principales de BS1 no necesitan transmitirse, lo que ahorra ancho de banda de transmisión.

Para resumir, puede no determinarse fácilmente en el receptor si la intrainimagen y las imágenes principales del primer secuencia de bits BS1 deberían recibirse y decodificarse, o si las imágenes principales del segundo secuencia de bits BS2 pueden decodificarse usando imágenes de referencia del primer secuencia de bits BS1 cuando sea necesario. Además, es preferible realizar la conmutación de flujos de bits con base en Segmento o Subsegmento y evitar solicitudes individuales de transmisión de una imagen particular o imágenes particulares.

Como alternativa al uso de los GOP abiertos en conmutación de representaciones podrían usarse los GOP cerrados, en donde se puede lograr una reproducción sin interrupciones, pero puede tener una eficacia de compresión menor en comparación con el uso de los GOP abiertos.

A continuación se presentan varias realizaciones para permitir la conmutación de representaciones en DASH o una transmisión continua adaptativa similar de tal manera que se mantenga una velocidad de imagen estable también durante la conmutación de representaciones y se usa una estructura de predicción de GOP abierta para obtener una mejor eficacia de compresión que cuando se usa una estructura de predicción de GOP cerrada.

En la siguiente se describe una codificación y preparación de segmentos según una realización con más detalle con referencia a las Figuras 9a y 10. Esta realización puede denominarse un cambio de resolución adaptativa redundante (RARC). Un codificador puede recibir o, a menos que se especifique lo contrario, obtener una primera presentación de vídeo y una segunda presentación de vídeo o una o más partes de ellas (bloques 100 y 102 en la Figura 10). La primera presentación de vídeo y la segunda presentación de vídeo pueden, pero no es necesario, recibirse o, a menos que se especifique lo contrario, obtenerse en un formato sin comprimir en los bloques 100 y 102. En algunos casos,

puede obtenerse una única presentación de vídeo o, a menos que se especifique lo contrario, recibirse, y la primera presentación de vídeo y/o la segunda presentación de vídeo pueden obtenerse de la única presentación de vídeo por remuestreo, escala de valor de muestras y/u otro procesamiento. El codificador puede codificar el primer secuencia de bits (Representación #1) 901 (bloque 104), que en este ejemplo es una representación de menor resolución. El codificador puede codificar una o más imágenes IDR/BLA 903, imágenes principales 904 (si las hubiera), tales como imágenes RADL, e imágenes 905 más recientes en la secuencia de bits para la Representación #1. Respectivamente, el codificador también puede codificar el segundo secuencia de bits (Representación #2) 902 (bloque 106), que en este ejemplo es una representación de mayor resolución. El codificador puede codificar una o más imágenes CRA 906, imágenes principales 907 (si las hubiera), tales como imágenes RASL, e imágenes 908 más recientes en la secuencia de bits para la Representación #2. El codificador u otra entidad, tal como un creador de MPD o un encapsulador de archivos, puede codificar información para decodificar una parte de Representación #2 usando Representaciones #1 (bloque 108).

En el cambio de resolución adaptativa redundante, la secuencia de bits de baja resolución puede codificarse convencionalmente, p. ej., como una secuencia de bits H.265/HEVC de una sola capa, mientras que la secuencia de bits de alta resolución puede, a menos que se especifique lo contrario, ser una secuencia de bits de capa única convencional, pero el identificador de capas (el elemento sintáctico `nuh_layer_id`) puede establecerse igual a 1 (o algún otro valor distinto de cero). Además, pueden crearse imágenes IRAP adicionales codificadas por omisión de la capa de mejora (también denominadas imágenes de omisión a continuación) para cada imagen de referencia de imágenes RASL de alta resolución que anteceden a la imagen CRA asociada en orden de decodificación. Esto puede hacerse solo para aquellas imágenes CRA que comienzan un Segmento o un Subsegmento. Las imágenes de omisión pertenecen conceptualmente a la capa de mejora (con `nuh_layer_id` igual a 1) de la secuencia de bits de baja resolución, y pueden encapsularse en los segmentos de la representación de baja resolución.

El funcionamiento del método de cambio de resolución adaptativa redundante durante la conmutación ascendente se ilustra en la Figura 9b y en el diagrama de flujo de la Figura 11, según una realización. Al obtener la representación de baja resolución 901 (bloques 120 y 122 en la Figura 11), el jugador también puede obtener las imágenes de omisión 910 en cada segmento. El jugador u otra entidad, tal como un analizador MPD o analizador de archivos, puede decodificar o analizar información para decodificar una parte de la segunda presentación de vídeo codificada al usar la primera presentación de vídeo codificada (bloque 128). La información puede indicar que la primera presentación de vídeo codificada incluye imágenes de omisión 910 que pueden usarse como referencia para la predicción en la decodificación de la parte de la segunda presentación de vídeo codificada. Estas imágenes de omisión 910 pueden usarse solo cuando se conmutan a la representación de alta resolución 902 (bloques 124 y 126). El jugador usa el proceso de decodificación SHVC (o similar) durante la conmutación ascendente y, por lo tanto, puede decodificar las imágenes de omisión 910. El siguiente segmento recibido es de la representación de alta resolución 902 (bloque 130). Las imágenes RSL 907 de la primera imagen CRA 906 de la representación de alta resolución también usan imágenes del segmento anterior como referencia. Sin embargo, el decodificador puede usar ahora las imágenes de omisión 910 decodificadas en su lugar. A medida que las imágenes de omisión se incluyen en la secuencia de bits para ser decodificadas, el decodificador puede seguir el proceso de decodificación SHVC (o similar) cuando se decodifica dicho siguiente segmento recibido (bloque 130) y puede que no sea necesario crear una imagen o imágenes de referencia sobremuestradas sin usar procesos no normativos.

Según una realización, la conmutación entre representaciones de diferente resolución espacial, profundidad de bits, gama de colores y/o formato de croma puede habilitarse a través del remuestreo y/u otro procesamiento.

Las características de las secuencias de (sub)segmentos no superpuestos pueden indicarse separadamente de las características de representaciones individuales o conjuntos de adaptación. En una realización que usa cambio de resolución adaptativa redundante, las secuencias de (sub)segmentos no superpuestos pueden indicar que son flujos de bits de vídeo escalables que están restringidos a usar solo la característica de cambio de resolución adaptativa. En esta realización, puede usarse un proceso de decodificación de vídeo escalable convencional, por lo que se puede habilitar la conmutación entre representaciones heterogéneas que usan la estructura de predicción de GOP abierto sin cambios en los procesos de decodificación estandarizados.

Un proceso de decodificación común para las extensiones de múltiples capas de HEVC, especificado en la cláusula F.8 de la versión 2 de HEVC (y sus subcláusulas), obtiene las siguientes dos variables a través de medios externos: `TargetOlsIdx`, que especifica el índice del conjunto de capas de salida que usa el decodificador; y `HighestTid`, que especifica la subcapa temporal más alta que se decodificará.

El decodificador puede no ser capaz de funcionar sin establecer estas variables. Por lo tanto, a continuación se proporcionan realizaciones para indicar estas variables y asociarlas a combinaciones de tripletes de perfil-clasificación-nivel.

En una realización, que puede aplicarse junto con o independientemente de otras realizaciones, el parámetro de códecs y/o cualquier otro parámetro de tipo MIME puede proporcionar información de al menos uno del conjunto de capas de salida y/o la subcapa más alta que se aplica a la lista asociada de valores perfil-clasificación-nivel.

Por ejemplo, la siguiente sintaxis y semántica pueden usarse para cada *ListItem* incluido en el parámetro de códecs (donde el parámetro de códecs es una lista separada por comas de *ListItems*, como se ha descrito anteriormente):

5 Cada *ListItem* puede tener la siguiente estructura:

SampleEntryType1.ProfileTierLevel1(.SampleEntryTypeN.ProfileTierLevelN)(#OutOpPoint)?* donde un asterisco (**) indica una repetición de 0 o más veces, (?) indica una ocurrencia de cero o una vez, es un carácter separador, y '#' es otro carácter separador dedicado a separar una subcadena de puntos de operación de salida *OutOpPoint* a partir de una lista de subcadenas de perfil-clasificación-nivel separados por comas. Deberá entenderse que el carácter separador puede ser igualmente cualquier otro carácter separador, o pueden usarse varios caracteres separadores distintos, posiblemente para diferente ubicación o propósito. Por ejemplo, se puede usar un carácter separador '!' en lugar de '.' delante de *SampleEntryTypeN*. Asimismo, debe entenderse que el carácter separador '#' puede ser igualmente cualquier otro carácter separador.

La sintaxis anterior para un *ListItem* se puede entender que comprende una o más subcadenas de perfil-clasificación-nivel, separadas por el carácter separador y cero o una subcadena de puntos de operación de salida, especificado a continuación, separada por el carácter '#' de las subcadenas de perfil-clasificación-nivel. La output operation point sub-string, si está presente, sigue las profile-tier-level sub-strings.

La subcadena de perfil-clasificación-nivel puede especificarse como se describió anteriormente. Más generalmente, el parámetro de códecs puede especificarse como se describió anteriormente, excepto por la subcadena de puntos de operación de salida descrita a continuación.

La subcadena de puntos de operación de salida, si está presente, puede comprender los valores de la siguiente lista, separados por caracteres de punto ("."):

- Índice de conjunto de capas de salida para un conjunto de capas de salida incluido en cada conjunto de parámetros de vídeo activo, codificado como un número decimal. Las capas necesarias del conjunto de capas de salida indicado deberán ajustarse a la información de perfil-clasificación-nivel indicada en la *ListItem*. Este índice del conjunto de capas de salida puede usarse como el valor *TargetOlsIdx* proporcionado como entrada al proceso de decodificación VC de HE. El índice del conjunto de capas de salida deberá proporcionarse para los flujos de bits de HEVC que no contengan una capa base de HEVC. Cuando no está presente, el índice del conjunto de capas de salida puede inferirse para que sea igual a 0.

- El valor *TemporalId* más alto, codificado como un número decimal, correspondiente a la información de perfil-clasificación-nivel dada a todas las capas necesarias. Este valor *TemporalId* más alto puede usarse como el valor *TemporalId* proporcionado como entrada al proceso de decodificación de HEVC. Cuando no está presente, puede inferirse que el valor *TemporalId* más alto es igual a 6.

A continuación, se proporcionan algunos ejemplos de valores de códecs (suponiendo el carácter separador '#' entre las subcadenas de perfil-clasificación-nivel):

codecs=hev1.A1.80.L93.B0!hev1.A7.1.L120.B0#2 puede indicar un flujo de dos capas, progresivo, no empaquetado, donde la pista que contiene la capa base usa el tipo de entrada de muestra 'hev1', la capa base se ajusta al Main Profile, Main Tier, Level 3.1, la pista que contiene la capa de mejora usa el tipo de entrada de muestra 'hev1' (p. ej., puede estar contenido en la pista que también contiene la capa base), la capa de mejora se ajusta al Scalable Main Profile, Main Tier, Level 4. El índice de conjuntos de capas de salida que contiene la capa de mejora y la capa base como capas necesarias es 2. *codecs=1hv1.A7.1.L120.B0#2, hev1.A1.80.L93.B0#1* puede indicar un flujo de dos capas, progresivo, no empaquetado, donde la pista que contiene la capa de mejora usa el tipo de entrada de muestra '1hv1', la capa de mejora se ajusta al Scalable Main Profile, Main Tier, Level 4, la pista que contiene la capa base usa el tipo de entrada de muestra 'hev1' y la capa base se ajusta al Main Profile, Main Tier, Level 3.1. El índice de conjuntos de capas de salida que contiene la capa de mejora y la capa base como capas necesarias es 2, mientras que el índice de conjuntos de capas de salida que contiene la capa base solamente es 1. *codecs=1hv1.A7.1.L120.B0#1, avc1.64081F#1* puede indicar que la pista que contiene la capa de mejora usa el tipo de entrada de muestra '1hv1', la capa de mejora se ajusta al Scalable Main Profile, Main Tier, Level 4, la capa de mejora contiene imágenes no empaquetadas, y la capa de mejora predice a partir de una capa base codificada de AVC. La capa base codificada de AVC utiliza el tipo de entrada de muestra 'avc1', y la capa base se ajusta al Progressive High Profile, Level 3.1. El índice de conjuntos de capas de salida que contiene la capa de mejora y la capa base como capas necesarias es 1.

En algunas realizaciones, el parámetro de códecs especificado en realizaciones anteriores o similar a realizaciones anteriores puede usarse con otros tipos de archivo contenedor distintos a ISO BMFF. Por ejemplo, en algunas realizaciones, el parámetro de códecs especificado en realizaciones anteriores o similar a realizaciones anteriores puede usarse con el atributo CODECS de la etiqueta EXT-X-STREAM-INF del archivo de lista de reproducción de HTTP Live Streaming (HLS). Esto puede permitir la señalización de una o más listas de perfil-clasificación-nivel (correspondientes a

capas), el índice de conjuntos de capas de salida que se usará en la decodificación, y la subcapa más alta se usará en la decodificación dentro del archivo de lista de reproducción. En otro ejemplo, puede definirse un nuevo atributo de la etiqueta EXT-X-STREAM-INF u otra etiqueta del archivo de lista de reproducción de HTTP Live Streaming (HLS) para transportar una o más listas de perfil-clasificación-nivel (correspondientes a capas), un índice de conjuntos de capas de salida que se usará en la decodificación, y la subcapa más alta que se usará en la decodificación dentro del archivo de lista de reproducción.

En una realización, que puede aplicarse junto con o independientemente de otras realizaciones, al menos uno del conjunto de capas de salida y la subcapa más alta que se aplica a la lista asociada de valores de perfil-clasificación-nivel se puede proporcionar en un manifiesto, tal como una MPD de DASH.

Se afirma que se pueden necesitar medios directos para concluir los valores de estas dos variables provenientes de la MPD debido a las siguientes razones.

Primero, sería posible usar dependencyId para concluir qué Representaciones (y, por consiguiente, capas) son necesarias para decodificar una Representación seleccionada (correspondiente a una capa particular o capas particulares). Después de eso, esta lista de capas podría usarse para identificar una capa establecida dentro del(los) conjunto(s) de parámetros de vídeo activo. Esto no solo puede requerir el análisis de parámetros de vídeo en el jugador (fuera del decodificador), pero también el mismo conjunto de capas puede estar asociado con más de un conjunto de capas de salida y, por lo tanto, puede no ser posible concluir en todas las situaciones cuál conjunto de capas de salida debería seleccionarse como el conjunto de capas de salida objetivo.

En segundo lugar, la MPD no contiene información que permita asociar una subcapa temporal más alta con la Representación.

En una realización, en al menos uno de los conjuntos de capas de salida y la subcapa más alta que se aplica a la lista asociada de valores de perfil-clasificación-nivel, puede indicarse dentro de un tipo MIME que incluye parámetros, y el tipo MIME se proporciona en un manifiesto, tal como una MPD de DASH. Por ejemplo, el atributo @mimeType de la MPD de DASH puede incluir una indicación de al menos uno del conjunto de capas de salida y la subcapa más alta.

En una realización, en al menos uno de los conjuntos de capas de salida y la subcapa más alta que se aplica a la lista asociada de valores de perfil-clasificación-nivel puede indicarse dentro de un parámetro de códecs y/o cualquier otro parámetro MIME, y el parámetro de códecs y/o cualquier otro parámetro MIME se proporciona en un manifiesto, tal como una MPD de DASH. Por ejemplo, el atributo @codecs de la MPD de DASH puede incluir una indicación de al menos uno del conjunto de capas de salida y la subcapa más alta.

En una realización, al menos uno del conjunto de capas de salida y la subcapa más alta que se aplica a la lista asociada de valores de perfil-clasificación-nivel puede indicarse dentro de atributos específicos o tal en un manifiesto, tal como una MPD de DASH.

En una realización, en al menos uno de los conjuntos de capas de salida y la subcapa más alta que se aplica a la lista asociada de valores de perfil-clasificación-nivel puede indicarse dentro de descriptores esenciales o tal en un manifiesto, tal como una MPD de DASH. Como TargetOlsIdx y HighestTid son específicos para HEVC, ellos pueden indicarse con descriptores (como se define para la MPD de DASH) en lugar de atributos genéricos. Además, como TargetOlsIdx y HighestTid pueden ser necesarios para corregir la decodificación de HEVC de multicapas, ellos pueden proporcionarse con un descriptor EssentialProperty.

