

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 812 807**

51 Int. Cl.:

H04N 19/593 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **06.08.2015 PCT/US2015/044067**

87 Fecha y número de publicación internacional: **11.02.2016 WO16022835**

96 Fecha de presentación y número de la solicitud europea: **06.08.2015 E 15756712 (4)**

97 Fecha y número de publicación de la concesión europea: **20.05.2020 EP 3178230**

54 Título: **Procedimiento para codificación de vídeo con modo de predicción espacial para codificación de vídeo multimodo**

30 Prioridad:

08.08.2014 US 201462035363 P
13.04.2015 US 201562146935 P
05.08.2015 US 201514819306

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
18.03.2021

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)
5775 Morehouse Drive
San Diego, CA 92121-1714, US

72 Inventor/es:

JACOBSON, NATAN HAIM;
THIRUMALAI, VIJAYARAGHAVAN;
JOSHI, RAJAN LAXMAN y
DAI, MIN

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 812 807 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

Procedimiento para codificación de vídeo con modo de predicción espacial para codificación de vídeo multimodo

5 CAMPO TÉCNICO

[0001] Esta divulgación se refiere al campo de la codificación y compresión de vídeo y, en particular, a la compresión de vídeo para su transmisión a través de enlaces de visualización, tal como la compresión de vídeo de enlace de visualización.

10

ANTECEDENTES

[0002] Las capacidades de vídeo digital se pueden incorporar a una amplia gama de dispositivos de visualización, que incluyen televisores digitales, asistentes digitales personales (PDA), ordenadores portátiles, monitores de escritorio, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, dispositivos de videoconferencia y similares. Los enlaces de visualización se utilizan para conectar dispositivos de visualización a dispositivos de origen adecuados. Los requisitos de ancho de banda de los enlaces de visualización son proporcionales a la resolución de los dispositivos de visualización y, por lo tanto, los dispositivos de visualización de alta resolución requieren enlaces de visualización de gran ancho de banda. Algunos enlaces de visualización no tienen el ancho de banda para prestar soporte a dispositivos de visualización de alta resolución. La compresión de vídeo se puede usar para reducir los requisitos de ancho de banda, de modo que se puedan usar enlaces de visualización de menor ancho de banda para proporcionar vídeo digital a dispositivos de visualización de alta resolución.

15

20

[0003] También se ha intentado utilizar la compresión de imágenes en los datos de píxeles. Sin embargo, tales esquemas a veces no son visualmente libres de pérdidas o pueden ser difíciles y caros de implementar en dispositivos de visualización convencionales.

25

[0004] La Asociación de normas de electrónica de vídeo (VESA) ha desarrollado la compresión de flujo de visualización (DSC) como norma para la compresión de vídeo de enlace de visualización. La técnica de compresión de vídeo de enlace de visualización, tal como DSC, debería proporcionar, entre otras cosas, una calidad de imagen que no tenga pérdidas visuales (es decir, las imágenes deben tener un nivel de calidad tal que los usuarios no puedan distinguir que la compresión está activa). La técnica de compresión de vídeo de enlace de visualización también debería proporcionar un esquema que sea fácil y económico de implementar en tiempo real con hardware convencional.

30

35

El documento US 2014/0092960 A1 se refiere a un sistema que implementa control de velocidad para operaciones de codificación y decodificación, por ejemplo, operaciones realizadas en fragmentos de datos tales como datos de imágenes. El sistema implementa una transformación de completitud de memoria intermedia real a completitud controlada por velocidad. Con el modelo de completitud controlada por velocidad, los codificadores y decodificadores adaptan la asignación de bits en respuesta a la completitud controlada por velocidad, en lugar de la completitud real. La búsqueda para encontrar el mejor vector se puede realizar en la línea de muestras previa, en lugar de la línea que se está codificando actualmente. En una implementación, la búsqueda de bloques compara un conjunto de 9 muestras consecutivas con muestras de referencia utilizando varios vectores potenciales con valores que van desde -3 a -10. Las muestras actuales y las muestras de referencia que se comparan están en la misma línea de exploración, por ejemplo, la línea por encima de la línea de la muestra a codificar.

40

45

BREVE EXPLICACIÓN

50

[0005] El alcance de la invención se determina por las reivindicaciones independientes.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

55

[0006]

La **FIG. 1A** es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo de ejemplo que puede utilizar técnicas de acuerdo con aspectos descritos en esta divulgación.

60

La **FIG. 1B** es un diagrama de bloques que ilustra otro sistema de codificación y decodificación de vídeo de ejemplo que puede llevar a cabo técnicas de acuerdo con aspectos descritos en esta divulgación.

La **FIG. 2A** es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación.

65

La **FIG. 2B** es un diagrama de bloques que ilustra un ejemplo de un decodificador de vídeo que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación.

La **FIG. 3** es un diagrama de bloques que ilustra un procedimiento de codificación de una primera línea de un fragmento de acuerdo con aspectos descritos en esta divulgación.

5 La **FIG. 4** es un diagrama de bloques que ilustra otro procedimiento de codificación de una primera línea de un fragmento de acuerdo con un modo de realización de la presente invención.

10 La **FIG. 5** es un diagrama de bloques que ilustra las relaciones espaciales entre un píxel actual y los píxeles de predicción en un procedimiento de codificación de un bloque de acuerdo con los aspectos descritos en esta divulgación.

15 La **FIG. 6** es un diagrama de bloques que ilustra las relaciones espaciales entre un píxel actual y los píxeles de predicción en un procedimiento de codificación de un bloque de acuerdo con los aspectos descritos en esta divulgación.

La **FIG. 7** es un diagrama de bloques que ilustra las relaciones espaciales entre un píxel actual y los píxeles de predicción en un procedimiento de codificación de bloque de acuerdo con un modo de realización de la presente invención.

20 La **FIG. 8** es un diagrama de bloques que ilustra una estrategia de agrupación de ejemplo de acuerdo con aspectos descritos en esta divulgación.

25 La **FIG. 9** es un diagrama de bloques que ilustra un procedimiento de codificación de una línea de un fragmento de acuerdo con aspectos descritos en esta divulgación.

La **FIG. 10** es un diagrama de bloques que ilustra otro procedimiento de codificación de una primera línea de un fragmento de acuerdo con aspectos descritos en esta divulgación.

30 La **FIG. 11** es un diagrama de flujo que ilustra un procedimiento de codificación de modo de predicción espacial de acuerdo con aspectos de la presente divulgación.

DESCRIPCIÓN DETALLADA

35 **[0007]** En general, esta divulgación se refiere a procedimientos para mejorar las técnicas de compresión de vídeo, tales como las utilizadas en la compresión de vídeo de enlace de visualización. Más específicamente, la presente divulgación se refiere a sistemas y procedimientos para mejorar la eficiencia de codificación de una primera línea en un fragmento cuando se usan técnicas de codificación multimodo.

40 **[0008]** Si bien determinados modos de realización se describen en el presente documento en el contexto de la norma DSC, un experto en la técnica apreciará que los sistemas y procedimientos divulgados en el presente documento pueden ser aplicables a cualquier norma de codificación de vídeo adecuada. Por ejemplo, los modos de realización divulgados en el presente documento pueden ser aplicables a una o más de las siguientes normas: la H.261 del Sector de Normalización de las Telecomunicaciones (ITU-T) de la Unión Internacional de Telecomunicación (ITU), la ITU-T H.262 del Grupo 1 de Expertos en Imágenes en Movimiento (MPEG-1) Visual de la Organización Internacional de Normalización/Comisión Electrotécnica Internacional (ISO/IEC), la ITU-T H.263 de MPEG-2 Visual de ISO/IEC, la ITU-T H.264 de MPEG-4 Visual de ISO/IEC (también conocida como ISO/IEC MPEG-4 AVC), codificación de vídeo de alta eficacia (HEVC) y extensiones de dichas normas. Además, las técnicas descritas en esta divulgación pueden formar parte de normas desarrolladas en el futuro. En otras palabras, las técnicas descritas en esta divulgación pueden ser aplicables a normas de codificación de vídeo desarrolladas previamente, normas de codificación de vídeo actualmente en desarrollo y normas inminentes de codificación de vídeo.

50 **[0009]** Una metodología propuesta para la norma DSC incluye una pluralidad de modos de codificación en los que cada bloque de datos de vídeo puede ser codificado por un codificador y, de manera similar, descodificado por un descodificador. En algunas implementaciones, los píxeles de una imagen pueden codificarse usando una pluralidad de modos de codificación diferentes (es decir, usando codificación de vídeo multimodo). Sin embargo, algunos de los modos usados en la codificación de un fragmento de la imagen pueden tener un mejor rendimiento para líneas del fragmento que no sean la primera línea del fragmento. Por ejemplo, determinados modos de codificación de vídeo pueden aprovechar los predictores que están ubicados espacialmente sobre un píxel actual a codificar. Dado que la primera línea de un fragmento no puede acceder a predictores ubicados espacialmente por encima de la primera línea del fragmento (es decir, las líneas anteriores no están disponibles para la codificación de una primera línea de un fragmento), estas técnicas de codificación de vídeo pueden tener una calidad de codificación reducida para la primera línea del fragmento.

65 **[0010]** En la presente divulgación, se describe un procedimiento mejorado de codificación de una primera línea de un fragmento usando un modo de predicción espacial. Por ejemplo, los píxeles de la primera línea del fragmento pueden codificarse usando píxeles anteriores de la primera línea como predictores. En otras palabras, un píxel anterior

que está ubicado antes de un píxel actual que se codificará en la primera línea del fragmento puede usarse como un predictor al codificar el píxel actual. El píxel anterior puede estar ubicado espacialmente antes del píxel actual en orden de codificación. Esto puede mejorar la calidad de codificación de la primera línea del fragmento sin acceder a las líneas anteriores ubicadas por encima de la primera línea del fragmento.

5

Normas de codificación de vídeo

[0011] Una imagen digital, tal como una imagen de vídeo, una imagen de televisión, una imagen fija o una imagen generada por una grabadora de vídeo o un ordenador, puede incluir píxeles o muestras dispuestas en líneas horizontales y verticales. La cantidad de píxeles en una sola imagen suele ser de decenas de miles. Cada píxel contiene típicamente información de luminancia y crominancia. Sin compresión, la mera cantidad de información a transmitir desde un codificador de imágenes a un decodificador de imágenes haría nada práctica la transmisión de imágenes en tiempo real. Para reducir la cantidad de información a transmitir, se han desarrollado una serie de procedimientos de compresión diferentes, tales como las normas JPEG, MPEG y H.263.