Los descriptores EssentialProperty con @schemeldURI iguales a "urn:mpeg:dash:hevc:2015" pueden proporcionar el (los) conjunto(s) de capas de salida objetivo y la(s) subcapa(s) temporal(es) más alta(s) asociada(s) con AdaptationSet, representación o subrepresentación que contiene el descriptor EssentialProperty.

El parámetro @value del descriptor EssentialProperty con @schemeldURI igual a "urn:mpeg:dash:hevc:2015" puede tener la siguiente sintaxis y semántica.

```
<!-- HevcDescriptor -->
```

```
<xs:complexType name="HevcDescriptorType">
```

```
<xs:sequence>
```

```
<xs:element name="OutputOp" type="OutputOpType" maxOccurs="unbounded"/>
```

```
<xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence>
```

```
<xs:any Attribute namespace="##other" processContents="lax"/>
```

```
</xs:complexType>
```

```
<!-- Output operation point -->
```

```
<xs:complexType name="OutputOpType">
```

```
<xs:sequence>
```

```
<xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
```

```
</xs:sequence>
```

```
<xs:attribute name="targetOlsIdx" type="xs:unsignedInt" use="required"/>
```

```
<xs:attribute name="highestTid" type="xs:unsignedInt" use="required"/>
```

```
<xs:attribute name="outputOpCodecs" type="xs:unsignedInt"/>
```

```
<xs:attribute name="outputReps" type="xs:string"/>
```

```
<xs:any Attribute namespace="##other" processContents="lax"/>
```

```
</xs:complexType>
```

A continuación, los elementos y los nombres de atributo de la sintaxis anterior se describen con más detalle, de acuerdo con una realización.

OutputOp puede tener valores 1..N y especifica un punto de operación de salida que incluye la representación o subrepresentación que contiene este descriptor EssentialProperty. Para la semántica a continuación, representación o subrepresentación se refiere a la representación o subrepresentación correspondiente al elemento representación o subrepresentación que contiene este descriptor EssentialProperty. Cuando este descriptor EssentialProperty se incluye en AdaptationSet, la representación que lo contiene se refiere a cualquier representación individual dentro de AdaptationSet.

@targetOlsIdx, que puede ser obligatorio, especifica el índice del conjunto de capas de salida asociadas con el punto de operación de salida.

@highestTid, que puede ser obligatorio, especifica el valor de TemporalId de la subcapa más alta asociada con el punto de operación de salida.

@outputOpCodecs, que puede ser opcional, especifica el parámetro de códecs para el punto de operación de salida. El contenido de este atributo debería ser conforme a las producciones de RFC6381 ya sea de la simp-list o la fancy-list, sección 3.2, sin los caracteres de inclusión DQUOTE. La simp-list o fancy-list usadas como contenido de este atributo debería consistir en un elemento de lista restringido como se especifica en los siguientes párrafos.

El identificador de códec que inicia el valor del parámetro de códecs debería ser el mismo que el que se aplica a la representación o subrepresentación.

El elemento de lista debería incluir una profile-tier-level sub-string para cada capa necesaria en el punto de operación de salida.

Cada profile-tier-level sub-string debería incluir la información de perfil y nivel.

@outputReps, que puede ser opcional, especifica una lista separada por espacios en blanco de los valores @id de las representaciones cuyas capas son capas de salida en el punto de operación de salida. Cuando este atributo no está presente, las capas de salida del punto de operación de salida son aquellas que están presentes en la representación o subrepresentación que las contienen. Cuando este atributo está presente, las capas de salida del punto de operación de salida son aquellas que están presentes en la representación o subrepresentación que las contienen y aquellas que están presentes en las representaciones con @id igual a cualquier elemento de la lista @outputReps.

A continuación, se proporciona un ejemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<MPD
```

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

5  xmlns= "urn: mpeg: dash: schema: mpd: 2011"
   xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 DASH-MPD.xsd"
   type=" static"

10  mediaPresentationDuration="PT30S"
   minBufferTime= "PT10.00S"
   profiles="urn:mpeg:dash:profile:isoff-main:2011">
15  <BaseURL>http://www.example.com/</BaseURL>
   <!-- En este período hay 3 vistas: provenientes de tres cameras alineadas: C1-C2-C3.
20  Cada uno de C1+C2 y C2+C3 forma un par estéreo a excepción de C1+C3.
   C2 se toma como la vista de base para MV-HEVC mientras que C1 y C3 son vistas de mejora
   -->
25  <Period start="PT0.00S" duration="PT30.00S">
   <SegmentList>
30  initialization sourceURL="seg-m-init.mp4"/>
   </SegmentList>
   <AdaptationSet mimeType="video/mp4" codecs="hev1.A1.80.L93.B0">
35  <Role schemeldUri="urn:mpeg:dash:stereoid:2011" value="11 r0"/>
   <Representation id="C2" bandwidth="2000000">
40  <SegmentList duration="10">
   <SegmentURL media="seg-m1-C2view-1.mp4"/>
   <SegmentURL media="seg-m1-C2view-2.mp4"/>
45  <SegmentURL media="seg-m1-C2view-3.mp4"/>
   </SegmentList>
50  </Representation>
   </AdaptationSet>
   <!-- El siguiente conjunto de adaptación contiene una representación funcionalmente idéntica a la representación
55  en el conjunto de adaptación anterior. Por lo tanto, estos tienen la misma Representation@id. Esto se hace para
   la compatibilidad con los receptores 2D que no comprenden el schemeldURI del descriptor de funciones y puede
   ignorar el conjunto de adaptación -->
   <AdaptationSet mimeType="video/mp4" codecs="hev1.A1.80.L93.B0">
60  <Representation id="C2" bandwidth="2000000">
   <SegmentList duration="10">
65  <SegmentURL media="seg-m1-C2view-1.mp4"/>

```

```

<SegmentURL media="seg-m1-C2view-2.mp4"/>
<SegmentURL media="seg-m1-C2view-3.mp4"/>
5  </SegmentList>
    </Representation>
    </AdaptationSet>
10  <!-- El siguiente conjunto de adaptación contiene la vista C1.
      Las vistas C1 y C2 forman un conjunto de capas de salida con índice 1. -->
15  <AdaptationSet mimeType="video/mp4" codecs="1hv1.A6.2.L93.B0">
    <Role schemeldUri="urn:mpeg:dash:stereoid:2011" value="10"/>
    <EssentialDescriptor schemeldUri="urn:mpeg:dash:hevc:2015" value="
20  <OutputOp targetOlsIdx="1" highestTid="6" outputReps="C2"/>
    "/>
25  <Representation id="C1" dependencyId="C2" bandwidth=" 1000000">
    <SegmentList duration="10">
    <SegmentURL media="seg-m1 -C1 view-1 .mp4"/>
30  <SegmentURL media=" seg-m1 -C1 view-2.mp4"/>
    <SegmentURL media="seg-m1 -C1 view-3.mp4"/>
35  </SegmentList>
    </Representation>
    </AdaptationSet>
40  <!-- El siguiente conjunto de adaptación contiene la vista C3.
      Las vistas C3 y C2 forman un conjunto de capas de salida con índice 2. -->
45  <AdaptationSet mimeType="video/mp4" codecs="1hv1.A6.2.L93.B0">
    <Role schemeldUri="urn:mpeg:dash:stereoid:2011" value="rl"/>
    <EssentialDescriptor schemeldUri="urn:mpeg:dash:hevc:2015" value="
50  <OutputOp targetOlsIdx="2" highestTid="6" outputReps="C2"/>
    "/>
55  <Representation id="C3" dependencyId="C2" bandwidth=" 1000000">
    <SegmentList duration="10">
    <SegmentURL media="seg-m1-C3view-1.mp4"/>
60  <SegmentURL media="seg-m1-C3view-2.mp4"/>
    <SegmentURL media="seg-m1-C3 view-3.mp4"/>
65  </SegmentList>

```

</Representation>

</AdaptationSet>

5 </Period>

</MPD>

10 En una realización, que puede aplicarse junto con o independientemente de otras realizaciones, las características que se aplican y/o se necesitan para decodificar una secuencia de segmentos (o respectivamente subsegmentos) consecutivos no superpuestos se codifican en un MPD (o cualquier otro manifiesto) y/o se analizan a partir del MPD (o cualquier otro manifiesto). Las representaciones de las que pueden originarse los segmentos (o respectivamente subsegmentos) pueden indicarse en el MPD o analizarse a partir del MPD. Las características pueden incluir, pero no se limitan a, uno o más de lo siguiente:

- 15 – Los códecs o formatos de codificación a los que se adapta la secuencia.
- La lista de valores de nivel de perfil o semejantes a los que se adapta la secuencia.
- 20 – El índice de conjunto de capas de salida que puede usarse como el índice de conjunto de capas de salida objetivo (p. ej., TargetOlsIdx de HEVC) cuando se decodifica la secuencia.
- El TemporalId más alto (o semejante, p. ej., HighestTid de HEVC) que puede usarse cuando se decodifica la secuencia.
- 25 – Información relacionada con el cambio de resolución adaptativa, p. ej., single_layer_for_non_irap_flag de HEVC o ambos single_layer_for_non_irap_flag y higher_layer_irap_skip_flag de HEVC, o información similar.
- Valores de identificador de capa (o semejantes, p. ej. valores nuh_layer_id de HEVC) incluidos en la secuencia.
- 30 – Valores de identificador de capa (o semejantes) de capas necesarias incluidas en la secuencia.
- Valores de identificador de capa (o semejantes) de capas de salida incluidas en la secuencia.
- 35 – Información sobre capas de salida alternativas, p. ej., el valor de alt_output_layer_flag de HEVC para el conjunto de capas de salida representado por la secuencia.
- El (los) conjuntos de parámetros de vídeo que pueden activarse al decodificar la secuencia.
- 40 – Otro(s) conjunto(s) de parámetro(s), tales como conjuntos de parámetros de secuencia y/o conjuntos de parámetros de imagen, que pueden activarse al decodificar la secuencia.
- Valores de identificador de escalabilidad (o semejantes, p. ej., en HEVC: ScalabilityId[i] para i en el rango de 0 a 15) para las capas incluidas en la secuencia.
- 45 – Información relacionada con el remuestreo implícito de imágenes de referencia, capaz de indicar, por ejemplo, uno de los siguientes casos: no se sugiere remuestreo implícito, puede ser necesario un remuestreo implícito. Adicional o alternativamente, las características pueden comprender un límite del número de imágenes que pueden necesitar remuestrearse al cambiar las representaciones. La información relacionada con el remuestreo implícito de imágenes de referencia puede darse por separado para la conmutación ascendente y la conmutación descendente y/o para cada par de representaciones o grupos de representaciones entre las que ocurre la conmutación.
- 50

55 Las características mencionadas anteriormente pueden indicarse de manera similar o idéntica a cualquier opción, combinación o subconjunto de la señalización para combinaciones de nivel de perfil y puntos de operación de salida como se presentan anteriormente. Por ejemplo, las características mencionadas anteriormente pueden incluirse como valores de parámetros MIME y el tipo MIME, incluyendo sus parámetros, puede proporcionarse para la secuencia de segmentos (o respectivamente subsegmentos) consecutivos no superpuestos dentro del MPD, o puede analizarse para la secuencia de segmentos (o respectivamente subsegmentos) consecutivos no superpuestos del MPD.

60 En una realización, las características mencionadas anteriormente se codifican en una estructura complementaria dentro de un MPD (o cualquier otro manifiesto). La estructura complementaria es tal que puede, pero no requiere, ser procesada por un cliente o lo similar. En una realización, las características mencionadas anteriormente se analizan a partir de una estructura complementaria del MPD (o cualquier otro manifiesto).

65

En una realización, las características mencionadas anteriormente están acompañadas por uno o más de los siguientes fragmentos de información, que pueden codificarse en un MPD (o cualquier otro manifiesto) y/o analizarse a partir del MPD (o cualquier otro manifiesto):

5 – Información indicativa de uno o más de: las características mencionadas anteriormente se aplican a una secuencia de segmentos consecutivos no superpuestos, las características mencionadas anteriormente aplican una secuencia de subsegmentos consecutivos no superpuestos.

10 – Identificación, tales como valores del identificador, de representaciones a las que se aplican las características mencionadas anteriormente. Estas representaciones identificadas pueden denominarse un conjunto de conmutación.

En una realización, se infieren las representaciones a las que se aplican las características mencionadas anteriormente. Por ejemplo, cuando las características se indican para una representación de un conjunto de adaptación, se puede inferir que las características se aplican a todas las representaciones del mismo conjunto de adaptación. Las representaciones a las que se aplican las características mencionadas anteriormente pueden denominarse un conjunto de conmutación.

En una realización, las características mencionadas anteriormente se aplican a una secuencia de segmentos o subsegmentos no superpuestos, que pueden originarse a partir de cualquier representación en el conjunto de conmutación. En otra realización, las características mencionadas anteriormente se aplican a una secuencia de segmentos no superpuestos o subsegmentos que originan representaciones en el conjunto de conmutación donde pueden colocarse restricciones en las representaciones a partir de las cuales pueden originarse segmentos o subsegmentos consecutivos en la secuencia. Por ejemplo, se puede requerir que segmentos o subsegmentos consecutivos se originen a partir de las mismas representaciones o representaciones consecutivas, donde las representaciones pueden ordenarse, por ejemplo, basándose en los valores o velocidades de bits del identificador.

En una realización, un descriptor de propiedad complementario que incluye las características mencionadas anteriormente se codifica en un MPD (o cualquier otro manifiesto) y/o se analiza a partir del MPD (o cualquier otro manifiesto).

En una realización, un descriptor de propiedad complementario que incluye las características mencionadas anteriormente también incluye uno o más de:

35 – Información indicativa de uno o más de: las características mencionadas anteriormente se aplican a una secuencia de segmentos consecutivos no superpuestos, las características mencionadas anteriormente aplican una secuencia de subsegmentos consecutivos no superpuestos.

40 – Identificación, tales como valores del identificador, de representaciones a las que se aplican las características mencionadas anteriormente. Estas representaciones identificadas pueden denominarse un conjunto de conmutación.

En una realización, un descriptor de propiedad complementario se especifica como se muestra a continuación o de manera similar.

Los descriptores SupplementalProperty con @schemeldURI iguales a “urn:mpeg:dash:hevc:segseq:2015” (o cualquier otro URN escogido para indicar la realización descrita, cuando se aplica a segmentos) puede proporcionar características de una secuencia de segmentos consecutivos de más de una representación identificada. Los descriptores SupplementalProperty con @schemeldURI iguales a “urn:mpeg:dash:hevc:subsegseq:2015” (o cualquier otro URN escogido para indicar la realización descrita, cuando se aplica a subsegmentos) puede proporcionar características de una secuencia de subsegmentos consecutivos de más de la representación identificada. Ambos tipos de descriptor pueden tener la misma sintaxis y semántica, especificadas, p. ej., como se muestra a continuación:

El valor @value de SupplementalProperty con @schemeldURI igual a “urn:mpagr:dashhvc:segment:2015” o “urn:mpeg:dash:hevc:subsegseq:2015” es una lista separada por comas de los siguientes valores (en el orden enumerado a continuación):

– Se puede incluir una lista separada por espacios en blanco de valores @id de representaciones de las cuales pueden incluirse segmentos o subsegmentos en la secuencia de segmentos o subsegmentos consecutivos, respectivamente.

– El parámetro de códecs y/o cualquier otro parámetro de medios MIME indicativo de la combinación de nivel de perfil o lo similar, que se aplica a la secuencia de segmentos o subsegmentos. Por ejemplo, el parámetro de códecs como se especificó anteriormente puede incluir una *ListItem* que se aplica a la secuencia de segmentos o subsegmentos. El parámetro de códecs se puede requerir para indicar un códec de la familia HEVC.

– La cadena de máscara binaria de 2 dígitos de elementos de sintaxis `single_layer_for_non_irap` y `higher_layer_irap_skip_flag` de HEVC que están en vigentes para la secuencia de segmentos `SegSeq` o para la secuencia subsegmentos `SeqSubSeq` (como se especifica a continuación). Los valores de máscara binaria permitidos son '00', '10' y '11'.

Debe entenderse que además o en lugar del parámetro de códecs y la cadena de máscara binaria de 2 dígitos, otros parámetros indicativos de las mismas o similares propiedades pueden incluirse en la sintaxis del atributo `@value`. Por ejemplo, el atributo `@value` puede incluir el tipo MIME, incluyendo sus parámetros MIME, que se aplica a la secuencia de segmentos o subsegmentos consecutivos.

Dejar que `R1` sea la representación que contiene el `SupplementalProperty` con `@schemeldURI` igual a “urn:mpeg:dash:hevc:segseq:2015” o “urn:mpeg:dash:hevc:subsegseq:2015”, y dejar que `R2`, `R3`, ..., `RN` sean las representaciones identificadas por sus valores `@id` dados en `@value` del `SupplementalProperty`. Se pueden requerir incluir las representaciones `R1`, `R2`, ..., `RN` en el mismo conjunto de adaptación y se puede requerir tener el mismo número `M` de segmentos de medios.

Cuando el `SupplementalProperty` con `@schemeldURI` igual a “urn:mpeg:dash:hevc:segseq:2015” está presente, se puede aplicar lo siguiente:

Dejar

- $S_{i,j}$, para $j > 0$, ser el j -ésimo segmento de medios en la i -ésima representación (es decir, R_i)
- si está presente, dejar que $S_{i,0}$ sea el segmento de inicialización en la i -ésima representación, y
- si está presente, dejar que B_i sea el segmento de conmutación de secuencia de bits en la i -ésima representación.

Se puede requerir que $S_{i,j}$ no se superponga con $S_{m,n}$, para cualquier valor de i y m en el rango de 1 a N , inclusive, y j y n en el rango de 1 a M , inclusive, de tal manera que i es no igual a M , y j no es igual a N .

Dejar que `SegSeq` sea la secuencia de los siguientes segmentos:

- El segmento de inicialización $S_{1,0}$, cuando está presente
- Si están presentes segmentos de conmutación de secuencia de bits,

$$B_{i(1)}, S_{i(1),1}, B_{i(2)}, S_{i(2),2}, \dots, B_{i(k)}, S_{i(k),k}, \dots, B_{i(M)}, S_{i(M),M}$$

o de otra manera

$$S_{i(1),1}, S_{i(2),2}, \dots, S_{i(k),k}, \dots, S_{i(M),M},$$

en donde cualquier $i(k)$ para todos los valores k en el rango de 1 a M , respectivamente, es un valor entero en el rango de 1 a N .

Cualquier `SeqSeq` puede resultar en una “conforming Segment sequence” con el formato de medios como se especifica en el parámetro de códecs proporcionado dentro de `@value` de `SupplementalProperty`.

Cualquier `SegSeq` puede proporcionar un playout continuo y puede tener una calidad aceptable.

Cuando la `SupplementalProperty` con `@schemeldURI` igual a

“urn:mpeg:dash:hevc:subsegseq:2015” está presente, puede aplicarse lo siguiente:

Dejar

- $S_{i,j,k}$, para $j > 0$, ser el k -ésimo subsegmento del j -ésimo segmento de medios en la i -ésima representación (es decir, R_i)
- $L_{i,j}$ ser igual al número de subsegmentos en el j -ésimo segmento de medios en la i -ésima representación.
- si está presente, dejar que $S_{i,0}$ sea el segmento de inicialización en la i -ésima representación, y
- si está presente, dejar que B_i sea el segmento de conmutación de secuencia de bits en la i -ésima representación.

Se puede requerir que $L_{i,j}$ que sea igual a $L_{h,j}$ para cualquier valor de h e i en el rango de 1 a N , inclusive, de tal manera que h no sea igual a i . Dejar que L_j sea igual a $L_{i,j}$ para cualquier valor de i en el rango de 1 a N , inclusive. Dejar que L sea igual a la suma de L_j para todos los valores de j en el rango de 1 a M , inclusive.

- 5 Se puede requerir que $S_{i,j,k}$ no se superponga con $S_{m,n,o}$ para cualquier valor de i y m en el rango de 1 a N , inclusive, y j y n en el rango de 1 a M , inclusive, y k y o en el rango de 1 al número de subsegmentos en el segmento, inclusive, de tal manera que i no es igual a m , y j no es igual a n , y k no es igual a o .

Dejar que SubSeqSeq sea la secuencia de los siguientes segmentos y subsegmentos:

- 10 – El segmento de inicialización $S_{i,o}$, cuando está presente
- Si están presentes segmentos de conmutación de secuencia de bits,
- 15 $B_{i(1)}, S_{i(1),1,1}, B_{i(2)}, S_{i(2),1,2}, \dots, B_{i(L)}, S_{i(L),1,L}, B_{i(L+1)}, S_{i(L+1)}, B_{i(L+2)}, S_{i(L+2),2,2}, \dots, B_{i(a)}, S_{i(a),j,k}, \dots, B_{i(L)}, S_{i(L),M,LM},$
o de otra manera

$$S_{i(1),1,1}, S_{i(2),1,2}, \dots, S_{i(L),1,L}, S_{i(L+1),2,1}, S_{i(L+2),2,2}, \dots, S_{i(a),j,k}, \dots, S_{i(L),M, LM},$$

- 20 en donde cualquier $i(a)$ para todos los valores en el rango de 1 a L , respectivamente, es un valor entero en el rango de 1 a N .

- 25 Cualquier SubSeqSeq puede resultar en una “conforming Segment sequence” con el formato de medios como se especifica en el parámetro de códecs proporcionado dentro de @valuer de SupplementalProperty.

Cualquier SubSeqSeq puede proporcionar un playout continuo y puede tener una calidad aceptable.

A continuación, se proporciona un ejemplo:

- 30 <?xml version="1.0" encoding="UTF-8"?>
- <MPD
- 35 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
- xmlns= "urn: mpeg: dash: schema: mpd: 2011"
- xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 DASH-MPD.xsd"
- 40 type=" static"
- mediaPresentationDuration="PT3256S"
- 45 minBufferTime= "PT 1.2S"
- profiles= "urn: mpeg: dash :profile: isoff-main: 2011 ">
- <BaseURL>http://cdn1.example.com/</BaseURL>
- 50 <BaseURL>http://cdn2.example.com/</BaseURL>
- <Period>
- 55 <AdaptationSet
- segment Alignment=" true"
- frameRate="30">
- 60 <!-- 720p Representation -->
- <Representation
- 65 mimeType=" video/mp4"

```

<!-- HEVC Main profile, Main tier, Level 3.1, progressive and non-packed content -- >
codecs="hev1.A1.80.L93.B0"
5 id="v720p"
startWithSAP="2"
width="1280"
10 height="720"
bandwidth=" 1500000">
15 <BaseURL> video-720p. mp4</BaseURL>
<SegmentBase indexRange="0-4332"/>
20 <!-- Cuando se concatenan segmentos de las representaciones 720p y 1080p, la secuencia de segmento resultante
se ajusta a una secuencia de bits de dos capas, donde la capa base es HEVC Main profile, Main tier, Level 3.1 y la
capa de mejora es HEVC Scalable Main Profile, Main tier, Level 4. Además, la secuencia de segmento resultante se
restringe de tal modo que single_layer_for_non_irap_flag es igual a 1 y higher_layer_irap_skip_flag es igual a 1. -->
25 <SupplementalDescriptor                                schemeldUri=
value="v1080p,hev1.A1.80.L93.B0!1he1.A7.1.L120.B0,11"/>                                "urn:mpeg:dash:hevc:seqseq:2015"
</Representation>
<!-- 1080p Representation -- >
30 <Representation
mimeType=" video/mp4"
35 <!--Main profile, Main tier, Level 4, progressive and non-packed content, general_inbld_flag es igual a 1 para indicar
que las unidades VCL NAL de representación no están en la capa base. -->
codecs="hev1.A1.80.L120.B0.0.0.0.1"
40 id="v1080p"
startWithSAP="3"
width="1920"
45 height="1080"
bandwidth="3000000">
50 <BaseURL>video-1080p.mp4</BaseURL>
<SegmentBase indexRange="0-3752"/>
</Representation>
55 </AdaptationSet>
</Period>
60 </MPD>

```

En una realización, que puede aplicarse junto con o independientemente de otras realizaciones, las secuencias de bits o representaciones según RARC, p. ej., las descritas con las figuras 9a y 9b, están encapsuladas en un formato de archivo y/o un formato de transporte como se muestra a continuación. La representación en baja resolución se incluye en un archivo o segmento diferente de aquel o aquellos, respectivamente, donde se incluye la representación en alta resolución. La pista de las imágenes de salto de la representación en baja resolución tiene el mismo valor de

identificador de pista (p. ej., track_id) que el de la pista que contiene la representación en alta resolución. La entrada de muestra de la pista que contiene imágenes de salto puede indicar la codificación en capas, p. ej., usar tipos de entrada de muestra '1hvc1' o '1he1', y/o puede indicarse que se requiera la decodificación de un perfil de múltiples capas, tal como el principal perfil escalable de HEVC. La pista que contiene la capa base de la representación en baja resolución y la pista que contiene la representación en alta resolución pueden indicar la codificación de una única capa, p. ej., usar tipos de entrada de muestra 'hvc1' o 'hev1', y/o puede indicarse que se requiera la decodificación de un perfil de una única capa, tal como el principal perfil escalable de HEVC, y/o puede indicarse que se requiera capacidad INBLD.

En una realización, que puede aplicarse junto con o independientemente de otras realizaciones, un cliente o reproductor de flujo continuo puede funcionar como se muestra a continuación al recibir representaciones según RARC que han sido encapsuladas como se describe en el párrafo anterior. El reproductor puede continuar usando el segmento de inicialización de la representación en baja resolución al conmutar para recibir la representación en alta resolución. En consecuencia, la entrada de muestra para la pista que contiene la representación en alta resolución puede indicar la codificación en capas, p. ej., los tipos de entrada de muestra '1hvc1' o '1he1', y/o puede indicar un perfil de múltiples capas. Por lo tanto, el reproductor puede concluir que la combinación de las pistas recibidas de la representación en baja resolución y la representación en alta resolución representa una secuencia de bits de vídeo escalable y puede requerir la capacidad de decodificación de un perfil de múltiples capas.

En una realización, que puede aplicarse junto con o independientemente de otras realizaciones, las imágenes de salto redundantes pueden usarse como se muestra a continuación. Esta realización puede denominarse cambio de resolución adaptativa redundante para conmutar a la representación dependiente (RARC-D). Una representación en baja resolución puede codificarse convencionalmente, p. ej., como una secuencia de bits H.265/HEVC de una única capa, mientras que la representación en alta resolución puede codificarse como una capa pronosticada, es decir, como una representación dependiente, en la que al menos algunas imágenes pueden ser pronosticadas entre capas de la(s) imagen(es) de la representación en baja resolución. Además, pueden crearse imágenes de salto adicionales de la capa de mejora para cada imagen de referencia de las imágenes RASL en alta resolución que anteceden a la imagen CRA asociada en orden de decodificación. Esto puede hacerse solo para aquellas imágenes CRA que comienzan un segmento o un subsegmento. Las imágenes de salto pertenecen conceptualmente a la capa de mejora (p. ej., con nuh_layer_id igual a 1) de la representación en baja resolución, y pueden estar encapsuladas en los segmentos de la representación en baja resolución.

En una realización, que puede aplicarse junto con o independientemente de otras realizaciones, un cliente o reproductor de flujo continuo puede funcionar como se muestra a continuación al recibir representaciones según RARC-D. El reproductor recibe primero segmento(s) de una representación en baja resolución. El reproductor determina entonces conmutar a una mayor resolución y solicita segmento(s) correspondiente(s) tanto de la representación en baja resolución como de la representación en alta resolución. Al conmutar de subida, el reproductor puede decodificar la(s) imagen(es) de salto del último segmento para el cual solo se recibe la representación en baja resolución. Esta(s) imagen(es) de salto pueden usarse como referencia cuando se decodifican las imágenes RASL de la imagen CRA inicial del primer segmento recibido de la representación en alta resolución. La determinación de conmutación puede resultar, por ejemplo, de la adaptación de la velocidad o del arranque rápido.

La Figura 9c ilustra la codificación y preparación de segmentos según una realización. Esta realización puede denominarse remuestreo implícito de imágenes de referencia (IRPR).

Tanto la representación 909 en baja resolución como la representación 911 en alta resolución pueden codificarse como secuencias de bits H.265/HEVC ordinarias de una única capa. La representación 909 en baja resolución puede codificarse usando GOP cerrados, mientras que la representación 911 en alta resolución usa GOP abiertos.

El funcionamiento del método IRPR durante la conmutación ascendente se ilustra en la Figura 9d. El reproductor determina conmutar de la representación 1 (la representación 909 en baja resolución en este ejemplo) a la representación 2 (la representación 911 en alta resolución en este ejemplo). En consecuencia, el reproductor puede solicitar y recibir el siguiente segmento (imagen 1 en adelante) desde la representación 2 en lugar de la representación 1. Para evitar un fallo técnico en la velocidad de imágenes mostradas, el reproductor puede desear decodificar las imágenes RASL 2..8. El desafío es que una (o más) de las imágenes de referencia de las imágenes RASL no se ha (o han) recibido, y por lo tanto un decodificador H.265/HEVC estándar puede no decodificar con éxito estas imágenes RASL. En IRPR, el reproductor muestra las imágenes de representación 1 que corresponden a la(s) imagen(es) de referencia de las imágenes RASL y usa la(s) imagen(es) muestreadas ascendentemente como referencia para la predicción de imágenes RASL. En el ejemplo, imágenes 2..8, la representación 2 usa la imagen 0 muestreada ascendentemente de la representación 1 (etiquetada como 912 en la Figura 9d) como la imagen de referencia (directa o indirecta). El muestreo ascendente realizado como parte del método IRPR cambia el proceso de decodificación de H.265/HEVC.

En algunas otras realizaciones, el codificador puede apagar selectivamente el filtrado en bucle, tal como SAO, p. ej., para todas las imágenes RASL o las imágenes RASL que inician segmentos.

Se han descrito realizaciones con referencia a segmentos. Debe entenderse que las realizaciones podrían describirse de manera similar con referencia a los subsegmentos.

Un beneficio del uso de GOP abiertos en lugar de GOP cerrados puede ser una mejor eficacia de compresión, que, por ejemplo, significa que se puede lograr la misma calidad de imagen con menos velocidad de bits.

La Figura 12 muestra un diagrama de bloques de un decodificador de vídeo adecuado para usar las realizaciones de la invención. La Figura 12 representa una estructura de un decodificador de dos capas, pero se apreciará que las operaciones de decodificación pueden usarse de manera similar en un decodificador de una única capa.

El decodificador de vídeo 550 comprende una primera sección de decodificador 552 para componentes de vista base y una segunda sección de decodificador 554 para componentes de vista no base. El bloque 556 ilustra un demultiplexor para entregar información sobre los componentes de vista base a la primera sección de decodificador 552 y para entregar información sobre los componentes de vista no base a la segunda sección de decodificador 554. La referencia P'n significa una representación pronosticada de un bloque de imagen. La referencia D'n significa una señal de error de predicción reconstruida. Los bloques 704, 804 ilustran imágenes reconstruidas preliminares (I'n). La referencia R'n significa una imagen reconstruida final. Los bloques 703, 803 ilustran la transformación 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 marco de referencia (RFM). Los bloques 706, 806 ilustran la predicción (P) (ya sea inter predicción o intra predicción). Los bloques 707, 807 ilustran el filtrado (F). Los bloques 708, 808 pueden usarse para combinar la información de error de predicción decodificada con componentes de vista base/no base pronosticadas para obtener las imágenes reconstruidas preliminares (I'n). Las imágenes de vista base preliminares reconstruidas y filtradas pueden ser la salida 709 de la primera sección de decodificador 552 y las imágenes de vista base preliminares reconstruidas y filtradas pueden ser la salida 809 de la primera sección de decodificador 554.