10

15

[0012] Las normas de codificación de vídeo incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual, ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC) y HEVC, incluidas las extensiones de dichas normas.

20

[0013] Además, VESA ha desarrollado una norma de codificación de vídeo, a saber, DSC. La norma DSC es una norma de compresión de vídeo que puede comprimir vídeo para su transmisión a través de enlaces de visualización. A medida que aumenta la resolución de los dispositivos de visualización, el ancho de banda de los datos de vídeo requeridos para controlar los dispositivos de visualización aumenta de manera correspondiente. Es posible que algunos enlaces de visualización no tengan el ancho de banda para transmitir todos los datos de vídeo al dispositivo de visualización para tales resoluciones. En consecuencia, la norma DSC especifica una norma de compresión para la compresión interoperable, sin pérdida visual, a través de enlaces de visualización.

25

[0014] La norma DSC es diferente a otras normas de codificación de vídeo, tales como H.264 y HEVC. La DSC incluye compresión intracadro, pero no incluye compresión intercadro, lo que significa que la norma DSC no puede usar información temporal para codificar los datos de vídeo. Por el contrario, otras normas de codificación de vídeo pueden emplear compresión intercadro en sus técnicas de codificación de vídeo.

30

Sistema de codificación de vídeo

[0015] A continuación, en el presente documento se describen de forma más detallada diversos aspectos de los sistemas, aparatos y procedimientos novedosos, con referencia a los dibujos adjuntos. Sin embargo, la presente divulgación se puede realizar de muchas formas diferentes y no se debería interpretar que está limitada a ninguna estructura o función específica presentada a lo largo de esta divulgación. En cambio, estos aspectos se proporcionan para que esta divulgación sea exhaustiva y completa, y transmita por completo el alcance de la divulgación a los expertos en la materia. Basándose en las enseñanzas del presente documento, un experto en la materia debería apreciar que el alcance de la divulgación pretende abarcar cualquier aspecto de los sistemas, aparatos y procedimientos novedosos divulgados en el presente documento, ya sean implementados de forma independiente de, o en combinación con, cualquier otro aspecto de la presente divulgación. Por ejemplo, un aparato se puede implementar o un procedimiento se puede llevar a la práctica usando un número cualquiera de los aspectos expuestos en el presente documento. Además, el alcance de la presente divulgación pretende abarcar dicho aparato o procedimiento que se lleva a la práctica usando otra estructura, funcionalidad o estructura y funcionalidad, además, o aparte, de los diversos aspectos de la presente divulgación expuestos en el presente documento. Se debe entender que cualquier aspecto divulgado en el presente documento se puede realizar mediante uno o más elementos de una reivindicación.

35

40

45

50

[0016] Aunque en el presente documento se describen aspectos particulares, muchas variaciones y permutaciones de estos aspectos se encuentran dentro del alcance de la divulgación. Aunque se mencionan algunos beneficios y ventajas de los aspectos preferentes, el alcance de la divulgación no pretende estar limitado a beneficios, usos u objetivos particulares. En cambio, los aspectos de la divulgación pretenden ser ampliamente aplicables a diferentes tecnologías inalámbricas, configuraciones de sistema, redes y protocolos de transmisión, algunos de los cuales se ilustran a modo de ejemplo en las figuras y en la siguiente descripción de los aspectos preferentes. La descripción detallada y los dibujos son meramente ilustrativos de la divulgación, en lugar de limitantes, estando definido el alcance de la divulgación por las reivindicaciones adjuntas y equivalentes de las mismas.

55

[0017] Los dibujos adjuntos ilustran ejemplos. Los elementos indicados mediante números de referencia en los dibujos adjuntos corresponden a elementos indicados mediante números de referencia similares en la siguiente descripción. En esta divulgación, los elementos que tienen nombres que comienzan con palabras ordinales (por ejemplo, "primero", "segundo", "tercero", y así sucesivamente) no implican necesariamente que los elementos tienen un orden particular. Más bien, dichas palabras ordinales se usan simplemente para referirse a diferentes elementos de un mismo tipo o un tipo similar.

60

65

[0018] La FIG. 1A es un diagrama de bloques que ilustra un sistema de codificación de vídeo 10 de ejemplo que puede utilizar técnicas de acuerdo con aspectos descritos en esta divulgación. Como se usa en el presente documento, el término "codificador de vídeo" o "codificador" se refiere genéricamente tanto a codificadores de vídeo como a descodificadores de vídeo. En esta divulgación, los términos "codificación de vídeo" o "codificación" pueden referirse genéricamente a la codificación de vídeo y a la descodificación de vídeo. Además de los codificadores de vídeo y los descodificadores de vídeo, los aspectos descritos en la presente solicitud pueden extenderse a otros dispositivos relacionados, tales como los transcodificadores (por ejemplo, dispositivos que pueden descodificar un flujo de bits y recodificar otro flujo de bits) y dispositivos intermedios (por ejemplo, dispositivos que pueden modificar, transformar y/o manipular de otro modo un flujo de bits).

[0019] Como se muestra en la FIG. 1A, el sistema de codificación de vídeo 10 incluye un dispositivo de origen 12 (es decir, "dispositivo de codificación de vídeo 12" o "dispositivo de codificación 12") que genera datos de vídeo codificados para ser descodificados posteriormente por un dispositivo de destino 14 (es decir, "dispositivo de codificación de vídeo 14" o "dispositivo de codificación 14"). En el ejemplo de la FIG. 1A, el dispositivo de origen 12 y el dispositivo de destino 14 constituyen dispositivos independientes. Sin embargo, se observa que el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar en el mismo dispositivo, o ser parte del mismo, como se muestra en el ejemplo de la FIG. 1B.

[0020] Con referencia de nuevo a la FIG. 1A, el dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender, respectivamente, cualquiera entre una amplia gama de dispositivos (también denominados dispositivos de codificación de vídeo) incluidos ordenadores de escritorio, ordenadores tipo *notebook* (por ejemplo, ordenadores portátiles), tabletas electrónicas, descodificadores, equipos telefónicos tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión continua de vídeo o similares. En diversos modos de realización, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para (es decir, configurados para comunicarse a través de) la comunicación inalámbrica.

[0021] Los dispositivos de codificación de vídeo 12, 14 del sistema de codificación de vídeo 10 pueden configurarse para comunicarse a través de redes inalámbricas y tecnologías de radio, tales como operadores de red de área amplia inalámbrica (WWAN) (por ejemplo, celular) y/o red de área local inalámbrica (WLAN). Los términos "red" y "sistema" se usan a menudo de manera intercambiable. Cada uno de los dispositivos de codificación de vídeo 12, 14 puede ser un equipo de usuario (UE), un dispositivo inalámbrico, un terminal, una estación móvil, una unidad de abonado, etc.

[0022] Los operadores WWAN pueden incluir, por ejemplo, redes de comunicación inalámbrica, tales como redes de acceso múltiple por división de código (CDMA), redes de acceso múltiple por división de tiempo (TDMA), redes de acceso múltiple por división de frecuencia (FDMA), redes de FDMA ortogonal (OFDMA), redes de FDMA de única portadora (SC-FDMA) y otras redes. Una red CDMA puede implementar una tecnología de radio tal como el Acceso de Radio Terrestre Universal (UTRA), CDMA2000, etc. UTRA incluye CDMA de Banda Ancha (WCDMA) y otras variantes de CDMA. CDMA2000 abarca las normas IS-2000, IS-95 e IS-856. Una red TDMA puede implementar una tecnología de radio tal como el Sistema Global de Comunicaciones Móviles (GSM). Una red OFDMA puede implementar una tecnología de radio tal como UTRA Evolucionado (E-UTRA), Banda Ancha Ultramóvil (UMB), IEEE 802.11 (Wi-Fi), IEEE 802.16 (WiMAX), IEEE 802.20, Flash-OFDMA, etc. UTRA y E-UTRA son parte del Sistema Universal de Telecomunicaciones Móviles (UMTS). La Evolución a Largo Plazo (LTE) y la LTE Avanzada (LTE-A) de 3GPP son versiones nuevas de UMTS que usan E-UTRA. UTRA, E-UTRA, UMTS, LTE, LTE-A y GSM se describen en documentos de una organización denominada "Proyecto de Colaboración de Tercera Generación" (3GPP). CDMA2000 y UMB se describen en documentos de una organización llamada "Segundo Proyecto de Colaboración de Tercera Generación" (3GPP2).

[0023] Los dispositivos de codificación de vídeo 12, 14 del sistema de codificación de vídeo 10 también pueden comunicarse entre sí a través de una estación base WLAN de acuerdo con una o más normas, tal como la norma IEEE 802.11, que incluye, por ejemplo, estas enmiendas: 802.11a-1999 (denominada comúnmente "802.11a"), 802.11b-1999 (denominada comúnmente "802.11b"), 802.11g-2003 (denominada comúnmente "802.11g"), etc.

[0024] El dispositivo de destino 14 puede recibir, a través del enlace 16, los datos de vídeo codificados a descodificar. El enlace 16 puede comprender cualquier tipo de medio o dispositivo capaz de transferir los datos de vídeo codificados desde el dispositivo de origen 12 al dispositivo de destino 14. En el ejemplo de la FIG. 1A, el enlace 16 puede comprender un medio de comunicación para permitir que el dispositivo de origen 12 transmita datos de vídeo codificados al dispositivo de destino 14 en tiempo real. Los datos de vídeo codificados pueden modularse de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitirse al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrica o por cable, tal como un espectro de radiofrecuencia (RF) o una o más líneas físicas de transmisión. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área amplia o una red global, tal como Internet. El medio de comunicación puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

[0025] En el ejemplo de la **FIG. 1A**, el dispositivo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20 y la interfaz de salida 22. En algunos casos, la interfaz de salida 22 puede incluir un modulador/desmodulador (módem) y/o un transmisor. En el dispositivo de origen 12, la fuente de vídeo 18 puede incluir una fuente tal como un dispositivo de captura de vídeo, por ejemplo, una videocámara, un archivo de vídeo que contiene vídeo previamente capturado, una interfaz de alimentación de vídeo para recibir vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos por ordenador para generar datos de gráficos por ordenador como el vídeo de origen, o una combinación de dichas fuentes. Como un ejemplo, si la fuente de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los denominados "teléfonos con cámara" o "videoteléfonos", tal como se ilustra en el ejemplo de la **FIG. 1B**. Sin embargo, las técnicas descritas en esta divulgación pueden ser aplicables a la codificación de vídeo en general, y se pueden aplicar a aplicaciones inalámbricas y/o por cable.