La Figura 13 es una representación gráfica de un ejemplo de sistema de comunicación multimedia dentro del cual pueden implementarse diversas realizaciones. Una fuente de datos 1510 proporciona una señal de origen en un formato analógico, digital no comprimido, o digital comprimido, o cualquier combinación de estos formatos. Un codificador 1520 puede incluir o conectarse con un preprocesamiento, tal como conversión de formato de datos y/o filtrado de la señal de origen. El codificador 1520 codifica la señal de origen en una secuencia de bits de medios codificados. Se debe tener en cuenta que una secuencia de bits que se decodifica puede recibirse directa o indirectamente desde un dispositivo remoto ubicado dentro de prácticamente cualquier tipo de red. Adicionalmente, la secuencia de bits puede recibirse desde un hardware o software local. El codificador 1520 puede ser capaz de codificar más de un tipo de medio, tal como audio y vídeo, o se puede requerir 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 secuencias de bits codificados de medios sintéticos. A continuación, se considera solo el procesamiento de una secuencia de bits de medios codificados de un tipo de medio para simplificar la descripción. Sin embargo, se debe tener en cuenta que típicamente los servicios de difusión en tiempo real comprenden varias transmisiones (típicamente al menos una transmisión de audio, vídeo y subtítulos). 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.

La secuencia de bits de medios codificados puede transferirse a un almacenamiento 1530. El almacenamiento 1530 puede comprender cualquier tipo de memoria masiva para almacenar la secuencia de bits de medios codificados. El formato de la secuencia de bits de medios codificados en el almacenamiento 1530 puede ser un formato de secuencia de bits autónomo elemental, o una o más secuencias de bits de medios codificados pueden encapsularse en un archivo contenedor. Si una o más secuencias de bits de medios están encapsulados en un archivo contenedor, un generador de archivos (no se muestra en la figura) puede usarse para almacenar la una o más secuencias 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 se conecta operativamente al codificador 1520 o al almacenamiento 1530. Algunos sistemas operan "en vivo", es decir, omiten el almacenamiento y transfieren la secuencia de bits de medios codificados desde el codificador 1520 directamente al emisor 1540. La secuencia de bits de medios codificados puede entonces transferirse al emisor 1540, también denominado servidor, según la necesidad. El formato usado en la transmisión puede ser un formato de secuencia de bits autónomo elemental, un formato de transmisión de paquete, o una o más secuencias de bits de medios codificados pueden encapsularse en un archivo contenedor. El codificador 1520, el almacenamiento 1530, y el servidor 1540 pueden encontrarse en el mismo dispositivo físico o pueden incluirse en dispositivos separados. El codificador 1520 y el servidor 1540 pueden operar con contenido en tiempo real en vivo, en cuyo caso la secuencia de bits de medios codificados típicamente no se almacena permanentemente, sino que se carga durante pequeños períodos de tiempo en el codificador

de contenido 1520 y/o en el servidor 1540 para suavizar variaciones en el retraso de procesamiento, retraso de transferencia, y velocidad de bits de medios codificados.

El servidor 1540 envía la secuencia de bits de medios codificados usando una pila de protocolos de comunicación. La pila puede incluir, pero no se limita a, uno o más del protocolo de transporte en tiempo real (RTP), protocolo de datagramas de usuario (UDP), el protocolo de transferencia de hipertexto (HTTP), el protocolo de control de transmisión (TCP), y el protocolo de Internet (IP). Cuando la pila del protocolo de comunicación se orienta a paquetes, el servidor 1540 encapsula la secuencia de bits de los medios codificados en paquetes. Por ejemplo, cuando se usa el RTP, el servidor 1540 encapsula la secuencia 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 dedicado. Nuevamente, se debe tener en cuenta que un sistema puede contener más de un servidor 1540, pero en aras de la simplicidad, la siguiente descripción solo considera un servidor 1540.

Si el contenido de medios se encapsula 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 “analizador de archivos de envío” (no se muestra en la figura). En particular, si el archivo del contenedor no se transmite como tal pero al menos uno de la secuencia de bits de medios codificados contenido está encapsulado para el transporte a través de un protocolo de comunicación, un analizador de archivos de envío ubica las partes apropiadas de la secuencia de bits de medios codificados que se transportará a través del protocolo de comunicación. El analizador de archivos de envío también puede ayudar a crear el formato correcto para el protocolo de comunicación, tal como cabeceras y cargas útiles de paquetes. El archivo contenedor multimedia puede contener instrucciones de encapsulación, como pistas de sugerencia en el formato ISO Base Media File, para la encapsulación de al menos uno de las secuencias de bits de medios contenidos en el protocolo de comunicación.

El servidor 1540 puede estar conectado o no a una puerta de enlace 1550 a través de una red de comunicación. La puerta de enlace también puede denominarse como una casilla intermediaria. Se tiene en cuenta que el sistema puede comprender generalmente cualquier número de puertas de enlace o lo similar, pero en aras de la simplicidad, la siguiente descripción solo considera una puerta de enlace 1550. La puerta de enlace 1550 puede realizar diferentes tipos de funciones, tales como paquetes de almacenamiento en caché, transmisiones o recursos, recolección previa proactiva de datos de medios, traducción de una transmisión de paquete según una pila de protocolos de comunicación a otra pila de protocolos de comunicación, fusión y bifurcación de flujo de datos, y manipulación de flujo de datos según las capacidades del enlace descendente y/o receptor, tal como controlar la velocidad de bits de la transmisión reenviada según las condiciones de red predominantes del enlace descendente. Los ejemplos de puertas de enlace 1550 incluyen unidades de control de conferencia multipunto (MCU), puertas de enlace entre telefonía de vídeo de circuitos conmutados y de paquetes conmutados, servidores de pulsar y hablar en el móvil (PoC), encapsuladores IP en sistemas de difusión de vídeo digital (DVB-H), o módulos de conexión u otros dispositivos que remiten transmisiones de difusión localmente a redes inalámbricas domésticas. Cuando se usa el RTP, la puerta de enlace 1550 puede denominarse una mezcladora RTP o un traductor RTP y puede actuar como un punto final de una conexión RTP. En lugar o además de la puerta de enlace 1550, el sistema puede incluir un empalmador que concatena la secuencia de vídeo o las secuencias de bits.

El sistema incluye uno o más receptores 1560, típicamente capaces de recibir, demodular, y desencapsular la señal transmitida en una secuencia de bits de medios codificados. La secuencia de bits de medios codificados puede transferirse a un almacenamiento de grabación 1570. El almacenamiento de grabación 1570 puede comprender cualquier tipo de memoria masiva para almacenar la secuencia de bits de medios codificados. El almacenamiento de grabación 1570 puede comprender de forma alternativa o aditiva una memoria de cálculo, tal como una memoria de acceso aleatorio. El formato de la secuencia de bits de medios codificados en el almacenamiento de grabación 1570 puede ser un formato de secuencia de bits autónomo elemental, o una o más secuencias de bits de medios codificados pueden encapsularse en un archivo contenedor. Si hay múltiples secuencias de bits de medios codificados, tales como un flujo de audio y un flujo de vídeo, asociados entre sí, típicamente se usa un archivo 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 de grabación 1570 y transfieren la secuencia de bits de medios codificados desde el receptor 1560 directamente al decodificador 1580. En algunos sistemas, sólo la parte más reciente de la transmisión grabada, por ejemplo, la extracción más reciente de 10 minutos de la transmisión grabada, se mantiene en el almacenamiento de grabación 1570, mientras que cualquier dato anterior grabado se descarta del almacenamiento de grabación 1570.

La secuencia de bits de medios codificados puede transferirse del almacenamiento de grabación 1570 al decodificador 1580. Si hay muchas secuencias 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 si una única secuencia de bits de medios se encapsula en un archivo contenedor, p. ej., para un acceso más fácil, un analizador de archivos (no se muestra en la figura) se usa para desencapsular cada secuencia de bits de medios codificados desde el archivo contenedor. El almacenamiento de grabación 1570 o un decodificador 1580 puede comprender el analizador de archivos, o el analizador de archivos está unido ya sea al almacenamiento de grabación 1570 o al decodificador

1580. También debe tenerse en cuenta que el sistema puede incluir muchos decodificadores, pero aquí solo se describe un decodificador 1570 para simplificar la descripción sin una falta de generalidad

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

Se puede configurar un emisor 1540 y/o una puerta de enlace 1550 para realizar la conmutación entre diferentes secuencias de bits 901, 902, 909, 911, p. ej., para adaptación de velocidad de bits, formato de croma, profundidad de bits de muestra, gama de colores, y/o resolución espacial, y/o un emisor 1540 y/o una puerta de enlace 1550 puede configurarse para seleccionar la secuencia de bits transmitida. La conmutación entre diferentes secuencias de bits 901, 902, 909, 911 puede darse debido a múltiples razones, tales como responder a solicitudes del receptor 1560 o condiciones predominantes, tal como el rendimiento, de la red a través de la cual se transporta la secuencia de bits. Una solicitud del receptor puede ser, p. ej., una solicitud para un segmento o un subsegmento de una representación diferente o secuencia de bits que la anterior, una solicitud para un cambio de capas y/o subcapas de escalabilidad transmitidas, o un cambio de un dispositivo de renderización que tiene diferentes capacidades en comparación con la anterior. Una solicitud de un segmento puede ser una solicitud HTTP GET. Una solicitud de un subsegmento puede ser una solicitud HTTP GET con un rango de bytes. Adicional o alternativamente, el ajuste de velocidad de bits o la adaptación de velocidad de bits puede usarse, por ejemplo, para proporcionar el denominado arranque rápido en servicios de transmisión, donde la velocidad de bits del flujo transmitido es menor que la velocidad de bits del canal después de iniciar o acceder aleatoriamente a la transmisión para iniciar la reproducción inmediatamente y para lograr un nivel de ocupancia de memoria intermedia que tolera retrasos ocasionales y/o retransmisiones de paquetes. La adaptación de la velocidad de bits puede incluir una representación múltiple o una conmutación ascendente de capas y una representación u operaciones de conmutación descendente de capas que se dan en varios órdenes.

Un decodificador 1580 puede configurarse para realizar la conmutación entre diferentes secuencias de bits 901, 902, 909, 911, p. ej., para adaptación de velocidad de bits, formato de croma, profundidad de bits de muestra, gama de colores, y/o resolución espacial, y/o un decodificador 1580 puede configurarse para seleccionar la secuencia de bits transmitida. La conmutación entre diferentes secuencias de bits 901, 902, 909, 911 puede darse debido a múltiples razones, tal como para lograr una operación de decodificación más rápida o para adaptar la secuencia de bits transmitida, p. ej., en términos de velocidad de bits, a condiciones predominantes, tales como el rendimiento, de la red a través de la cual se transporta la secuencia de bits. Una operación de decodificación más rápida podría ser necesaria, por ejemplo, si el dispositivo que incluye el decodificador 580 es multitarea y usa recursos informáticos para otros fines distintos de la decodificación de la secuencia de bits de vídeo escalable. 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 velocidad de reproducción convencional en tiempo real. La velocidad de la operación del decodificador se puede cambiar durante la decodificación o la reproducción, por ejemplo, como respuesta al cambio de una reproducción rápida a una reproducción normal o viceversa, y en consecuencia pueden tener lugar múltiples operaciones de conmutación de capas y de conmutación de capas en varios órdenes.

En lo anterior, se han descrito algunas realizaciones en relación con los términos representación y/o secuencia de bits. Debe entenderse que las realizaciones pueden describirse de manera similar en relación con términos similares, p. ej., una representación en lugar de una secuencia de bits, una secuencia de bits en lugar de una representación, o un flujo elemental en lugar de una secuencia de bits o una representación.

En lo anterior, se han descrito algunas realizaciones en relación con las imágenes de salto. Una motivación de usar imágenes de salto es limitar los requisitos de procesamiento para una unidad de acceso. Debe entenderse que las realizaciones pueden realizarse de manera similar reemplazando el uso de imágenes de salto con el uso de cualquier imagen IRAP de capa de mejora. En otras palabras, algunas o todas las imágenes IRAP de capa mejorada para la representación en baja resolución en realizaciones relacionadas con RARC pueden incluir información de error de predicción codificada. Es decir, los codificadores pueden codificar tales imágenes IRAP de capa de mejora; de manera que pueden incluir información de error de predicción codificada, y los decodificadores pueden decodificar tales imágenes IRAP de capa de mejora de manera que también se decodifica la información de error de predicción.

En lo anterior, algunas realizaciones se han descrito en relación con una representación o secuencia de bits en baja resolución y a una representación o secuencia de bits en alta resolución, lo que indica que la resolución espacial de las representaciones difiere y en consecuencia ese remuestreo en términos de resolución espacial se usa en las realizaciones. Debe entenderse que además o en lugar de diferente resolución espacial, las representaciones pueden tener otros tipos de relación o relaciones y pueden requerir otro tipo de proceso o procesos de remuestreo. Por ejemplo, la profundidad de bits y/o la gama de colores de las representaciones pueden diferir, y el remuestreo similar al usado en la escalabilidad de la gama de colores SHVC puede usarse en las realizaciones. Las realizaciones no se limitan a un tipo de

remuestreo, pero, por ejemplo, remuestreos en términos de espacio, profundidad de bits, y gama de colores pueden aplicarse juntos. Por ejemplo, en realizaciones para RARC, las imágenes de salto pueden remuestrearse en términos de resolución espacial, profundidad de bits y gama de colores, p. ej., usando el procesamiento entre capas SHVC.

En lo anterior, se han descrito algunas realizaciones en relación con la representación de baja resolución que se codifica usando GOP cerrados. Una motivación para usar GOP cerrados en la representación en baja resolución es hacer que la representación de conmutación de bajada funcione a una velocidad de imagen completa sin ningún cambio en el proceso de decodificación estandarizado. Sin embargo, debe entenderse que las realizaciones podrían realizarse de manera similar cuando la representación en baja resolución se codifica usando GOP abiertos.

En lo anterior, se han descrito algunas realizaciones en relación con el muestreo ascendente de una imagen de una representación en baja resolución para conmutar a una representación en alta resolución. Debe entenderse que las realizaciones pueden realizarse de manera similar para submuestrear una imagen de una representación en alta resolución para actuar como una imagen de referencia para decodificar una representación en baja resolución que se conmuta. Por ejemplo, se puede usar imágenes CRA (o generalmente las imágenes de los SAP Tipo 3) en la representación en baja resolución y, por lo tanto, la decodificación de imágenes RASE asociadas con la imagen CRA inicial de una representación en baja resolución a la que se conmuta puede requerir imágenes de referencia que anteceden, en orden de decodificación, a la imagen CRA. Estas imágenes de referencia pueden generarse remuestreando las imágenes respectivas de la representación en alta resolución. También debe entenderse que las realizaciones pueden realizarse de manera similar cuando el orden de representaciones o capas es de tal manera que una representación o capa más alta puede requerir el submuestreo de una imagen de una representación o capa más baja para servir como imagen de referencia. En algunos casos, el remuestreo para generar una imagen de referencia puede requerir operaciones a lo largo de más de una dimensión o tipo de escalabilidad. Por ejemplo, una representación o capa más alta puede tener una resolución espacial más grande pero una menor profundidad de bits que una representación o capa más baja y por lo tanto el uso de una imagen de la representación o capa más baja como imagen de referencia para la representación o capa más alta puede requerir muestreo ascendente en términos de resolución espacial pero submuestreo o escalado en términos de profundidad de bits.

En lo anterior, algunas realizaciones se han descrito en relación con dos representaciones o secuencias de bits o capas, tales como una representación o secuencia de bits en baja resolución y una representación o secuencia de bits en alta resolución. Debe entenderse que las realizaciones no se limitan a dos representaciones o secuencias de bits o capas, sino que se aplican igualmente a un mayor número de representaciones o secuencias de bits o capas.

En lo anterior, se han descrito algunas realizaciones en relación con el remuestreo de una única imagen para un punto de conmutación de representación. Debe entenderse que las realizaciones no se limitan a una sola imagen que se está remuestreando por un punto de conmutación de representación, sino que se aplica generalmente a cualquier número de imágenes que se remuestrean. Por ejemplo, en RARC más de una imagen de salto pueden estar presentes para un punto de conmutación. El número de imágenes remuestreadas puede determinarse por el número de imágenes de referencia que se requieren para decodificar las imágenes RASL de la imagen IRAP del punto de conmutación de la representación de conmutación y anteceder en orden de decodificación a la imagen IRAP del punto de conmutación.

En lo anterior, algunas realizaciones se han descrito en relación con los segmentos. Debe entenderse que las realizaciones pueden realizarse de manera similar en relación con los subsegmentos.

En lo anterior, algunas realizaciones se han descrito en relación con solicitudes de HTTP y/o HTTP GET. Debe entenderse que las realizaciones no se limitan al uso de HTTP, sino que pueden usarse adicionalmente o en su lugar otros protocolos, tales como WebSockets. También debe entenderse que la invención puede aplicarse en casos en los que un cliente no realiza solicitudes, tales como HTTP GET, p. ej., cuando un cliente recibe datos a través de una difusión, tal como un servicio de difusión según el Servicio de Difusión/Multidifusión Multimedia 3GPP (MBMS).

En lo anterior, algunas realizaciones se han descrito en relación con indicaciones de codificación, elementos de sintaxis, y/o estructuras de sintaxis en una secuencia de bits o en una secuencia de vídeo codificada y/o indicaciones de decodificación, elementos de sintaxis, y/o estructuras de sintaxis de una secuencia de bits o de una secuencia de vídeo codificada. Sin embargo, debe entenderse que las realizaciones podrían realizarse al codificar indicaciones, elementos sintácticos, y/o estructuras sintácticas en una estructura sintáctica o una unidad de datos que es externa a partir de una secuencia de bits o una secuencia de vídeo codificada que comprende datos de capa de codificación de vídeo, tales como fragmentos codificados, y/o indicaciones de decodificación, elementos sintácticos, y/o estructuras sintácticas de una estructura sintáctica o una unidad de datos que es externa a una secuencia de bits o una secuencia de vídeo codificada que comprende datos de capa de codificación de vídeo, tales como fragmentos codificados.

En lo anterior, donde las realizaciones ilustrativas se han descrito con referencia a un codificador, debe entenderse que la secuencia de bits resultante y el decodificador pueden tener elementos correspondientes en estos. Del mismo modo, cuando las realizaciones ilustrativas se han descrito con referencia a un decodificador,

debe entenderse que el codificador puede tener una estructura y/o un programa informático para generar la secuencia de bits que será decodificada por el decodificador.

Las realizaciones de la invención descritas anteriormente describen el códec en términos de un aparato codificador y decodificador por separado para ayudar a la comprensión de los procesos implicados. Sin embargo, se apreciará que el aparato, las estructuras y las operaciones pueden implementarse como un único aparato codificador-decodificador/estructura/operación. 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 operan 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 cableadas.

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

Además, los elementos de una red móvil terrestre pública (PLMN) también pueden comprender códecs de vídeo como se describió anteriormente.

En general, las diversas realizaciones de la invención pueden implementarse en hardware o circuitos con fines específicos, software, lógica o cualquier combinación de estos. Por ejemplo, algunos aspectos pueden implementarse en hardware, mientras que otros aspectos pueden implementarse en firmware o software que puede ejecutarse mediante un controlador, microprocesador u otro dispositivo informático, aunque la invención no se limita a estos. Aunque diversos aspectos de la invención pueden ilustrarse y describirse como diagramas de bloques, diagramas de flujo, o usando alguna otra representación pictórica, se entiende bien que estos bloques, aparatos, sistemas, técnicas o métodos descritos en la presente descripción pueden implementarse, como ejemplos no limitantes, en hardware, software, firmware, circuitos o lógica con fines específicos, hardware o controlador con fines generales u otros dispositivos informáticos, o alguna combinación de estos.

Las realizaciones de esta invención pueden implementarse mediante software informático ejecutable por un procesador de datos del dispositivo móvil, tal como en la entidad procesadora, o por hardware, o por una combinación de software y hardware. Además, en este sentido, se debe tener en cuenta que cualquier bloque del flujo lógico como en las figuras puede representar etapas del programa, o circuitos lógicos, bloques y funciones interconectados, o una combinación de pasos de programa y circuitos lógicos, bloques y funciones. El software puede almacenarse en tales medios físicos 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 variantes de datos de estos, 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, tales como dispositivos de memoria basados en semiconductores, dispositivos y sistemas de memoria magnética, dispositivos y sistemas de memoria óptica, memoria fija y memoria extraíble. Los procesadores de datos pueden ser de cualquier tipo adecuado para el entorno técnico local 5, y pueden incluir uno o más de los ordenadores con fines generales, ordenadores con fines especiales, microprocesadores, procesadores de señal digital (DSP) y procesadores basados en la arquitectura del 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 su mayoría un proceso altamente automatizado.) Las herramientas de software complejas y potentes están disponibles para convertir un diseño de nivel lógico en un diseño de circuito semiconductor listo para ser grabado y formado 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 los conductores y localizan componentes en un chip semiconductor usando reglas de diseño bien establecidas, así como bibliotecas de módulos de diseño 5 previamente almacenados. Una vez que se ha completado el diseño para un circuito semiconductor, el diseño resultante, en un formato electrónico estandarizado (p. ej., Opus, GDSII, o lo similar) puede transmitirse a una instalación de fabricación de semiconductores o "fab" para la fabricación.

La descripción anterior se ha proporcionado a manera de ejemplo ilustrativo y no limitativo, una descripción completa e informativa de la realización ilustrativa de esta invención. La invención se expone en el conjunto de reivindicaciones anexas.

REIVINDICACIONES

1. Un método que comprende:

5 decodificar un primer conjunto (904) de imágenes de una primera representación de vídeo codificada (901) incluyendo una imagen de punto de acceso intraaleatorio;
 procesar una o más imágenes decodificadas del primer conjunto (904) de imágenes de la primera
 representación de vídeo codificada (901) en una o más imágenes decodificadas procesadas de la
 10 primera representación de vídeo codificada (901), en donde el procesamiento comprende uno o ambos
 del remuestreo y escala de valor de muestra del primer conjunto (904) de imágenes de la primera
 representación de vídeo codificada (901);
caracterizado por que el método comprende además:

15 recibir y analizar información sobre imágenes de decodificación de una segunda
 representación de vídeo codificada (902),
 en donde la segunda representación de vídeo codificada (902) difiere de la primera
 representación de vídeo codificada (901) en uno o más de lo siguiente: formato de croma,
 profundidad de bits de muestra, gama de colores, resolución espacial, y la información
 20 indica si la segunda representación de vídeo codificada (902) es decodificable usando una o
 más imágenes del punto de acceso intraaleatorio codificadas por salto (910) codificado en la
 primera representación de vídeo codificada (901) y que contiene información que se usará
 como imágenes de referencia para la predicción en la decodificación de la segunda
 presentación de vídeo codificada;
 25 siempre que la información sobre imágenes de decodificación de la segunda
 representación de vídeo codificada (902) indica que la segunda representación de vídeo
 codificada (902) es decodificable usando dicha una o más imágenes del punto de acceso
 intraaleatorio codificadas por salto (910) como una o más imágenes de referencia para la
 predicción, el método comprende además:

30 decodificar dicha una o más imágenes de punto de acceso intraaleatorio
 codificadas por salto de la primera representación de vídeo codificada (901) para
 usar como una o más imágenes de referencia para la predicción en la
 decodificación de la segunda presentación de vídeo codificada;
 35 conmutar para decodificar la segunda representación de vídeo codificada (902)
 comenzando desde un segundo conjunto (908) de imágenes de la segunda
 representación de vídeo codificada (902), el segundo conjunto (908) sigue al
 primer conjunto (904) de la primera representación de vídeo codificada (901) en
 el orden de salida de imagen; y
 40 decodificar al menos el segundo conjunto (908) de imágenes de la segunda
 representación de vídeo (902) usando dicha una o más imágenes de punto de
 acceso intraaleatorio codificadas por salto como una o más imágenes de
 referencia para la predicción.

2. El método según la reivindicación 1 que comprende además:
 45 recibir una descripción de presentación de medios que comprende dicha información.

3. El método según la reivindicación 2 que comprende además:

50 determinar, en base a la descripción de presentación de medios, un primer identificador que
 identifica el primer conjunto (904) de la primera representación de vídeo codificada (901) y un
 segundo identificador que identifica el segundo conjunto (908) de la segunda representación de
 vídeo codificada (902);
 solicitar el primer conjunto (904) usando el primer identificador;
 recibir el primer conjunto (904);
 55 solicitar el segundo conjunto (908) usando el segundo identificador; y
 recibir el segundo conjunto (908).

4. El método según la reivindicación 3, en donde el primer identificador y el segundo identificador son
 60 identificadores de recursos uniformes.

5. El procedimiento según cualquiera de las reivindicaciones 1 a 4, en donde la resolución espacial de la
 primera representación de vídeo codificada (901) se determina para una región de referencia y la
 resolución espacial de la segunda representación de vídeo codificada (902) se determina para una región
 pronosticada, en donde el método comprende además:

65

recibir información sobre la región de referencia, o de no recibir información sobre la región de referencia, determinar que la región de referencia es un área completa de una imagen de la primera representación de vídeo codificada (901);

recibir información sobre la región pronosticada, o de no recibir información sobre la región pronosticada,

determinar que la región pronosticada es un área completa de una imagen de la segunda representación de vídeo codificada (902).

6. El método según cualquiera de las reivindicaciones 1 a 5, en donde dicha una o más imágenes decodificadas procesadas tienen el mismo formato de croma y profundidad de bits que las imágenes de la segunda representación de vídeo codificada (902).

7. El método según cualquiera de las reivindicaciones 1 a 6 que comprende además:

recibir y analizar información que la segunda representación de vídeo codificada (902) pertenece a una capa no base y se puede decodificar de forma independiente.

8. El método según cualquiera de las reivindicaciones 1 a 7 que comprende además: recibir y analizar información de que el primer conjunto (904) de la primera representación codificada (902) y el segundo conjunto (908) de la segunda representación codificada (902) forman una secuencia de bits que se ajusta a un perfil de codificación de vídeo escalable con limitaciones para que las unidades de acceso comprendan no más de dos imágenes y que, cuando una unidad de acceso comprende dos imágenes, la imagen que pertenece a la capa más alta es una imagen del punto de acceso intraaleatorio, que no usa imágenes anteriores en la misma capa como una referencia para la predicción sino que puede usar una imagen en la capa más baja en la misma unidad de acceso como una referencia para la predicción.

9. Un aparato (550) para decodificar vídeo, que comprende:

medios para decodificar un primer conjunto (904) de imágenes de una primera representación de vídeo codificada (901) incluyendo una imagen de punto de acceso intraaleatorio;

medios para procesar una o más imágenes decodificadas del primer conjunto en una o más imágenes decodificadas procesadas de la primera representación de vídeo codificada (901), en donde el procesamiento comprende uno o ambos de remuestreo y escala del valor de muestra del primer conjunto (904) de imágenes de la primera representación de vídeo codificada (901);

caracterizado por que el aparato (550) comprende además:

medios para recibir y analizar información sobre la decodificación de una segunda representación de vídeo codificada (902),

en donde la segunda representación de vídeo codificada (902) difiere de la primera representación de vídeo codificada (901) en uno o más de lo siguiente: formato de croma, profundidad de bits de muestra, gama de colores, resolución espacial, y la información indica si la segunda representación de vídeo codificada (902) es decodificable usando una o más imágenes de punto de acceso aleatorio codificadas por salto (910) codificado en la primera representación de vídeo codificada (901) y que contiene información que se usará como una o más imágenes de referencia para la predicción en la decodificación de la segunda presentación de vídeo codificada;

medios para decodificar dicha una o más imágenes de punto de acceso intraaleatorio codificadas por salto de la primera representación de vídeo codificada (901) para usar como una o más imágenes de referencia para la predicción en la decodificación de la segunda representación de vídeo codificada, siempre que la información sobre decodificación de imágenes de la segunda representación de vídeo codificada (902) indica que la segunda representación de vídeo codificada (902) es decodificable usando dicha una o más imágenes de punto de acceso intraaleatorio codificadas por salto (910) como una o más imágenes de referencia para la predicción;

medios de conmutación para decodificar la segunda representación de vídeo codificado (902) a partir de un segundo conjunto (908) de imágenes de la segunda representación de vídeo codificado (902), el segundo conjunto (908) sigue al primer conjunto (904) de la primera representación de vídeo codificado (901) en el orden de salida de imágenes siempre que la información indique que la segunda representación de vídeo codificado (902) puede decodificarse usando dicha una o más imágenes de punto de acceso intraaleatorio codificado por salto (910) como una o más imágenes de referencia para la predicción; y

medios para decodificar al menos el segundo conjunto (908) de las imágenes de la segunda representación de vídeo (902) usando dichas imágenes de punto de acceso intraaleatorio codificadas por salto decodificadas como una o más imágenes de referencia para la predicción.

10. El aparato (550) según la reivindicación 9, en donde el aparato (550) comprende además:
medios para recibir una descripción de presentación de medios que comprende dicha información.
- 5 11. El aparato (550) según la reivindicación 10, en donde el aparato (550) comprende además:

Medios para determinar, en base a la descripción de presentación de medios, un primer
identificador que identifica el primer conjunto de la primera representación de vídeo codificada y
un segundo identificador que identifica el segundo conjunto de la segunda representación de
10 vídeo codificada;
Medios para solicitar el primer conjunto (904) usando el primer identificador;
Medios para recibir el primer conjunto (904);
Medios para solicitar el segundo conjunto (908) usando el segundo identificador; y
Medios para recibir el segundo conjunto (908).
- 15 12. El aparato (550) según la reivindicación 11 en donde el primer identificador y el segundo identificador son
identificadores de recursos uniformes.
- 20 13. El aparato (550) según cualquiera de las reivindicaciones 9 a 12, en donde la resolución espacial de la
primera representación de vídeo codificada (901) se determina para una región de referencia y la
resolución espacial de la segunda representación de vídeo codificada (902) se determina para una región
pronosticada, en donde el aparato (550) comprende además:

Medios para recibir información sobre la región de referencia;
25 Medios para determinar que la región de referencia es un área completa de una imagen de la primera
representación de vídeo codificada (901), de no recibir información sobre la región de referencia;
Medios para recibir información sobre la región pronosticada; y
Medios para determinar que la región pronosticada es un área completa de una imagen de la segunda
representación de vídeo codificada (902), de no recibir información sobre la región pronosticada.
- 30 14. El aparato (550) según cualquiera de las reivindicaciones 9 a 13, en donde dicha una o más imágenes
decodificadas procesadas tienen el mismo formato de croma y profundidad de bits que las imágenes de
la segunda representación de vídeo codificada (902).
- 35 15. El aparato (550) según cualquiera de las reivindicaciones 9 a 14, en donde el aparato (550) comprende además:
medios para recibir y analizar información que la segunda representación de vídeo codificada (902)
pertenece a una capa no base y se puede decodificar de forma independiente.
- 40 16. El aparato (550) según cualquiera de las reivindicaciones 9 a 15, en donde el aparato (550) comprende además:
medios para recibir y analizar información que el primer conjunto (904) de la primera representación de
vídeo codificada (901) y el segundo conjunto (908) de la segunda representación de vídeo codificada
forman una secuencia de bits que se ajusta a un perfil de codificación de vídeo escalable con limitaciones
para que las unidades de acceso comprendan no más de dos imágenes y que, cuando una unidad de
45 acceso comprende dos imágenes, la imagen que pertenece a la capa más alta es una imagen del punto de
acceso intraaleatorio, que no usa imágenes anteriores en la misma capa como una referencia para la
predicción sino que puede usar una imagen en la capa más baja en la misma unidad de acceso como una
referencia para la predicción.
- 50 17. Un medio de almacenamiento legible por ordenador almacenado con código en este para su uso por un
aparato, que al ejecutarse por un procesador, permite al aparato:

decodificar un primer conjunto (904) de imágenes de una primera representación de vídeo codificada
(901) incluyendo una imagen de punto de acceso intraaleatorio;
procesar una o más imágenes decodificadas del primer conjunto (904) de imágenes de la primera
55 representación de vídeo codificada (901) en una o más imágenes decodificadas procesadas de la
primera representación de vídeo codificada (901), en donde el procesamiento comprende uno o ambos
del remuestreo y escala de valor de muestra del primer conjunto (904) de imágenes de la primera
representación de vídeo codificada (901);
caracterizado por que el medio de almacenamiento legible por ordenador comprende además
60 el código de este para su uso por el aparato, que al ejecutarse por un procesador, permite al
aparato:

recibir y analizar información sobre imágenes de decodificación de una segunda
representación de vídeo codificada (902),
65 en donde la segunda representación de vídeo codificada (902) difiere de la primera
representación de vídeo codificada (901) en uno o más de lo siguiente: formato de croma,

profundidad de bits de muestra, gama de colores, resolución espacial, y la información indica si la segunda representación de vídeo codificada (902) es decodificable usando una o más imágenes de punto de acceso aleatorio codificadas por salto (910) codificado en la primera representación de vídeo codificada (901) y que contiene información que se usará como una o más imágenes de referencia para la predicción en la decodificación de la segunda presentación de vídeo codificada;

siempre que la información sobre imágenes de decodificación de la segunda representación de vídeo codificada (902) indica que la segunda representación de vídeo codificada (902) es decodificable usando dicha una o más imágenes del punto de acceso intraaleatorio codificadas por salto (910) como una o más imágenes de referencia para la predicción, dicho código, que al ejecutarse por dicho al menos un procesador, causa adicionalmente que el aparato realice al menos:

decodificar dicha una o más imágenes de punto de acceso intraaleatorio codificadas por salto de la primera representación de vídeo codificada (901) para usar como una o más imágenes de referencia para la predicción en la decodificación de la segunda presentación de vídeo codificada;

conmutar para decodificar la segunda representación de vídeo codificada (902) comenzando desde un segundo conjunto (908) de imágenes de la segunda representación de vídeo codificada (902), el segundo conjunto (908) sigue al primer conjunto (904) de la primera representación de vídeo codificada (901) en el orden de salida de imagen; y

decodificar un segundo conjunto (908) de una segunda representación de vídeo (902) usando la dicha una o más imágenes de acceso intraaleatorio codificadas por salto como una o más imágenes de referencia para la predicción.