[0026] El vídeo capturado, precapturado o generado por ordenador puede ser codificado por el codificador de vídeo 20. Los datos de vídeo codificados pueden ser transmitidos al dispositivo de destino 14 por medio de la interfaz de salida 22 del dispositivo de origen 12. Los datos de vídeo codificados pueden almacenarse también (o de forma alternativa) en un dispositivo de almacenamiento (no ilustrado) para un acceso posterior por parte del dispositivo de destino 14 u otros dispositivos, para su descodificación y/o reproducción. El codificador de vídeo 20 ilustrado en las **FIGS. 1A** y **1B** puede comprender el codificador de vídeo 20 ilustrado en la **FIG. 2A** o cualquier otro codificador de vídeo descrito en el presente documento.

[0027] En el ejemplo de la **FIG. 1A**, el dispositivo de destino 14 incluye la interfaz de entrada 28, un descodificador de vídeo 30 y un dispositivo de visualización 32. En algunos casos, la interfaz de entrada 28 puede incluir un receptor y/o un módem. La interfaz de entrada 28 del dispositivo de destino 14 puede recibir los datos de vídeo codificados a través del enlace 16 y/o desde el dispositivo de almacenamiento. Los datos de vídeo codificados, comunicados a través del enlace 16, o proporcionados en el dispositivo de almacenamiento, pueden incluir una variedad de elementos sintácticos generados por el codificador de vídeo 20 para su uso por parte de un descodificador de vídeo, tal como el descodificador de vídeo 30, en la descodificación de los datos de vídeo. Dichos elementos sintácticos se pueden incluir con los datos de vídeo codificados transmitidos en un medio de comunicación, almacenados en un medio de almacenamiento o almacenados en un servidor de archivos. El descodificador de vídeo 30 ilustrado en las **FIGS. 1A** y **1B** puede comprender el descodificador de vídeo 30 ilustrado en la **FIG. 2B** o cualquier otro descodificador de vídeo descrito en el presente documento.

[0028] El dispositivo de visualización 32 puede estar integrado con, o ser externo a, el dispositivo de destino 14. En algunos ejemplos, el dispositivo de destino 14 puede incluir un dispositivo de visualización integrado y también estar configurado para interactuar con un dispositivo de visualización externo. En otros ejemplos, el dispositivo de destino 14 puede ser un dispositivo de visualización. En general, el dispositivo de visualización 32 muestra los datos de vídeo descodificados a un usuario, y puede comprender cualquiera de una variedad de dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

[0029] En aspectos relacionados, la **FIG. 1B** muestra un sistema de codificación de vídeo 10' de ejemplo en el que el dispositivo de origen 12 y el dispositivo de destino 14 están en, o forman parte de, un dispositivo 11. El dispositivo 11 puede ser un equipo telefónico de mano, tal como un teléfono "inteligente" o similar. El dispositivo 11 puede incluir un dispositivo controlador/procesador 13 (opcionalmente presente) en comunicación operativa con el dispositivo de origen 12 y el dispositivo de destino 14. El sistema de codificación de vídeo 10' de la **FIG. 1B**, y sus componentes, son similares de otro modo al sistema de codificación de vídeo 10 de la **FIG. 1A** y sus componentes.

[0030] Aunque cada una de las **FIGS. 1A** y **1B** muestran que el dispositivo de origen 12 incluye la cámara/fuente de vídeo 18, el codificador de vídeo 20, y la interfaz de salida 22 y el dispositivo de destino 14 incluye la interfaz de entrada 28, el descodificador de vídeo 30 y el dispositivo de visualización 32, donde cada uno de los dispositivos de origen y destino 12 y 14 pueden incluir elementos adicionales. Por ejemplo, cada uno de los dispositivos de origen y destino 12 y 14 puede tener una estructura similar que incluye la cámara/fuente de vídeo 18, el codificador de vídeo 20, la interfaz de salida 22, la interfaz de entrada 28, el descodificador de vídeo 30 y el dispositivo de visualización 32. Como tal, en determinadas implementaciones, los dispositivos de origen y de destino 12 y 14 pueden ser intercambiables.

[0031] El codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con una norma de compresión de vídeo, tal como DSC. De forma alternativa, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con otras normas propietarias o industriales, tales como la norma ITU-T H.264, denominada de forma alternativa MPEG-4, Parte 10, AVC, HEVC o ampliaciones de dichas normas. Sin embargo, las técnicas de esta divulgación no están limitadas a ninguna norma de codificación particular. Otros ejemplos de normas de compresión de vídeo incluyen MPEG-2 e ITU-T H.263.

[0032] Aunque no se muestra en los ejemplos de las **FIGS. 1A** y **1B**, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar integrados con un codificador y un descodificador de audio, y pueden incluir unidades MUX-DEMUX adecuadas, u otro hardware y software, para gestionar la codificación, tanto de audio como de vídeo, en un flujo de datos común o en flujos de datos diferentes. Si procede, en algunos ejemplos, las unidades MUX-DEMUX se

pueden ajustar al protocolo de multiplexación ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

[0033] Tanto el codificador de vídeo 20 como el decodificador de vídeo 30 puede implementarse como cualquiera de entre variedad de circuitos codificadores adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), formaciones de compuertas programables in situ (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado no transitorio legible por ordenador y ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta divulgación. Tanto el codificador de vídeo 20 como el decodificador de vídeo 30 pueden estar incluidos en uno o más codificadores o decodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/decodificador combinado en un dispositivo respectivo.

Proceso de codificación de vídeo

[0034] Como se ha mencionado brevemente en lo que antecede, el codificador de vídeo 20 codifica datos de vídeo. Los datos de vídeo pueden comprender una o más imágenes. Cada una de las imágenes es una imagen fija que forma parte de un vídeo. En algunos casos, una imagen puede denominarse una "trama" de vídeo. Cuando el codificador de vídeo 20 codifica los datos de vídeo (por ejemplo, datos de capa de codificación de vídeo (VCL) y/o datos que no son VCL), el codificador de vídeo 20 puede generar un flujo de bits. El flujo de bits puede incluir una secuencia de bits que forma una representación codificada de los datos de vídeo. El flujo de bits puede incluir imágenes codificadas y datos asociados. Una imagen codificada es una representación codificada de una imagen. Los datos VCL pueden incluir datos de imágenes codificadas (es decir, información asociada a muestras de imágenes codificadas) y los datos que no son VCL pueden incluir información de control (por ejemplo, conjuntos de parámetros y/o información de mejora complementaria) asociada a una o más imágenes codificadas.

[0035] Para generar el flujo de bits, el codificador de vídeo 20 puede llevar a cabo operaciones de codificación en cada imagen en los datos de vídeo. Cuando el codificador de vídeo 20 realiza operaciones de codificación en las imágenes, el codificador de vídeo 20 puede generar una serie de imágenes codificadas y datos asociados. Los datos asociados pueden incluir un conjunto de parámetros de codificación, tales como un parámetro de cuantificación (QP). Para generar una imagen codificada, el codificador de vídeo 20 puede dividir una imagen en bloques de vídeo de igual tamaño. Un bloque de vídeo puede ser una matriz bidimensional de muestras. Los parámetros de codificación pueden definir una opción de codificación (por ejemplo, un modo de codificación) para cada bloque de los datos de vídeo. La opción de codificación se puede seleccionar para lograr un rendimiento deseado de velocidad-distorsión.

[0036] En algunos ejemplos, el codificador de vídeo 20 puede dividir una imagen en una pluralidad de fragmentos. Cada uno de los fragmentos puede incluir una región espacialmente distinta en una imagen (por ejemplo, un cuadro) que puede decodificarse de forma independiente sin información del resto de las regiones en la imagen o cuadro. Cada imagen o cuadro de vídeo se puede codificar en un solo fragmento, o cada imagen o cuadro de vídeo se puede codificar en varios fragmentos. En la DSC, el número de bits asignados para codificar cada fragmento pueden ser sustancialmente constante. Como parte de la realización de una operación de codificación en una imagen, el codificador de vídeo 20 puede realizar operaciones de codificación en cada fragmento de la imagen. Cuando el codificador de vídeo 20 realiza una operación de codificación en un fragmento, el codificador de vídeo 20 puede generar datos codificados asociados al fragmento. Los datos codificados asociados al fragmento pueden denominarse "fragmento codificado".

Codificador de vídeo DSC

[0037] La FIG. 2A es un diagrama de bloques que ilustra un ejemplo del codificador de vídeo 20 que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación. El codificador de vídeo 20 puede configurarse para realizar algunas de, o todas, las técnicas de esta divulgación. En algunos ejemplos, las técnicas descritas en esta divulgación pueden compartirse entre los diversos componentes del codificador de vídeo 20. En algunos ejemplos, adicionalmente o de forma alternativa, un procesador (no mostrado) puede configurarse para realizar algunas de, o todas, las técnicas descritas en esta divulgación.

[0038] Con fines explicativos, esta divulgación describe el codificador de vídeo 20 en el contexto de la codificación DSC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o procedimientos de codificación.

[0039] En el ejemplo de la FIG. 2A, el codificador de vídeo 20 incluye una pluralidad de componentes funcionales. Los componentes funcionales del codificador de vídeo 20 incluyen un convertidor de espacio cromático 105, una memoria intermedia 110, un detector de planicidad 115, un controlador de velocidad 120, un componente de predicción, cuantificación y reconstrucción 125, una memoria intermedia de líneas 130, un historial cromático indizado 135, un codificador por entropía 140, un multiplexor de subflujos 145 y una memoria intermedia de velocidad 150. En otros ejemplos, el codificador de vídeo 20 puede incluir más, menos o diferentes componentes funcionales.

[0040] El convertidor de espacio cromático 105 puede convertir un espacio cromático de entrada en el espacio cromático usado en la implementación de codificación. Por ejemplo, en un modo de realización ejemplar, el espacio cromático de los datos de vídeo de entrada está en el espacio cromático de rojo, verde y azul (RGB) y la codificación se implementa en el espacio cromático de la luminancia Y, el verde de crominancia Cg y el naranja de crominancia Co (YCgCo). La conversión del espacio cromático se puede realizar mediante uno o varios procedimientos que incluyen desplazamientos y adiciones en los datos de vídeo. Cabe señalar que los datos de vídeo de entrada en otros espacios cromáticos pueden procesarse y también se pueden realizar conversiones a otros espacios cromáticos.

[0041] En aspectos relacionados, el codificador de vídeo 20 puede incluir la memoria intermedia 110, la memoria intermedia de líneas 130 y/o la memoria intermedia de velocidad 150. Por ejemplo, la memoria intermedia 110 puede contener los datos de vídeo convertidos de espacio cromático antes de su uso por otras partes del codificador de vídeo 20. En otro ejemplo, los datos de vídeo pueden almacenarse en el espacio cromático RGB y la conversión de espacio cromático puede realizarse según sea necesario, ya que los datos convertidos de espacio cromático pueden requerir más bits.

[0042] La memoria intermedia de velocidad 150 puede funcionar como parte del mecanismo de control de velocidad en el codificador de vídeo 20, que se describirá con mayor detalle posteriormente en relación con el controlador de velocidad 120. Los bits empleados en la codificación de (es decir, los bits utilizados para codificar) cada bloque de datos de vídeo pueden variar considerablemente en función de las propiedades (por ejemplo, tamaño, número de bits, etc.) del bloque. La memoria intermedia de velocidad 150 puede suavizar las variaciones de velocidad en el vídeo comprimido (es decir, el flujo de vídeo de salida). En algunos modos de realización, se emplea un modelo de memoria intermedia de velocidad de bits constante (CBR) en el que los bits se extraen de la memoria intermedia a una velocidad de bits constante durante la transmisión de datos a través de un enlace físico por cable. En el modelo de memoria intermedia CBR, si el codificador de vídeo 20 añade demasiados bits al flujo de bits, el número de bits en la memoria intermedia de velocidad 150 puede exceder la capacidad de la memoria intermedia de velocidad 150, causando un desbordamiento. Por otro lado, el codificador de vídeo 20 debe añadir bits a una velocidad suficiente para evitar una infrautilización de la memoria intermedia de velocidad 150.

[0043] En el lado del descodificador de vídeo, los bits se pueden añadir a la memoria intermedia de velocidad 155 del descodificador de vídeo 30 (véase la **FIG. 2B** que se describe posteriormente con más detalle) a una velocidad de bits constante, y el descodificador de vídeo 30 puede eliminar números variables de bits por cada bloque. Para garantizar una correcta descodificación, la memoria intermedia de velocidad 155 del descodificador de vídeo 30 no debe infrautilizarse o desbordarse durante la descodificación del flujo de bits comprimido.

[0044] En algunos modos de realización, la completitud de memoria intermedia (BF) se puede definir en base a una variable TamañoActualMemorialIntermedia, que representa el número de bits actualmente almacenado en la memoria intermedia, y una variable TamañoMáxMemorialIntermedia que representa el tamaño de la memoria intermedia de velocidad 150, es decir, el número máximo global de bits que se pueden almacenar en la memoria intermedia de velocidad 150. La "completitud" de la memoria intermedia (también denominada completitud de memoria intermedia (BF)) puede calcularse como se muestra en la siguiente Ecuación 1. BF representa un porcentaje de la capacidad de una memoria intermedia que se utiliza para el almacenamiento de bits en un momento determinado.

$$BF = ((\text{TamañoActualMemorialIntermedia} * 100) / \text{TamañoMáxMemorialIntermedia}) \quad (\text{Ecuación 1})$$

[0045] El detector de planicidad 115 puede detectar cambios desde áreas complejas (es decir, no planas) en los datos de vídeo a áreas planas (es decir, simples o uniformes) en los datos de vídeo. Los términos "complejo" y "plano" se usarán en el presente documento para referirse, en general, a la dificultad del codificador de vídeo 20 en codificar las regiones respectivas de los datos de vídeo. Por lo tanto, el término "complejo", como se usa en el presente documento, describe, en general, una región de los datos de vídeo como compleja de codificar por el codificador de vídeo 20 y puede, por ejemplo, incluir datos de vídeo texturizados, alta frecuencia espacial y/u otras características que son complejas de codificar. El término "plano", como se usa en el presente documento, describe, en general, una región de los datos de vídeo como simple de codificar por el codificador de vídeo 20 y puede, por ejemplo, incluir un gradiente suave en los datos de vídeo, baja frecuencia espacial y/u otras características que son fáciles de codificar. Las transiciones entre regiones complejas y planas pueden ser usadas por el codificador de vídeo 20 para reducir los artefactos de cuantificación en los datos de vídeo codificados. Específicamente, el controlador de frecuencia 120 y el componente de predicción, cuantificación y reconstrucción 125 pueden reducir dichos artefactos de cuantificación cuando se identifican las transiciones de regiones complejas a planas.

[0046] El controlador de velocidad 120 determina un conjunto de parámetros de codificación, por ejemplo, un QP. El QP puede ajustarse por el controlador de velocidad 120 en base a la completitud de memoria intermedia de la memoria intermedia de velocidad 150 y la actividad de imagen de los datos de vídeo a fin de maximizar la calidad de imagen para una velocidad de bits objetivo, lo que garantiza que la memoria intermedia de velocidad 150 no se desborde o se infrautilice. El controlador de velocidad 120 también selecciona una opción de codificación particular (por ejemplo, un modo particular) para cada bloque de los datos de vídeo a fin de lograr el rendimiento óptimo de velocidad-distorsión. El controlador de velocidad 120 minimiza la distorsión de las imágenes reconstruidas de manera

que el controlador de velocidad 120 satisfaga la restricción de velocidad de bits, es decir, la velocidad de codificación real global está dentro de la velocidad de bits objetivo.

[0047] El componente de predicción, cuantificación y reconstrucción 125 puede realizar al menos tres operaciones de codificación del codificador de vídeo 20. El componente de predicción, cuantificación y reconstrucción 125 puede realizar la predicción en varios modos diferentes. Un modo de predicción de ejemplo es una versión modificada de la predicción adaptativa a la mediana. La norma JPEG sin pérdida (JPEG-LS) puede implementar una predicción adaptativa a la mediana. La versión modificada de la predicción adaptativa a la mediana que puede ser realizada por el componente de predicción, cuantificación y reconstrucción 125 puede permitir la predicción en paralelo de tres valores de muestra consecutivos. Otro modo de predicción de ejemplo es la predicción de bloques. En la predicción de bloques, las muestras se predicen a partir de píxeles reconstruidos previamente en la línea superior o a la izquierda en la misma línea. En algunos modos de realización, tanto el codificador de vídeo 20 como el descodificador de vídeo 30 pueden realizar una búsqueda idéntica en píxeles reconstruidos para determinar los usos de predicción de bloques y, por lo tanto, no es necesario enviar bits en el modo de predicción de bloques. En otros modos de realización, el codificador de vídeo 20 puede realizar la búsqueda y señalar vectores de predicción de bloques en el flujo de bits, de modo que el descodificador de vídeo 30 no necesita realizar una búsqueda aparte. También se puede implementar un modo de predicción de punto medio en el que las muestras se predicen usando el punto medio del intervalo de componente. El modo de predicción de punto medio puede permitir la acotación del número de bits necesarios para el vídeo comprimido, incluso en la muestra del caso más desfavorable.

[0048] El componente de predicción, cuantificación y reconstrucción 125 también realiza la cuantificación. Por ejemplo, la cuantificación se puede realizar por medio de un cuantificador de potencia de 2 que puede implementarse usando un desplazador. Cabe señalar que se pueden implementar otras técnicas de cuantificación en lugar del cuantificador de potencia de 2. La cuantificación realizada por el componente de predicción, cuantificación y reconstrucción 125 puede basarse en el QP determinado por el controlador de velocidad 120. Finalmente, el componente de predicción, cuantificación y reconstrucción 125 también realiza una reconstrucción que incluye añadir el valor residual cuantificado inverso al valor predicho y asegurar que el resultado no esté fuera del intervalo válido de valores de muestra.

[0049] Cabe señalar que los enfoques de ejemplo descritos anteriormente para la predicción, cuantificación y reconstrucción realizadas por el componente de predicción, cuantificación y reconstrucción 125 son meramente ilustrativos y que se pueden implementar otros enfoques. Cabe señalar también que el componente de predicción, cuantificación y reconstrucción 125 puede incluir uno o más subcomponentes para realizar la predicción, la cuantificación y/o la reconstrucción. Además, cabe señalar que la predicción, la cuantificación y/o la reconstrucción pueden realizarse mediante varios componentes codificadores por separado en lugar del componente de predicción, cuantificación y reconstrucción 125.

[0050] La memoria intermedia de líneas 130 contiene la salida del componente de predicción, cuantificación y reconstrucción 125, de modo que el componente de predicción, cuantificación y reconstrucción 125 y el historial cromático indizado 135 pueden utilizar los datos de vídeo almacenados en memoria intermedia. El historial cromático indizado 135 almacena valores de píxeles usados recientemente. El codificador de vídeo 20 puede hacer referencia directamente a estos valores de píxeles usados recientemente por medio de una sintaxis dedicada.

[0051] El codificador por entropía 140 codifica los valores residuales de predicción y cualquier otro dato (por ejemplo, índices identificados por el componente de predicción, cuantificación y reconstrucción 125) recibidos desde el componente de predicción, cuantificación y reconstrucción 125 basándose en el historial cromático indizado 135 y las transiciones de planicidad identificadas por el detector de planicidad 115. En algunos ejemplos, el codificador por entropía 140 puede codificar tres muestras por reloj por cada codificador de subflujo. El multiplexor de subflujo 145 puede multiplexar el flujo de bits basándose en un esquema de multiplexación de paquetes sin cabecera. Esto permite que el descodificador de vídeo 30 ejecute tres descodificadores por entropía en paralelo, facilitando la descodificación de tres píxeles por reloj. El multiplexor de subflujo 145 puede optimizar el orden de los paquetes de modo que el descodificador de vídeo 30 pueda descodificar los paquetes de manera eficaz. Cabe señalar que se pueden implementar diferentes enfoques para la codificación por entropía, lo que puede facilitar la descodificación de una cantidad de píxeles expresada en una potencia de dos, por reloj (por ejemplo, 2 píxeles/reloj o 4 píxeles/reloj).