18. Un método que comprende:

proporcionar, en una descripción de presentación de medios, metadatos para al menos un primer conjunto de imágenes de una primera representación de vídeo codificada (901) y un segundo conjunto de imágenes de una segunda representación de vídeo codificada (902);

proporcionar, en la descripción de presentación de medios, una o más características para decodificar una secuencia de segmentos o subsegmentos consecutivos no superpuestos de la al menos primera representación de vídeo codificada (901) y la segunda representación de vídeo codificada (902);

proporcionar, en la descripción de presentación de medios, una indicación de que la primera representación de vídeo codificada (901) incluye una o más imágenes de acceso aleatorio codificadas por salto (910) que pueden usarse como una o más imágenes de referencia para la predicción en la decodificación del segundo conjunto de imágenes de la segunda representación de vídeo codificada (902); codificar dicha una o más imágenes de acceso intraaleatorio codificadas por salto en la primera representación de vídeo codificada (901) y que contiene información que se utilizará como una o más imágenes de referencia para la predicción en la decodificación de la segunda presentación de vídeo codificada;

en donde la segunda representación codificada (902) difiere de la primera representación de vídeo codificada (901) en uno o más de lo siguiente: formato de croma, profundidad de bits de muestra, gama de colores, resolución espacial.

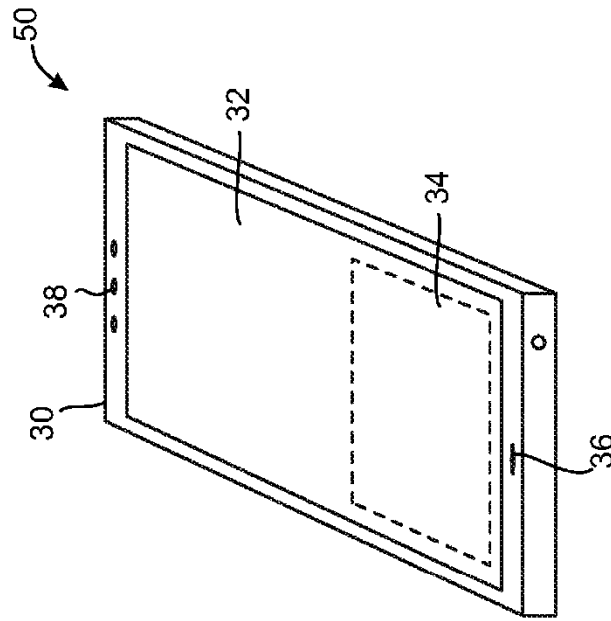


Fig. 2

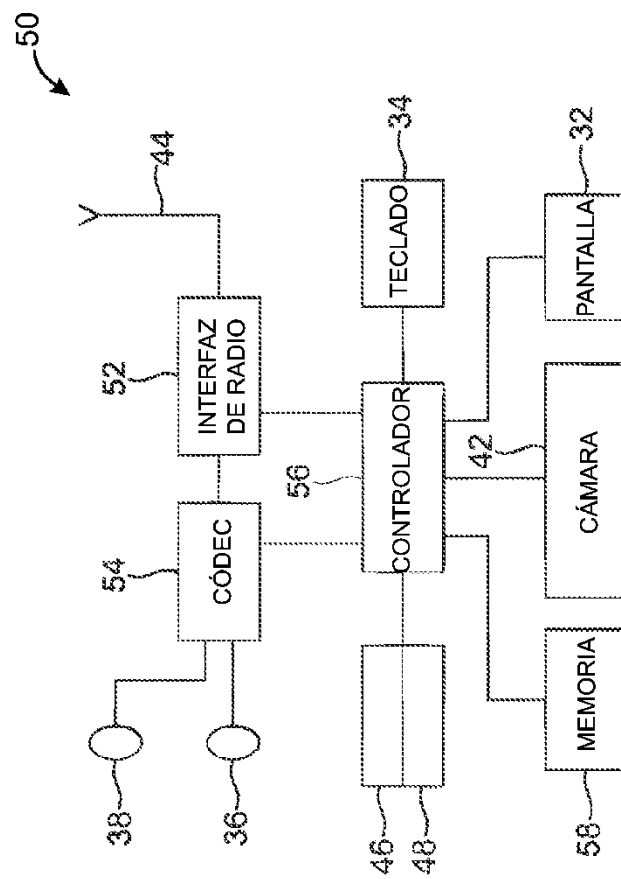