Descodificador de vídeo DSC

[0052] La **FIG. 2B** es un diagrama de bloques que ilustra un ejemplo del descodificador de vídeo 30 que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación. El descodificador de vídeo 30 puede configurarse para realizar algunas de, o todas, las técnicas de esta divulgación. En algunos ejemplos, las técnicas descritas en esta divulgación pueden compartirse entre los diversos componentes del descodificador de vídeo 30. En algunos ejemplos, adicionalmente o de forma alternativa, un procesador (no mostrado) puede configurarse para realizar algunas de, o todas, las técnicas descritas en esta divulgación.

[0053] Con fines explicativos, esta divulgación describe el descodificador de vídeo 30 en el contexto de la codificación DSC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o

procedimientos de codificación.

[0054] En el ejemplo de la **FIG. 2B**, el descodificador de vídeo 30 incluye una pluralidad de componentes funcionales. Los componentes funcionales del descodificador de vídeo 30 incluyen una memoria intermedia de velocidad 155, un desmultiplexor de subflujo 160, un descodificador por entropía 165, un controlador de velocidad 170, un componente de predicción, cuantificación y reconstrucción 175, un historial cromático indizado 180, una memoria intermedia de líneas 185 y un convertidor de espacio cromático 190. Los componentes ilustrados del descodificador de vídeo 30 son análogos a los componentes correspondientes descritos anteriormente en relación con el codificador de vídeo 20 en la **FIG. 2A**. Como tal, cada uno de los componentes del descodificador de vídeo 30 puede funcionar de una manera similar a los componentes correspondientes del codificador de vídeo 20, como se ha descrito anteriormente.

Fragmentos en la norma DSC

[0055] Como se ha indicado anteriormente, un fragmento se refiere, en general, a una región espacialmente distinta en una imagen o un cuadro que puede descodificarse independientemente sin usar la información del resto de las regiones en la imagen o cuadro. Cada imagen o cuadro de vídeo se puede codificar en un solo fragmento, o cada imagen o cuadro de vídeo se puede codificar en varios fragmentos. En la DSC, el número de bits asignados para codificar cada fragmento pueden ser sustancialmente constante.

[0056] De acuerdo con aspectos de la presente divulgación, la codificación de imágenes puede estar basada en bloques e incluir numerosos modos de codificación, cada uno dirigido a comprimir diferentes tipos de contenido. La selección de modo puede gestionarse mediante una característica de control de velocidad (es decir, el controlador de velocidad 120 o el controlador de velocidad 170) destinada a seleccionar un modo apropiado para cada bloque considerando tanto la velocidad como la distorsión de los modos. La característica de control de velocidad puede permitirse gracias a un modelo de memoria intermedia de descodificador de referencia hipotético (HRD) (es decir, la memoria intermedia de velocidad 150 o la memoria intermedia de velocidad 155). En un ejemplo, la memoria intermedia HRD nunca está en un estado de infrautilización (por ejemplo, menos de cero bits en la memoria intermedia) o desbordamiento (por ejemplo, el tamaño de memoria intermedia ha aumentado más allá de un tamaño máximo establecido).

[0057] En lo que respecta a un códec de vídeo basado en fragmentos, existen desafíos con la compresión de contenido en la primera línea de un fragmento, en relación con no primeras líneas (es decir, líneas distintas a la primera línea) del fragmento. Esto se debe a que las no primeras líneas pueden utilizar predictores altamente correlacionados en las líneas reconstruidas por encima de las no primeras líneas del fragmento, mientras que las primeras líneas no pueden utilizarlos debido al hecho de que las líneas reconstruidas de un fragmento por encima de una primera línea del fragmento dado no están disponibles. Por consiguiente, un modo de codificación que utiliza predictores horizontales puede ser deseable para bloques en la primera línea del fragmento. Este modo puede ser una variante del conjunto más amplio de modos de modulación delta de código de pulso (DPCM) y puede ajustarse para implementarse de manera eficiente en hardware. Además, se puede describir una variante que no usa ningún predictor en absoluto para un primer conjunto de píxeles en el bloque de la primera línea de un fragmento dado. El equilibrio entre las dos técnicas es la complejidad/rendimiento del hardware frente a la tasa/rendimiento de compresión.

Predicción espacial

[0058] Para superar los desafíos descritos anteriormente con los enfoques existentes para técnicas de codificación como DSC, a continuación esta divulgación describe mejoras. En esta divulgación, las siguientes técnicas y enfoques descritos se pueden usar de manera independiente o en cualquier combinación.

[0059] De acuerdo con uno o más aspectos de la presente divulgación, se proporciona un modo/técnica de predicción espacial para la codificación de datos de vídeo, que puede ser, por ejemplo, contenido gráfico. En aspectos relacionados, la complejidad del hardware puede minimizarse al tiempo que se mantiene un alto nivel de rendimiento.

[0060] La **FIG. 3** es un diagrama de bloques que ilustra un procedimiento de codificación de una primera línea de un fragmento de acuerdo con aspectos descritos en esta divulgación. La **FIG. 4** es un diagrama de bloques que ilustra otro procedimiento de codificación de una primera línea de un fragmento de acuerdo con aspectos descritos en esta divulgación. La **FIG. 3** ilustra una parte de un fragmento que incluye una primera línea 203 que incluye un bloque anterior 205 y un bloque actual 207. El bloque anterior 205 incluye una pluralidad de píxeles, que incluyen los píxeles X_{-2} y X_{-1} . El bloque actual 207 incluye una pluralidad de píxeles X_0 a X_{N-1} , donde cada uno de los bloques incluye N píxeles. La **FIG. 4** ilustra una parte de un fragmento que incluye una primera línea 213 que incluye un bloque anterior 215 y un bloque actual 217. En la implementación de la **FIG. 4**, los bloques 215 y 217 son bloques bidimensionales (2D).

[0061] Un píxel actual X_0 del bloque actual 207 puede predecirse en base al segundo píxel vecino izquierdo, es decir, el píxel X_{-2} del bloque anterior 205. Esto puede generalizarse a cualquier píxel de la primera línea 203, por

ejemplo, un píxel actual X_i puede predecirse a partir de un píxel anterior X_{i-2} , que es el segundo vecino izquierdo del píxel actual X_i . Cabe señalar que la predicción del píxel actual X_i a partir del segundo vecino izquierdo permite ejecutar dos rutas de codificación en paralelo (una ruta en píxeles pares y otra ruta en píxeles impares). Esto reduce drásticamente la ruta crítica en una implementación en hardware. Este procedimiento de predicción se denomina "predicción intercalada" en el presente documento.

[0062] Cabe señalar además que, para reducir la dependencia de los bloques reconstruidos anteriores, uno o más píxeles al comienzo del bloque pueden codificarse usando la modulación de código de pulso (PCM) (sin predicción).

[0063] Las técnicas divulgadas en el presente documento se pueden usar para cualquier códec basado en bloques, independientemente de si el tamaño del bloque es unidimensional (1D) o 2D y pueden aplicarse además a bloques de dimensiones superiores (por ejemplo, bloques 3D). Las técnicas divulgadas en el presente documento también se pueden usar junto con cualquier tipo de metodología DPCM. Por ejemplo, las técnicas divulgadas se pueden usar con un esquema DPCM estándar o un esquema de predictor adaptativo a la mediana (MAP).

Predicción espacial unidimensional

[0064] Ejemplos de rutas de predicción para el modo de predicción espacial se ilustran en las FIGS. 3 y 4. Como se describe anteriormente, cada píxel en el bloque actual 207 o 217 puede predecirse a partir de su segundo píxel vecino izquierdo. En algunas implementaciones, cuando el vecino anterior está fuera del fragmento actual, se puede usar un predictor predeterminado, que puede ser, por ejemplo, igual a la mitad del rango dinámico del píxel actual (es decir, el píxel que se está codificando actualmente). El rango dinámico de cada uno de los píxeles en un cuadro de vídeo dado puede ser común para todo el cuadro de vídeo. En este ejemplo, para un contenido RGB que tiene 8 bits/componente (contenido RGB888), el valor predeterminado será de 128 para R, G y B. Para un contenido YCoCg que tenga 8 bits para Y, 9 para Co/Cg, los predictores tendrán un valor predeterminado de 128 para Y y un valor predeterminado de 0 para Co y Cg. Estos valores se pueden escalar en consecuencia para obtener un contenido fuente de mayor profundidad de bits. Por ejemplo, si la fuente es RGB10/10/10, se puede usar un valor predeterminado de 512 para R/G/B. El bucle de predicción se puede calcular de la siguiente manera:

$p(x_0)=x_{-2}$ $a_0=Q(x_0-p(x_0))$
$\hat{x}_0=p(x_0)+Q^{-1}(a_0)$

[0065] En este caso, los coeficientes residuales espaciales cuantificados vienen dados por a ; la cuantificación y la descuantificación vienen dadas por $Q(\cdot)$ y $Q^{-1}(\cdot)$, respectivamente; el píxel reconstruido para x_i se denota como \hat{x}_i ; y la predicción viene dada por $p(\cdot)$. Como se ilustra en las FIGS. 3 y 4, los píxeles x_{-2} y x_{-1} son del bloque reconstruido anterior 205 o 215. La predicción procede para los siguientes dos píxeles de la siguiente manera:

$p(x_1)=x_{-1}$ $a_1=Q(x_1-p(x_1))$ $\hat{x}_1=p(x_1)+Q^{-1}(a_1)$	$p(x_2)=\hat{x}_0$ $a_2=Q(x_2-p(x_2))$ $\hat{x}_2=p(x_2)+Q^{-1}(a_2)$
--	---

[0066] Se puede usar un enfoque similar para los píxeles restantes en el bloque actual.

[0067] Si el bloque actual incluye múltiples líneas (por ejemplo, se emplea un tamaño de bloque de 2x8) como en la implementación de la FIG. 4, entonces el paralelismo se puede aumentar aún más. La predicción intercalada permite 2 rutas de codificación paralelas por línea o fila de cada bloque 215 y 217, y múltiples líneas de los bloques 215 y 217 también se pueden procesar de forma independiente. Por lo tanto, para el tamaño de bloque 2x8, se puede utilizar un total de cuatro rutas paralelas de codificación. Aunque lo anterior ha descrito que cada uno de los bloques 215 y 217 de 2x8 incluye dos líneas, el término "primera línea" como se usa en el presente documento generalmente se refiere a la primera línea de bloques codificada independientemente en el fragmento. Por tanto, la primera línea del fragmento ilustrado en la FIG. 4 puede incluir todos los píxeles de los bloques 215 y 217 aunque estos bloques 215 y 217 son 2D. Por ejemplo, las muestras $x_{0,0}$, $x_{1,0}$, $x_{0,1}$, $x_{1,1}$ en la FIG. 4 pueden predecirse/cuantificarse/reconstruirse en paralelo:

$P(x_{0,0}) = x_{-2,0}$
 $P(x_{1,0}) = x_{-1,0}$
 $P(x_{0,1}) = x_{-2,1}$
 $P(x_{1,1}) = x_{-1,1}$

Predicción espacial MAP

5 **[0068]** Las **FIGS. 5-7** son diagramas de bloques que ilustran las relaciones espaciales entre un píxel actual y los píxeles de predicción en varios procedimientos de codificación de un bloque de acuerdo con los aspectos descritos en esta divulgación. En particular, las **FIGS. 5-7** ilustran rutas de predicción para la predicción espacial MAP 2D. En las **FIGS. 5-7**, los píxeles ilustrados incluyen una línea anterior 221, 231 o 241, una línea actual 223, 233 o 243, un bloque anterior 225, 235 o 245 de la línea actual 223, 233 o 243 y un bloque actual 227, 237 o 247 de la línea actual 223, 233 o 243. La línea anterior 221, 231 o 241 incluye píxeles A a F, y el bloque anterior 225, 235 o 245 incluye los píxeles X_{-2,0}, X_{-1,0}, X_{-2,1} y X_{-1,1}. En las **FIGS. 5-7**, se han usado números de referencia similares para ilustrar diseños similares de los bloques y/o píxeles ilustrados.

15 **[0069]** La **FIG. 5** ilustra un conjunto de píxeles de predicción que se pueden usar en una implementación MAP completa. Los píxeles de predicción ilustrados B, C y X_{-1,0} son los vecinos espaciales más cercanos al píxel actual X_{0,0} y, por lo tanto, pueden tener la mayor probabilidad de estar estrechamente correlacionados con el píxel actual X_{0,0}. La **FIG. 5** tiene el rendimiento más bajo de las implementaciones ilustradas, con una ruta de codificación para todos los píxeles dentro del bloque. En las **FIGS. 6 y 7** se ilustran dos enfoques paralelos, es decir, un conjunto de píxeles de predicción que se pueden usar para DPCM, que aumentan el rendimiento de la implementación de la **FIG. 5** en 2x y 4x, respectivamente. Por ejemplo, el conjunto de píxeles de predicción para la **FIG. 6** puede seleccionarse para permitir que dos rutas de codificación se ejecuten en paralelo ya que puede no haber dependencias de superposición entre las dos rutas de codificación. De manera similar, la **FIG. 7** ilustra una implementación en la que cuatro rutas de codificación pueden ejecutarse en paralelo. Por ejemplo, en la implementación de la **FIG. 7**, dos rutas paralelas por fila pueden ser procesos independientes, donde cada una de las rutas no incluye dependencias de superposición. Esto es beneficioso para el hardware, ya que se reducirá la ruta crítica del caso más desfavorable, disminuyendo la complejidad de la implementación en hardware.

25 **[0070]** El bucle de predicción para la implementación de la **FIG. 5** se puede calcular de la siguiente manera:

$pA=\widehat{x_{-1,0}}, pB=C, pC=B$ $p(x_{0,0})=MED(pA,pB,pA+pB-pC)$ $a_{0,0}=Q(x_{0,0}-p(x_{0,0}))$
$\widehat{x_{0,0}}=p(x_{0,0})+Q^{-1}(a_{0,0})$

30 **[0071]** En este caso, los coeficientes residuales espaciales cuantificados vienen dados por $a_{i,j}$, la cuantificación y la descuantificación vienen dadas por $Q(\cdot)$ y $Q^{-1}(\cdot)$, respectivamente; el píxel reconstruido para x_{ij} se denota como $\widehat{x_{i,j}}$; y la predicción viene dada por $p(\cdot)$. Como se ilustra en la **FIG. 5**, los píxeles x_{-1} son del bloque reconstruido anterior 225 y los píxeles B y C son de la línea anterior 221. Además, $MED(\cdot)$ es una función de mediana que devuelve la mediana de los datos de entrada a la función.

35 **[0072]** El bucle de predicción para la implementación de la **FIG. 6** se puede calcular de la siguiente manera, donde las funciones y valores se han definido anteriormente:

$pA=\widehat{x_{-2,0}}, pB=C, pC=A$ $p(x_{0,0})=MED(pA,pB,pA+pB-pC)$ $a_{0,0}=Q(x_{0,0}-p(x_{0,0}))$ $\widehat{x_{0,0}}=p(x_{0,0})+Q^{-1}(a_{0,0})$

40 **[0073]** Asimismo, el bucle de predicción para la implementación de la **FIG. 7** se puede calcular de la siguiente manera, donde las funciones y valores se han definido anteriormente:

$pA=\widehat{x_{-2,0}}, pB=C, pC=A$ $p(x_{0,0})=MED(pA,pB,pA+pB-pC)$ $a_{0,0}=Q(x_{0,0}-p(x_{0,0}))$ $\widehat{x_{0,0}}=p(x_{0,0})+Q^{-1}(a_{0,0})$	$pA=\widehat{x_{-2,1}}, pB=C, pC=A$ $p(x_{0,1})=MED(pA,pB,pA+pB-pC)$ $a_{0,1}=Q(x_{0,1}-p(x_{0,1}))$ $\widehat{x_{0,1}}=p(x_{0,1})+Q^{-1}(a_{0,1})$
---	---

45 **[0074]** La elección del cuantificador puede depender de los criterios de diseño del códec en conjunto. Sin embargo, se puede emplear cualquier cuantificador adecuado. Por ejemplo, un cuantificador simple se puede usar de la

siguiente manera para un diseño de hardware de bajo coste:

```

Quantificar (valor, bits):
if (value == 0 || bits == 0)
    return value;
Int sign = (value > 0 ? +1 : -1);
Int key = abs(value);
Int round = (1 << (bits - 1));
return (sign * ((key + round) >> bits));

```

```

Descuantificar (valor, bits):
if (value == 0 || bits == 0)
    return value;
Int sign = (value > 0 ? +1 : -1);
Int key = abs(value);
return (sign * (key << bits));

```

5

[0075] Una vez que los valores residuales cuantificados a_i se calculan para todo el bloque, los valores residuales cuantificados a_i pueden codificarse por entropía antes de transmitirse al decodificador.

10

[0076] La FIG. 8 es un diagrama de bloques que ilustra una estrategia de agrupación de ejemplo de acuerdo con aspectos descritos en esta divulgación. Se observa que, dependiendo del tipo de codificador por entropía usado, la estrategia de agrupación para coeficientes cuantificados puede ser uniforme o ceñirse a una estrategia de agrupación mostrada en la FIG. 8. La FIG. 8 incluye una pluralidad de valores residuales cuantificados a_0 a a_{15} que se agrupan en una pluralidad de grupos: grupo1 a grupo4. En la FIG. 8, los valores residuales cuantificados a_0 a a_{15} se pueden agrupar en cuatro grupos: grupo 1, que incluye los valores residuales cuantificados a_0 y a_1 ; grupo2, que incluye los valores residuales cuantificados a_2 , a_3 y a_4 ; grupo3, que incluyen los valores residuales cuantificados a_5 a a_9 ; y grupo 4, que incluye los residuos cuantificados a_{10} a a_{15} .

15

[0077] En otro enfoque, se puede usar un cuantificador que no está basado en una potencia de 2 en lugar de una cuantificación simple. Por ejemplo, se puede usar un cuantificador de estilo HEVC, que mejoraría el rendimiento siempre que la complejidad adicional esté dentro del balance del codificador.

20

[0078] En aún otro enfoque de la técnica descrita anteriormente, los valores residuales espaciales cuantificados pueden señalizarse usando algún otro procedimiento, tal como, por ejemplo, codificación de longitud fija o codificación de código de longitud variable (VLC) usando un diccionario, o similares. Si se usa la codificación VLC, el diccionario puede entrenarse en base a , por ejemplo, las estadísticas esperadas del tipo de codificación.

25

[0079] La FIG. 9 es un diagrama de bloques que ilustra un procedimiento de codificación de una línea de un fragmento de acuerdo con aspectos descritos en esta divulgación. La FIG. 10 es un diagrama de bloques que ilustra otro procedimiento de codificación de una primera línea de un fragmento de acuerdo con aspectos descritos en esta divulgación. En algunas implementaciones, los procedimientos de las FIGS. 9 y 10 se pueden aplicar a la primera línea de un fragmento. En particular, las FIGS. 9 y 10 son técnicas que ilustran implementaciones que corresponden respectivamente al procedimiento de la FIG. 3. En consecuencia, los mismos números de referencia designan los mismos elementos que se ilustraron anteriormente. Específicamente, cada una de las FIGS. 9 y 10 ilustra una parte de un fragmento que incluye una primera línea 203 que incluye un bloque anterior 205 y un bloque actual 207.

30

35

[0080] En el enfoque ilustrado en las FIGS. 9 y 10, los n primeros píxeles en el bloque actual 207 o 217 pueden codificarse explícitamente en el flujo de bits, en lugar de predecirse a partir del bloque reconstruido anterior 205 o 215. La FIG. 9 ilustra el caso en que $n = 2$ y la FIG. 10 ilustra el caso en que $n = 1$. Una ventaja de este enfoque es la independencia total del bloque actual con respecto a datos reconstruidos anteriores. Con este enfoque se puede esperar una velocidad de bits más alta por bloque. En una modificación de este enfoque, los n primeros píxeles pueden codificarse con algún subconjunto de los planos de bits eliminados. Por ejemplo, si el escenario está limitado por la velocidad, los píxeles PCM pueden truncarse a una profundidad de bits más baja antes de la codificación.

45

[0081] El modo de predicción espacial divulgado en el presente documento puede realizarse en cualquier espacio cromático. Por ejemplo, RGB o YCoCg pueden usarse para este tipo de compresión de cuadros. Puede ser conveniente asegurarse de que los predictores usen el mismo espacio cromático que el bloque actual. Por ejemplo, si el bloque actual se está procesando en el espacio YCoCg pero los píxeles reconstruidos están almacenados en RGB, se debe realizar una conversión del espacio cromático de los píxeles reconstruidos.