Fig. 1

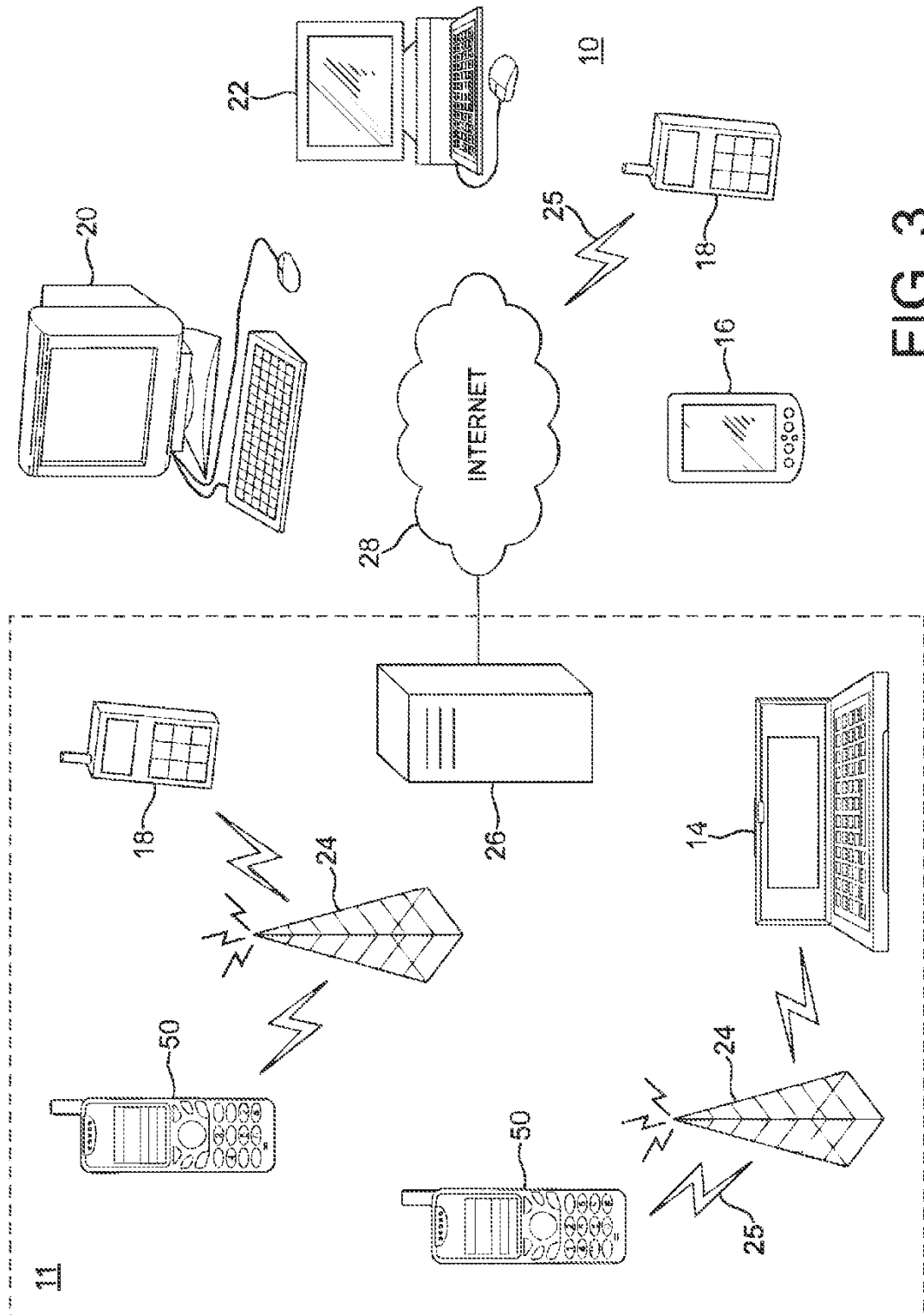


FIG. 3

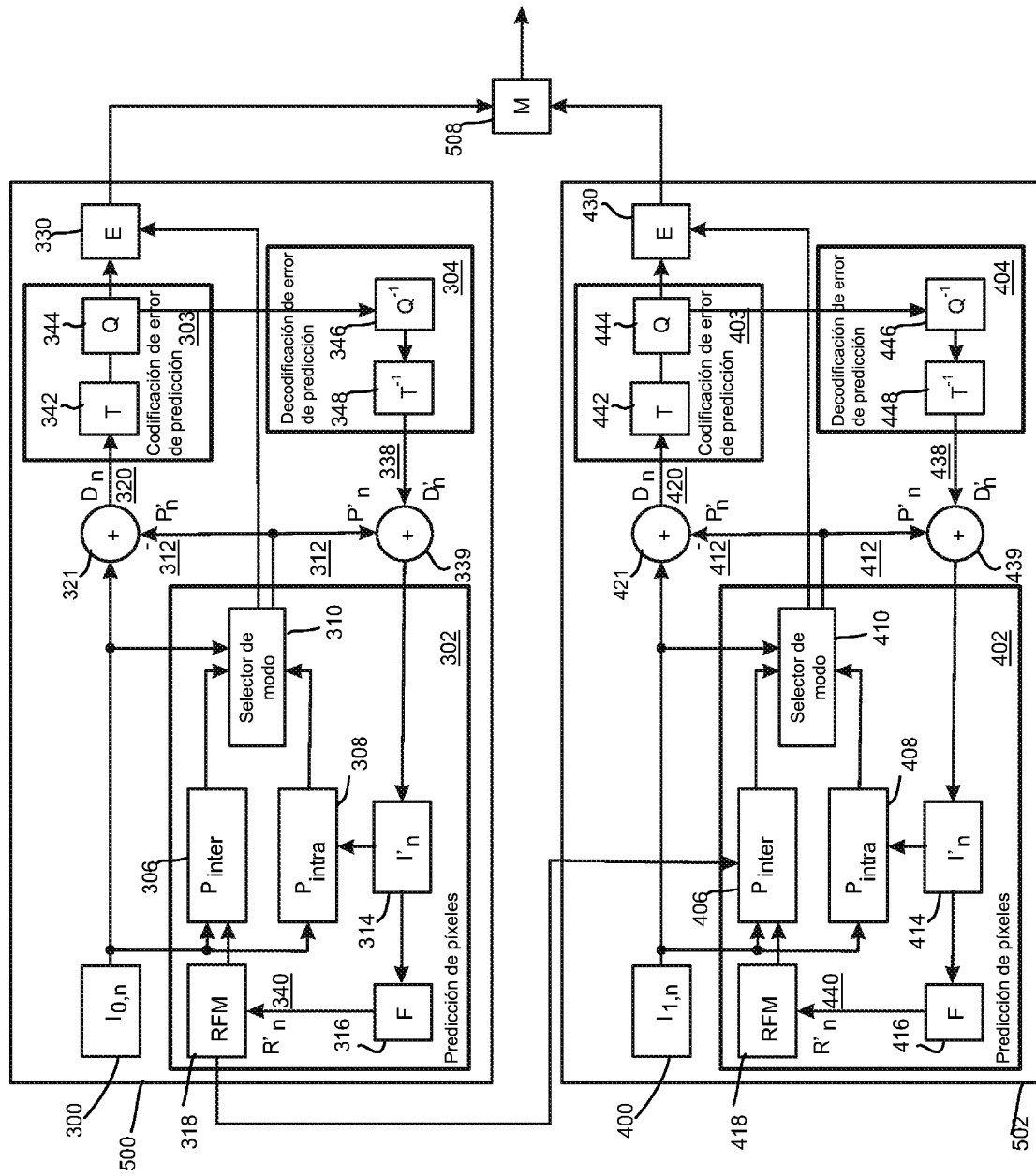


Fig. 4

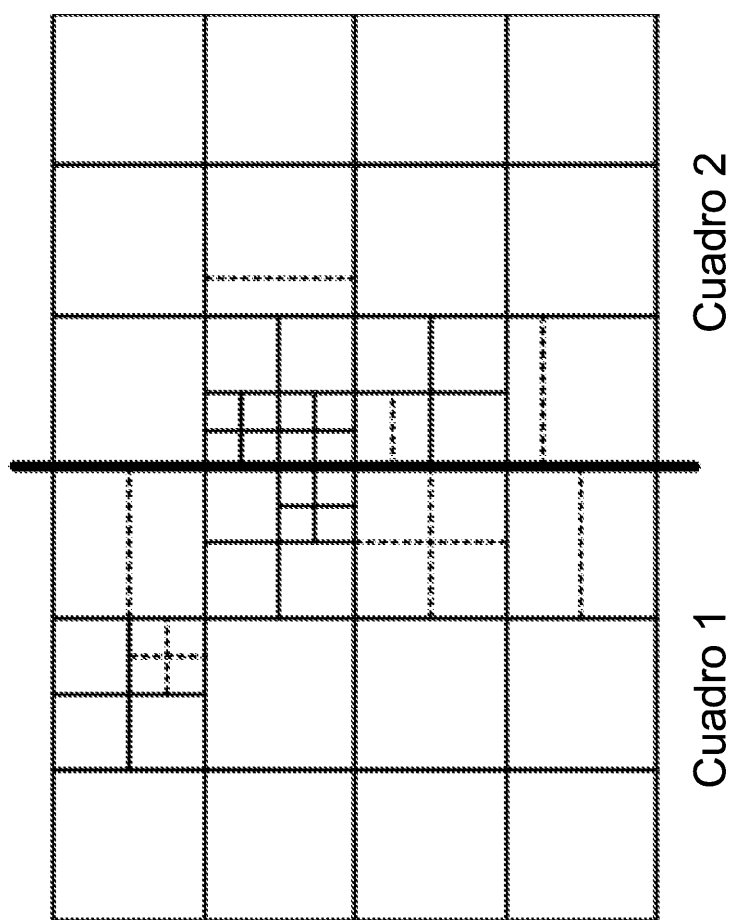


Fig. 5

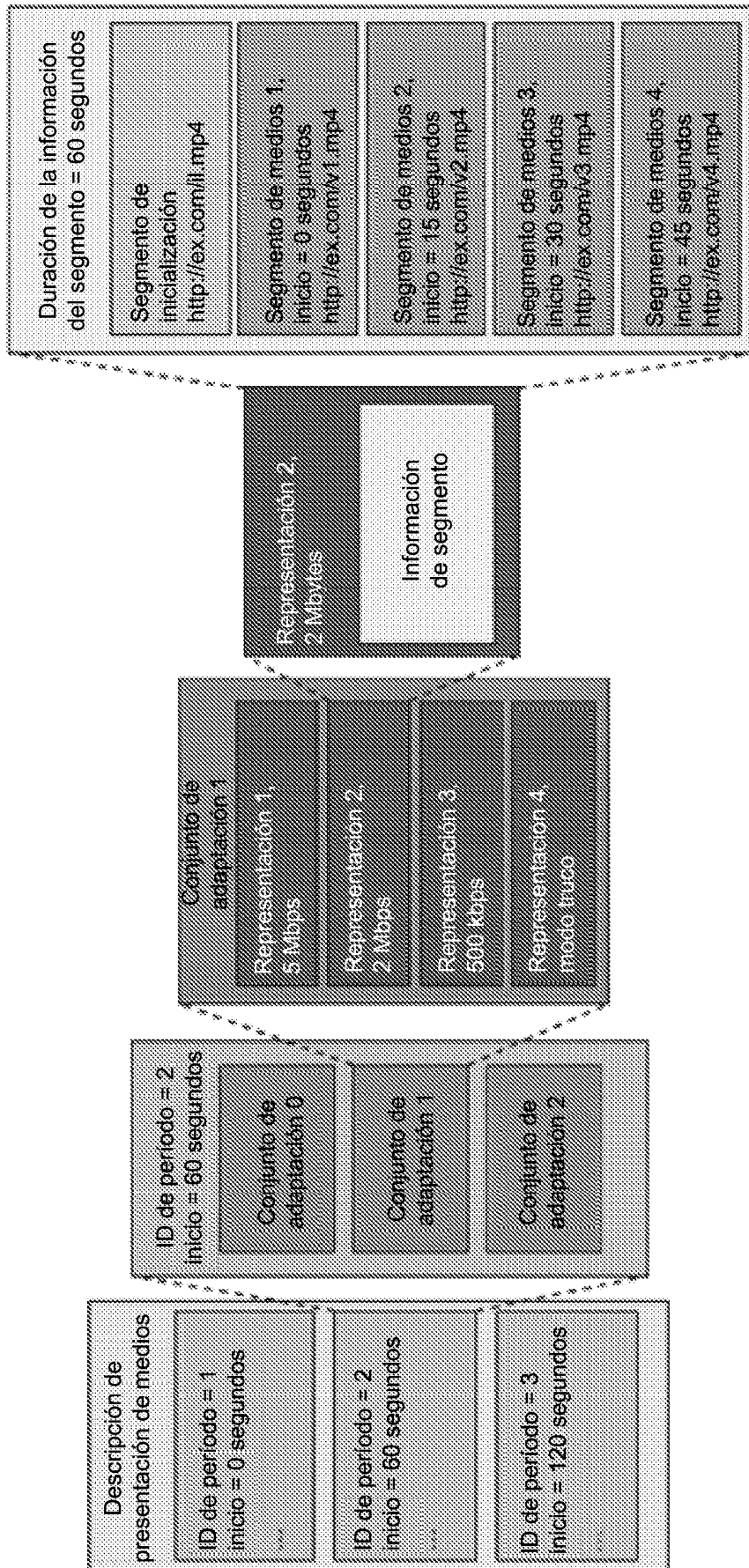


Fig. 6

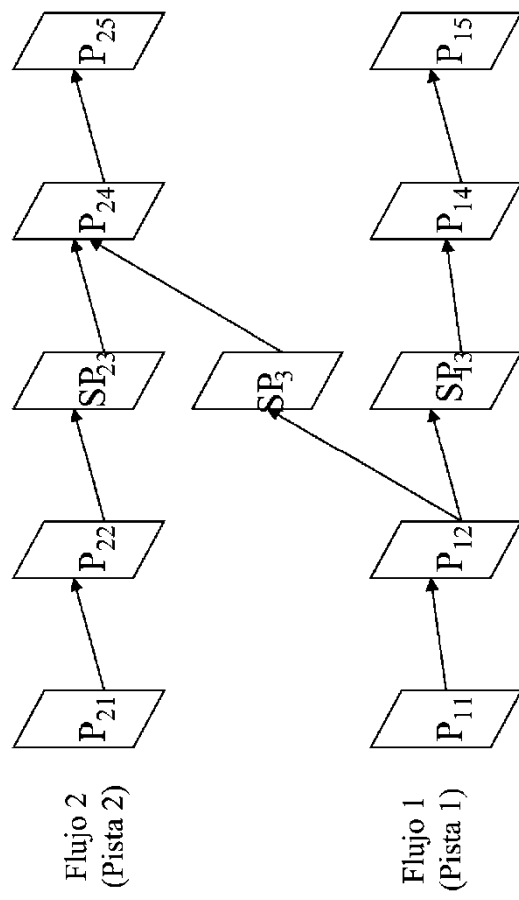


Fig. 7

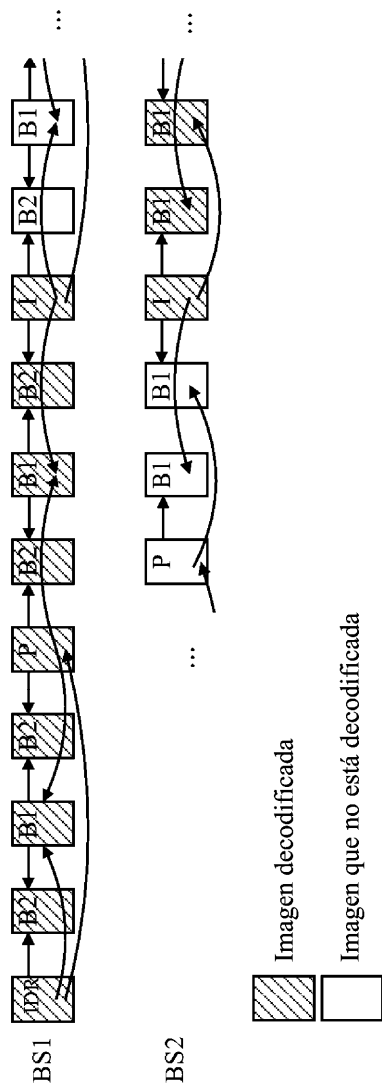


Fig. 8d

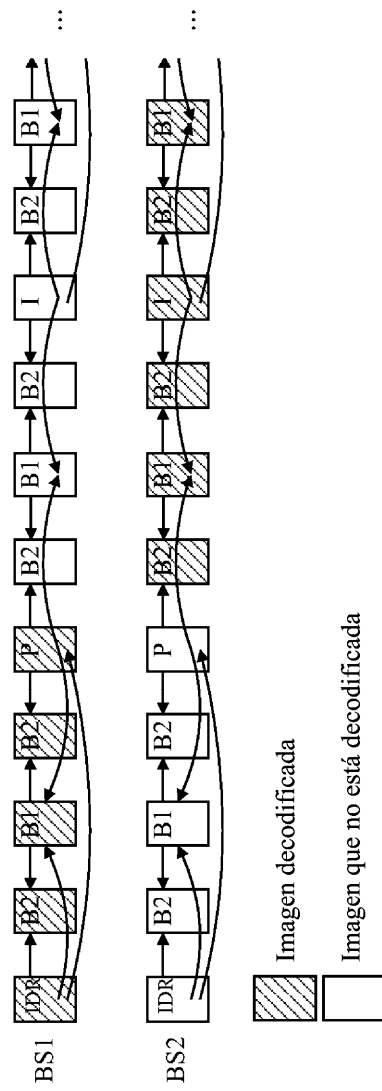


Fig. 8e

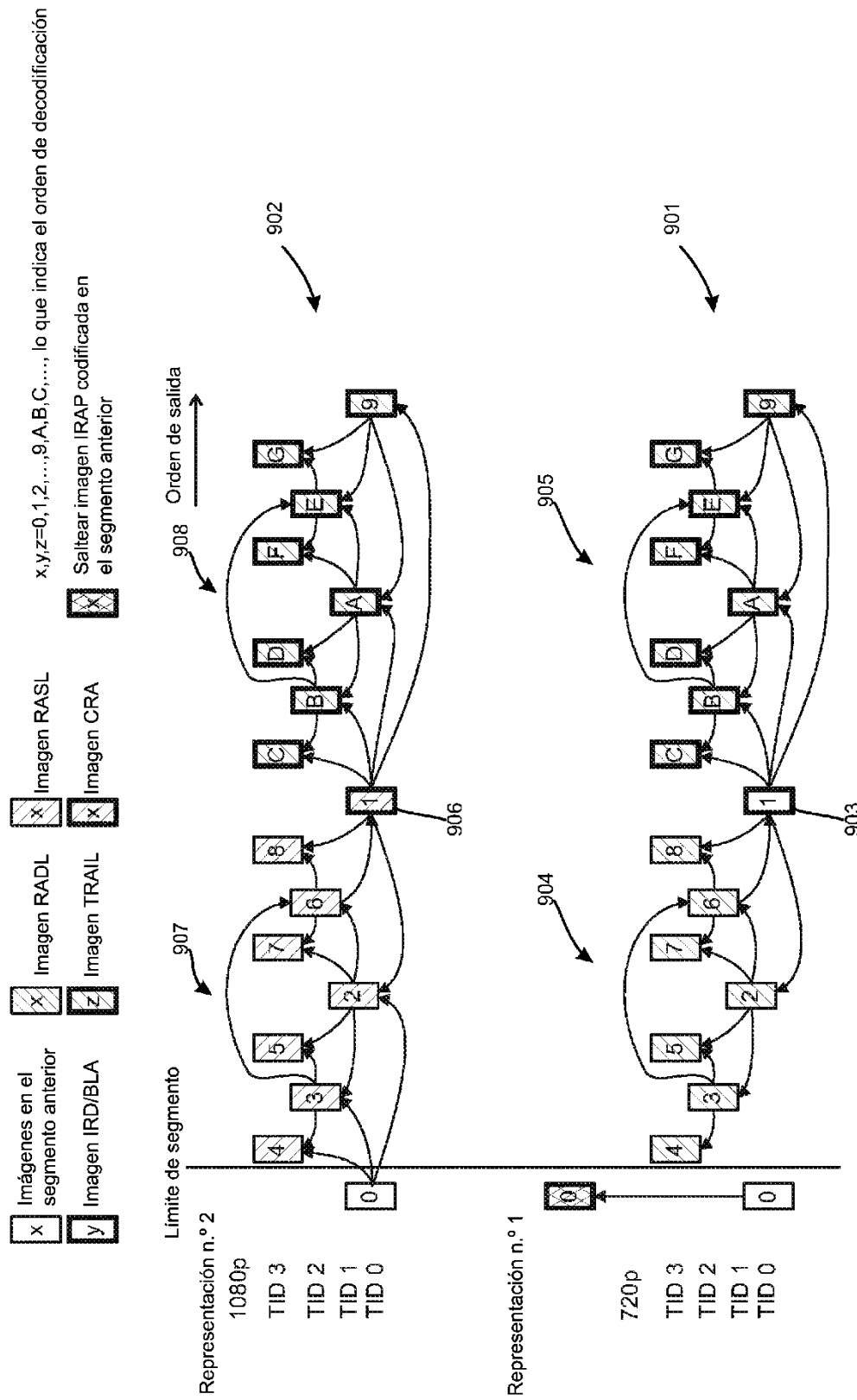


Fig. 9a

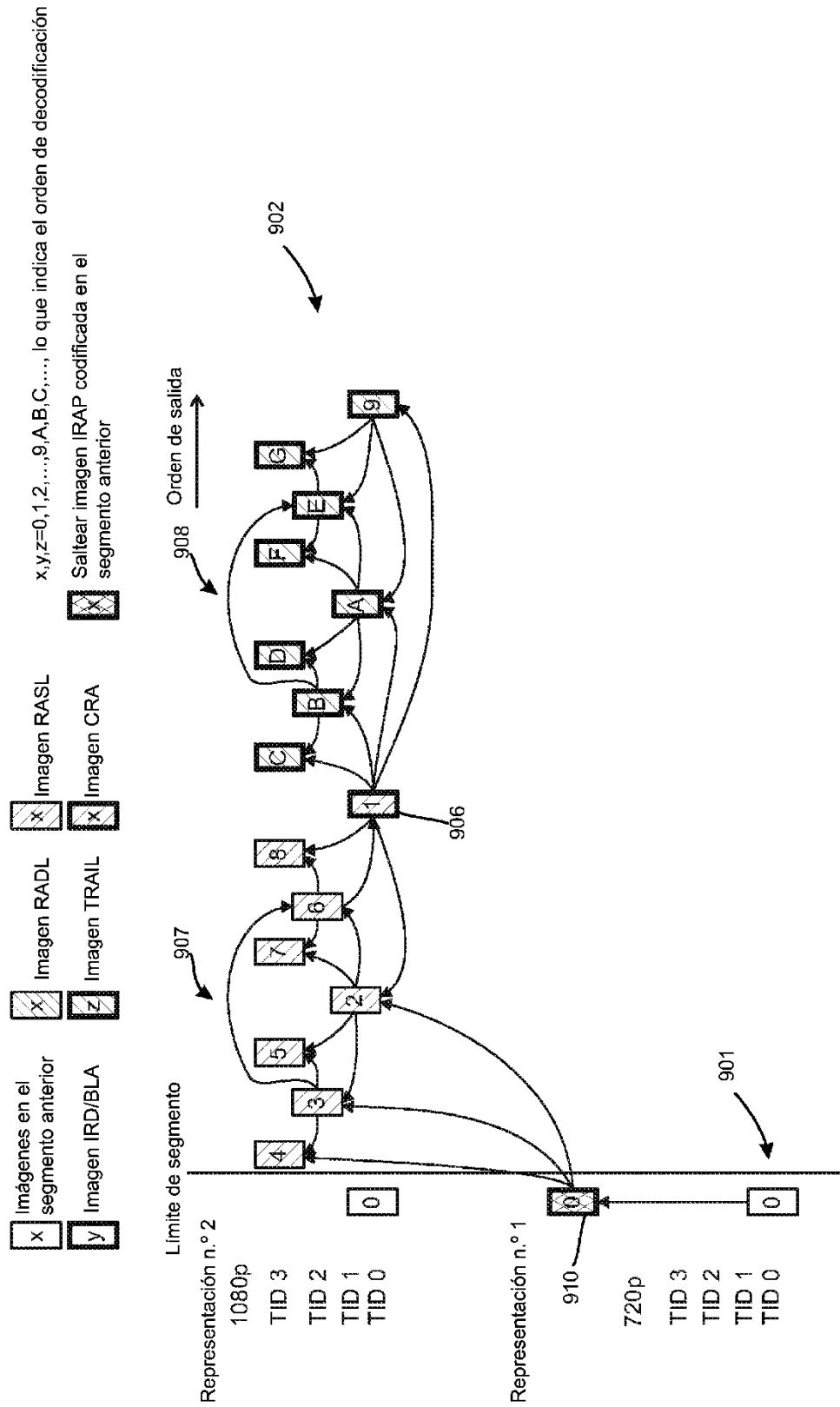


Fig. 9b

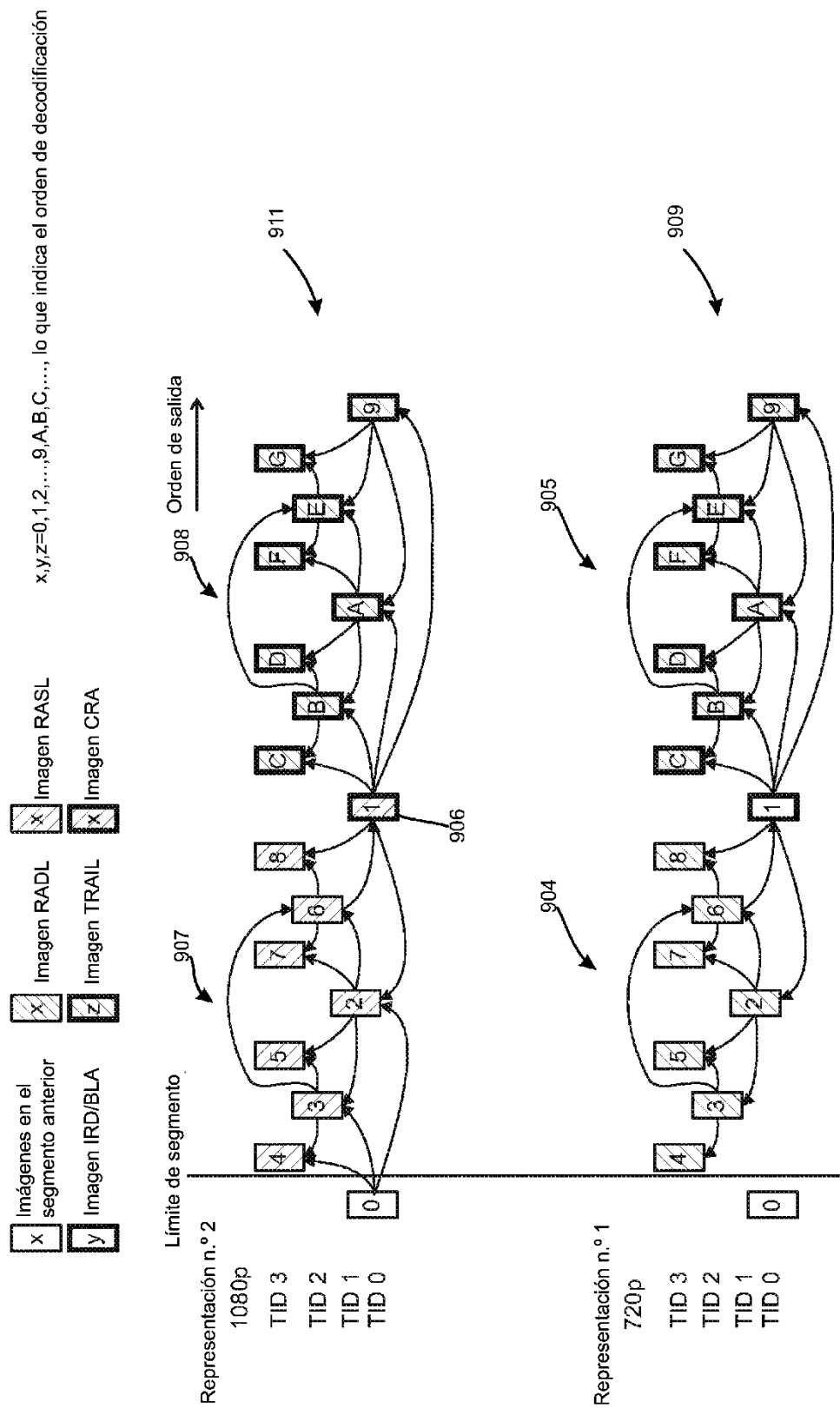


Fig. 9c

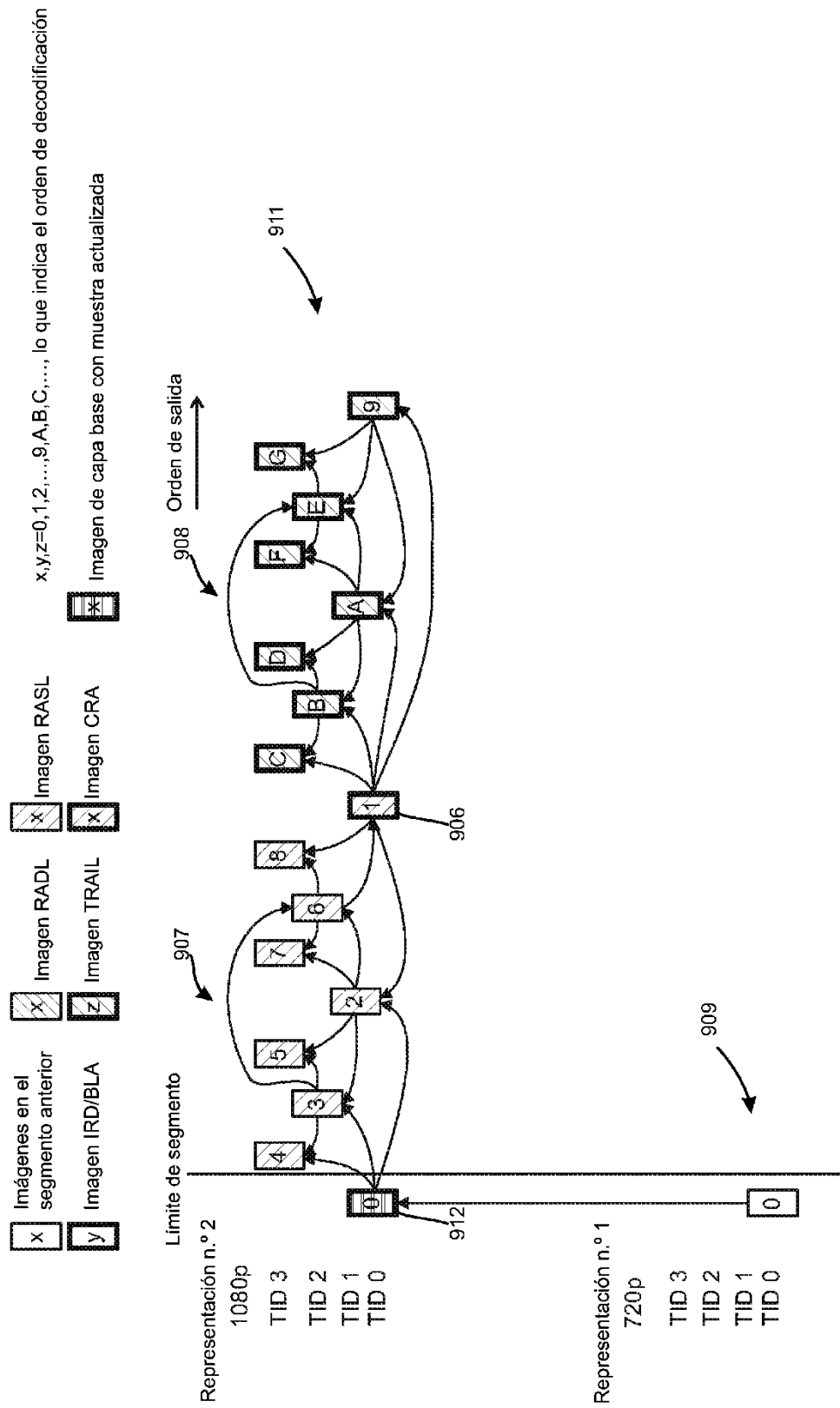


Fig. 9d

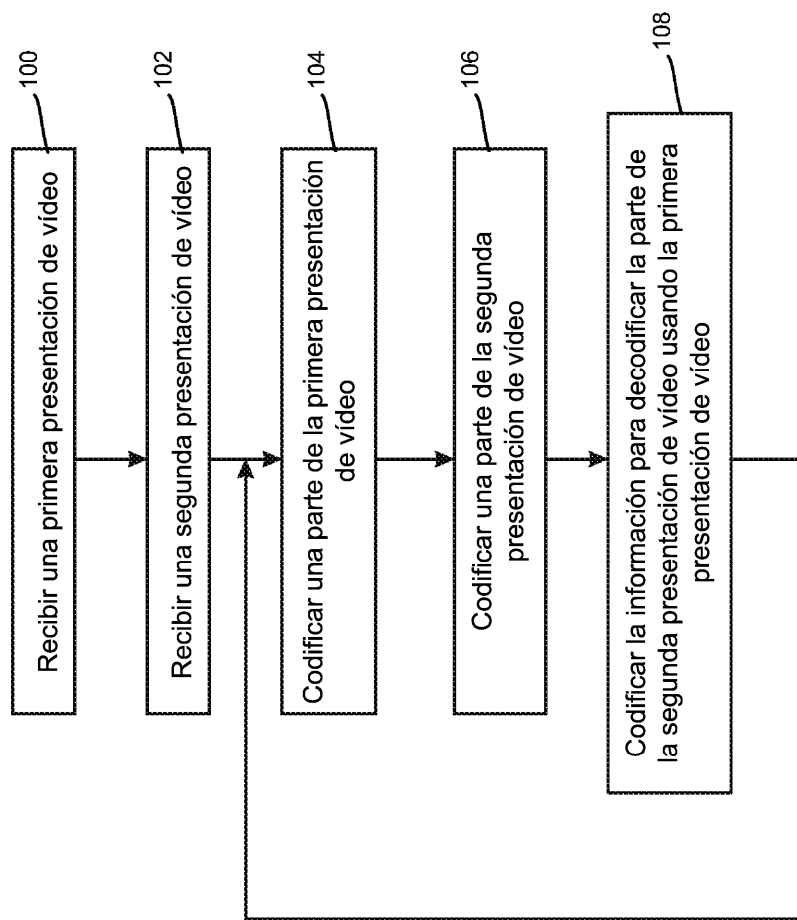


Fig. 10

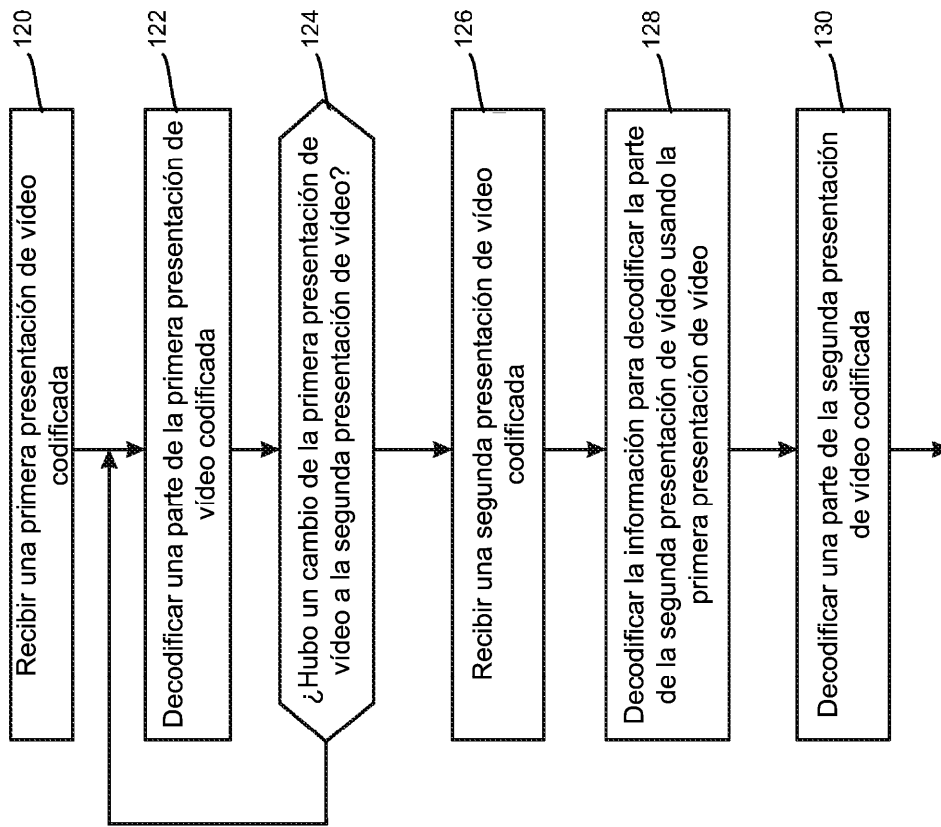


Fig. 11

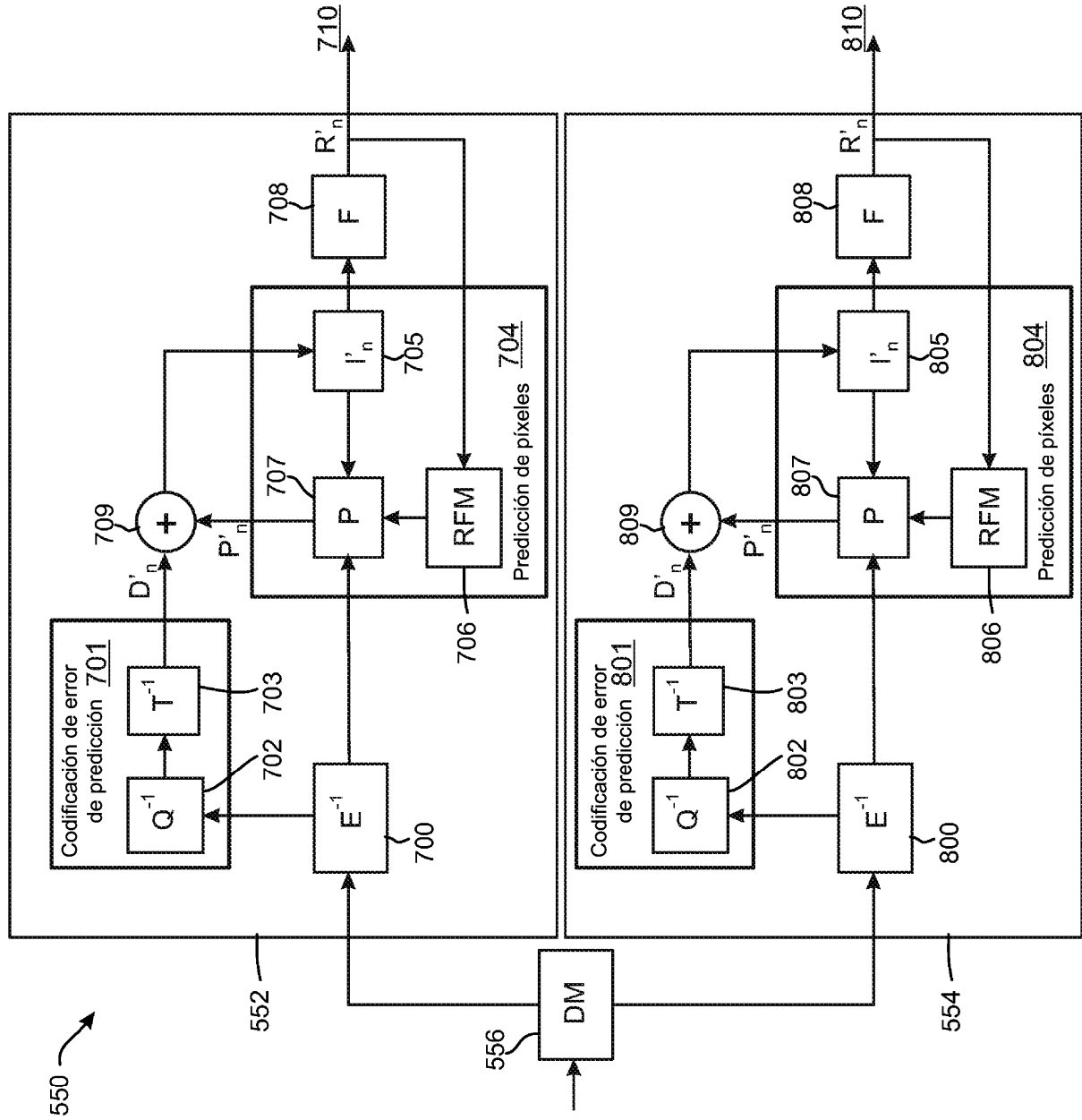


Fig. 12

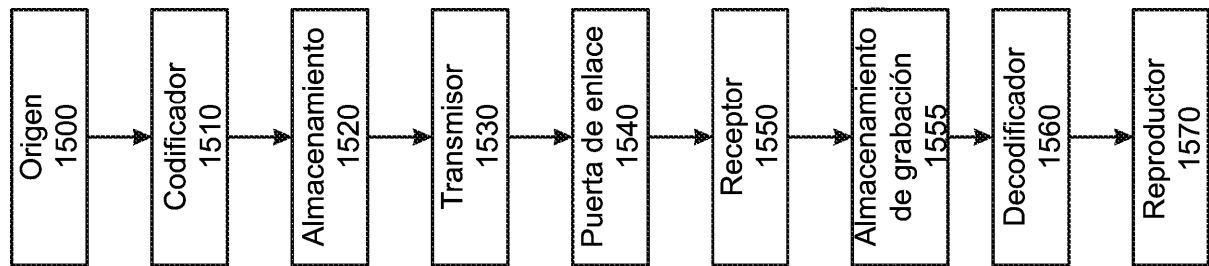


Fig. 13