50

[0082] El rendimiento del modo de predicción espacial de acuerdo con los aspectos de esta divulgación puede demostrarse examinando contenido que es difícil de codificar sin el mismo. En un ejemplo, el contenido de imagen se codifica con y sin el modo propuesto. Estos ejemplos pueden usar una compresión 4:1 de velocidad fija. La calidad objetiva se mide usando la relación de señal pico con respecto a ruido (PSNR), que es una métrica objetiva común

55

para la calidad de imagen.

[0083] El mecanismo de control de velocidad del códec propuesto puede diseñarse para seleccionar el mejor modo de codificación para cada bloque en base al equilibrio entre velocidad y distorsión.

[0084] Especialmente para el contenido en las primeras líneas de un fragmento, el modo de predicción espacial propuesto proporciona una opción de codificación de bajo coste en relación con el coste $D + \lambda \cdot R$. En este caso, el parámetro R se refiere a la velocidad de bits del bloque actual, que puede ser el número total de bits transmitidos entre el codificador 20 y el decodificador 30 para el bloque actual; el parámetro D se refiere a la distorsión del bloque actual, que puede ser la diferencia entre el bloque original y el bloque reconstruido. El parámetro D puede calcularse de varias maneras diferentes, por ejemplo, como la suma de las diferencias absolutas (SAD), la suma de las diferencias al cuadrado, etc., entre los bloques originales y los bloques reconstruidos. El parámetro λ es el parámetro lagrangiano que puede ser un equilibrio entre los parámetros R y D. Cabe señalar que el parámetro lagrangiano λ puede calcularse de varias maneras, y el enfoque seleccionado para el cálculo de λ puede variar según el contexto y la aplicación. Por ejemplo, el parámetro lagrangiano λ puede calcularse en base a una serie de factores, tales como el estado de la memoria intermedia de velocidad 155, las condiciones de primera línea o de no primera línea del bloque, etc. Incluso para condiciones de no primeras líneas de un fragmento, el modo de predicción espacial puede elegirse para numerosos tipos de contenido de imagen.

Diagrama de flujo de ejemplo para la codificación de modo de predicción espacial

[0085] Con referencia a la **FIG. 11**, se describirá un procedimiento de ejemplo para la codificación del modo de predicción espacial. Las etapas ilustradas en la **FIG. 11** se pueden realizar mediante un codificador de vídeo (por ejemplo, el codificador de vídeo 20 en la **FIG. 2A**), un decodificador de vídeo (por ejemplo, el decodificador de vídeo 30 en la **FIG. 2B**) o uno o más componentes de los mismos. Por conveniencia, el procedimiento 300 se describe como realizado por un codificador de vídeo (también denominado simplemente codificador), que puede ser el codificador de vídeo 20, el decodificador de vídeo 30 u otro componente.

[0086] El procedimiento 300 comienza en el bloque 301. En el bloque 305, el codificador codifica un fragmento de datos de vídeo. El fragmento comprende una pluralidad de píxeles organizados en una primera línea y una pluralidad de no primeras líneas. En el bloque 310, el codificador puede codificar el fragmento mediante la codificación de un píxel actual de la primera línea en un modo de predicción espacial usando un píxel anterior de la primera línea como predictor. Puede haber un píxel intermedio entre el píxel anterior y el píxel actual. En el bloque 315, el codificador puede continuar codificando el fragmento por medio de la codificación de al menos otro píxel de la no primera línea en un modo de codificación distinto del modo de predicción espacial. La codificación del fragmento puede comprender además codificar la primera línea por medio de una primera y una segunda ruta de codificación intercaladas. Cada uno de los píxeles en la primera y segunda rutas de codificación puede codificarse independientemente de los píxeles fuera de la ruta de codificación correspondiente. El procedimiento 300 termina en el bloque 320.

[0087] En el procedimiento 300 se puede cambiar el orden en que se realiza el procedimiento. En algunos modos de realización, se pueden añadir bloques adicionales al procedimiento 300.

Otras Consideraciones

[0088] La información y las señales divulgadas en el presente documento se pueden representar usando cualquiera de varias tecnologías y técnicas diferentes. Por ejemplo, los datos, instrucciones, comandos, información, señales, bits, símbolos y fragmentos que se pueden haber referenciado a lo largo de la descripción anterior se pueden representar mediante tensiones, corrientes, ondas electromagnéticas, campos o partículas magnéticos, campos o partículas ópticos o cualquier combinación de los mismos.

[0089] Los diversos bloques lógicos y etapas de algoritmo ilustrativos descritos en relación con los modos de realización divulgados en el presente documento pueden implementarse como hardware electrónico, software informático o combinaciones de ambos. Para ilustrar claramente esta intercambiabilidad de hardware y software, anteriormente se han descrito, en general, diversos componentes, bloques y etapas ilustrativos en términos de su funcionalidad. Que dicha funcionalidad se implemente como hardware o software depende de las restricciones de aplicación y diseño particulares impuestas al sistema global. Los expertos en la técnica pueden implementar la funcionalidad descrita de distintas formas para cada aplicación particular, pero no se debe interpretar que dichas decisiones de implementación suponen apartarse del alcance de la presente divulgación.

[0090] Las técnicas descritas en el presente documento pueden implementarse en hardware, software, firmware o en cualquier combinación de los mismos. Dichas técnicas pueden implementarse en cualquiera entre una variedad de dispositivos tales como ordenadores de propósito general, equipos manuales de dispositivos de comunicación inalámbrica o dispositivos de circuitos integrados que tienen múltiples usos, incluyendo su aplicación en equipos manuales de dispositivos de comunicación inalámbrica y otros dispositivos. Cualquier característica descrita como dispositivos o componentes pueden implementarse juntas en un dispositivo lógico integrado o por separado, como dispositivos lógicos discretos pero interoperables. Si se implementan en software, las técnicas pueden realizarse, al

menos en parte, mediante un medio de almacenamiento de datos legible por ordenador que comprenda código de programa que incluya instrucciones que, cuando se ejecuten, realicen uno o más de los procedimientos descritos anteriormente. El medio de almacenamiento de datos legible por ordenador puede formar parte de un producto de programa informático, que puede incluir materiales de embalaje. El medio legible por ordenador puede comprender memoria o medios de almacenamiento de datos, tales como memoria de acceso aleatorio (RAM), tal como memoria de acceso aleatorio dinámica síncrona (SDRAM), memoria de solo lectura (ROM), memoria de acceso aleatorio no volátil (NVRAM), memoria de solo lectura programable y borrrable eléctricamente (EEPROM), memoria FLASH, medios de almacenamiento de datos magnéticos u ópticos, y similares. Las técnicas se pueden realizar, adicionalmente o de forma alternativa, al menos en parte, por un medio de comunicación legible por ordenador que transporta o comunica código de programa en forma de instrucciones o estructuras de datos y al que se puede acceder, leer y/o ejecutar por medio de un ordenador, tales como señales u ondas propagadas.

[0091] El/los procesador(es) en comunicación con (por ejemplo, funcionando conjuntamente con) el medio legible por ordenador (por ejemplo, memoria u otro dispositivo de almacenamiento de datos) puede(n) ejecutar las instrucciones del código de programa, y puede(n) incluir uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), formaciones lógicas programables in situ (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. Un procesador de este tipo puede estar configurado para realizar cualquiera de las técnicas descritas en esta divulgación. Un procesador de propósito general puede ser un microprocesador, pero, de forma alternativa, el procesador puede ser cualquier procesador, controlador, microcontrolador o máquina de estados convencional. Un procesador también puede implementarse como una combinación de dispositivos informáticos, por ejemplo, una combinación de un DSP y un microprocesador, una pluralidad de microprocesadores, uno o más microprocesadores junto con un núcleo de DSP o cualquier otra configuración de este tipo. Por consiguiente, el término "procesador", como se usa en el presente documento, puede referirse a cualquier estructura anterior, cualquier combinación de la estructura anterior, o cualquier otra estructura o aparato adecuados para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento puede proporcionarse en un software o hardware dedicados configurados para la codificación y la decodificación, o incorporarse en un codificador-descodificador de vídeo combinado (CODEC). Además, las técnicas se podrían implementar por completo en uno o más circuitos o elementos lógicos.

[0092] Las técnicas de esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluyendo un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). Diversos componentes o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no necesariamente requieren su realización por diferentes unidades de hardware. Más bien, como se ha describe anteriormente, diversas unidades pueden combinarse en una unidad de hardware de códec o proporcionarse por un conjunto de unidades de hardware interoperativas, que incluyen uno o más procesadores como se describe anteriormente, junto con el software y/o firmware adecuados.

REIVINDICACIONES

- 5 1. Un procedimiento (300) para codificar datos de vídeo por medio de una pluralidad de modos de codificación en compresión de vídeo de enlace de visualización, que comprende:
- codificar un fragmento de los datos de vídeo, comprendiendo el fragmento una pluralidad de píxeles en una primera línea (213) del fragmento y una pluralidad de no primeras líneas del fragmento, comprendiendo la codificación del fragmento:
- 10 codificar un píxel actual de la primera línea (213) del fragmento en un modo de predicción espacial usando un píxel anterior de la primera línea (213) como predictor; y
- codificar otro píxel de una no primera línea del fragmento en un modo de codificación que no sea el modo de predicción espacial; donde
- 15 el fragmento de los datos de vídeo se organiza además en una pluralidad de bloques (215, 217), donde cada bloque es un bloque bidimensional, 2D, que incluye al menos dos filas de píxeles, comprendiendo cada línea del fragmento una pluralidad de bloques (215, 217), donde el píxel actual y el píxel anterior están en la misma fila, y donde la codificación del fragmento de los datos de vídeo comprende además codificar
- 20 cada fila de la primera línea (213) por medio de una primera y una segunda rutas de codificación intercaladas correspondientes, donde cada uno de los píxeles en las primera y segunda rutas de codificación intercaladas se codifican independientemente de los píxeles fuera de la ruta de codificación intercalada correspondiente.
- 25 2. El procedimiento (300) de la reivindicación 1, en el que:
- el píxel actual de la primera línea (213) y el píxel anterior de la primera línea (213) están separados por un píxel intermedio; y
- 30 la codificación del fragmento de los datos de vídeo comprende además codificar la primera línea (213) por medio de las primera y segunda rutas de codificación intercaladas, donde cada uno de los píxeles en las primera y segunda rutas de codificación intercaladas se codifica independientemente de los píxeles fuera de la ruta de codificación intercalada correspondiente.
- 35 3. El procedimiento (300) de la reivindicación 1, que comprende además codificar un primer píxel de la primera línea (213) del fragmento sin predicción.
- 40 4. El procedimiento (300) de la reivindicación 3, que comprende además codificar un segundo píxel de la primera línea (213) del fragmento sin predicción, donde el píxel actual de la primera línea (213) y el píxel anterior de la primera línea (213) están separados por un píxel intermedio, la codificación del fragmento de los datos de vídeo comprende además la codificación de la primera línea (213) por medio de las primera y segunda rutas de codificación intercaladas, donde la primera ruta de codificación intercalada comienza por el primer píxel, la segunda ruta de codificación intercalada comienza por el segundo píxel, y cada uno de los píxeles de las primera y segunda rutas de codificación intercaladas se codifican independientemente de los píxeles fuera de la ruta de codificación intercalada correspondiente.
- 45 5. El procedimiento (300) de la reivindicación 3, que comprende además codificar un segundo píxel de la primera línea (213) del fragmento en el modo de predicción espacial usando el primer píxel de la primera línea (213) como predictor, donde el píxel actual de la primera línea (213) y el píxel anterior de la primera línea (213) están separados por un píxel intermedio, la codificación del fragmento de los datos de vídeo comprende además la codificación de la primera línea (213) por medio de las primera y segunda rutas de codificación intercaladas, donde la primera ruta de codificación intercalada comienza por el primer píxel, la segunda ruta de codificación intercalada comienza por el segundo píxel, y cada uno de los píxeles de las primera y segunda rutas de codificación intercaladas se codifican independientemente de los píxeles fuera de la ruta de codificación intercalada correspondiente.
- 50 6. El procedimiento (300) de la reivindicación 3, que comprende además:
- determinar si existe una restricción de velocidad en la codificación de los datos de vídeo; y
- 60 truncar una profundidad de bits del primer píxel en respuesta a la determinación de que existe la restricción de velocidad.
- 65 7. El procedimiento (300) de la reivindicación 1, que comprende además codificar un primer píxel de la primera línea (213) en el modo de predicción espacial usando un predictor predeterminado, donde el predictor predeterminado depende de una profundidad de bits de los datos de vídeo.

- 5 8. El procedimiento (300) de la reivindicación 1, que comprende además codificar un píxel actual de una línea actual en un modo de predicción adaptativa a la mediana (MAP) usando un píxel anterior de la línea actual y los primer y segundo píxeles de una línea anterior como predictores, donde el píxel actual de la línea actual y el píxel anterior de la línea actual están separados por un primer píxel intermedio y los primer y segundo píxeles de la línea anterior están separados por un segundo píxel intermedio, donde la codificación del fragmento de los datos de vídeo comprende además la codificación de la línea actual por medio de las primera y segunda rutas de codificación intercaladas, donde cada uno de los píxeles en las primera y segunda rutas de codificación intercaladas se codifica independientemente de los píxeles fuera de la ruta de codificación intercalada correspondiente.
- 10 9. Un aparato, que comprende:
- 15 medios para codificar un fragmento de los datos de vídeo, comprendiendo el fragmento una pluralidad de píxeles en una primera línea (213) del fragmento y una pluralidad de no primeras líneas del fragmento;
- medios para codificar un píxel actual de la primera línea (213) del fragmento en un modo de predicción espacial usando un píxel anterior de la primera línea (213) como predictor;
- 20 medios para codificar otro píxel de una no primera línea del fragmento en un modo de codificación que no sea el modo de predicción espacial; y donde
- 25 el fragmento de los datos de vídeo se organiza además en una pluralidad de bloques (215, 217), donde cada bloque es un bloque bidimensional, 2D, que incluye al menos dos filas de píxeles, comprendiendo cada línea del fragmento una pluralidad de bloques (215, 217), donde el píxel actual y el píxel anterior están en la misma fila, y donde el aparato comprende además medios para codificar cada fila de la primera línea (213) por medio de una primera y una segunda rutas de codificación intercaladas correspondientes, donde cada uno de los píxeles en las primera y segunda rutas de codificación intercaladas se codifican independientemente de los píxeles fuera de la ruta de codificación intercalada correspondiente.
- 30 10. El aparato de la reivindicación 9, en el que
- 35 el píxel actual de la primera línea (213) y el píxel anterior de la primera línea (213) están separados por un píxel intermedio; y
- el aparato comprende además medios para codificar la primera línea (213) por medio de las primera y segunda rutas de codificación intercaladas, donde cada uno de los píxeles en las primera y segunda rutas de codificación intercaladas se codifica independientemente de los píxeles fuera de la ruta de codificación intercalada correspondiente.
- 40 11. El aparato de la reivindicación 9, que comprende además medios para codificar un primer píxel de la primera línea del fragmento sin predicción.
- 45 12. El aparato de la reivindicación 11, que comprende además:
- medios para codificar un segundo píxel de la primera línea (213) del fragmento sin predicción, donde el píxel actual de la primera línea (213) y el píxel anterior de la primera línea (213) están separados por un píxel intermedio; y
- 50 medios para codificar la primera línea (213) por medio de las primera y segunda rutas de codificación intercaladas, donde la primera ruta de codificación intercalada comienza por el primer píxel, la segunda ruta de codificación intercalada comienza por el segundo píxel, y cada uno de los píxeles de las primera y segunda rutas de codificación intercaladas se codifican independientemente de los píxeles fuera de la ruta de codificación intercalada correspondiente.
- 55 13. El aparato de la reivindicación 11, que comprende además:
- 60 medios para codificar un segundo píxel de la primera línea (213) del fragmento en el modo de predicción espacial usando el primer píxel de la primera línea (213) como predictor, donde el píxel actual de la primera línea (213) y el píxel anterior de la primera línea (213) están separados por un píxel intermedio; y
- 65 medios para codificar la primera línea (213) por medio de las primera y segunda rutas de codificación intercaladas, donde la primera ruta de codificación intercalada comienza por el primer píxel, la segunda ruta de codificación intercalada comienza por el segundo píxel, y cada uno de los píxeles en las primera y segunda rutas de codificación intercaladas se codifican independientemente de los píxeles fuera de la ruta de codificación intercalada correspondiente.

14. El aparato de la reivindicación 11, que comprende además:

5 medios para determinar si existe una restricción de velocidad en la codificación de los datos de vídeo; y

 medios para truncar una profundidad de bits del primer píxel en respuesta a la determinación de que existe la restricción de velocidad.

10 15. Un programa informático que comprende código que, cuando se ejecuta, hace que un ordenador lleve a cabo un procedimiento (300) de acuerdo con una de las reivindicaciones 1 a 8.

10

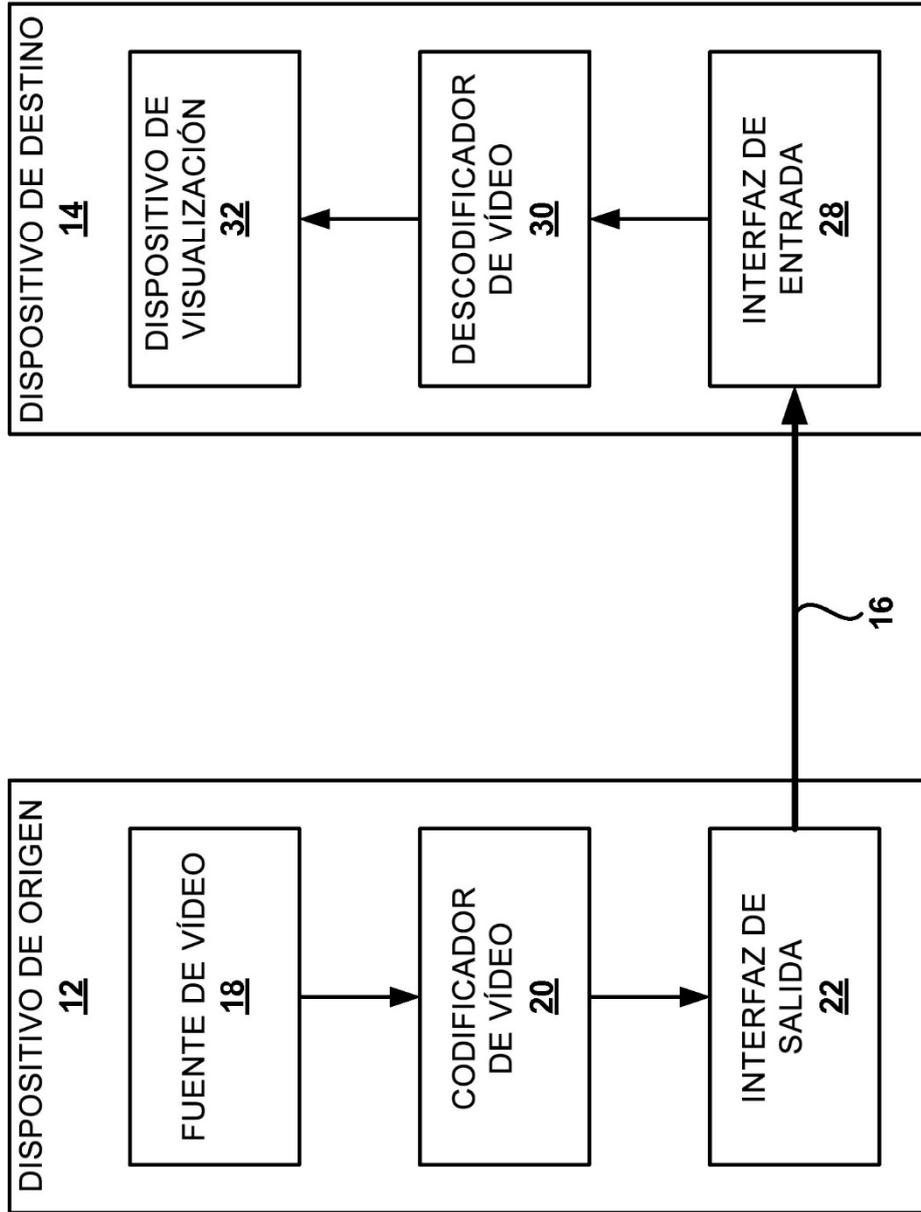


FIG. 1A

10'

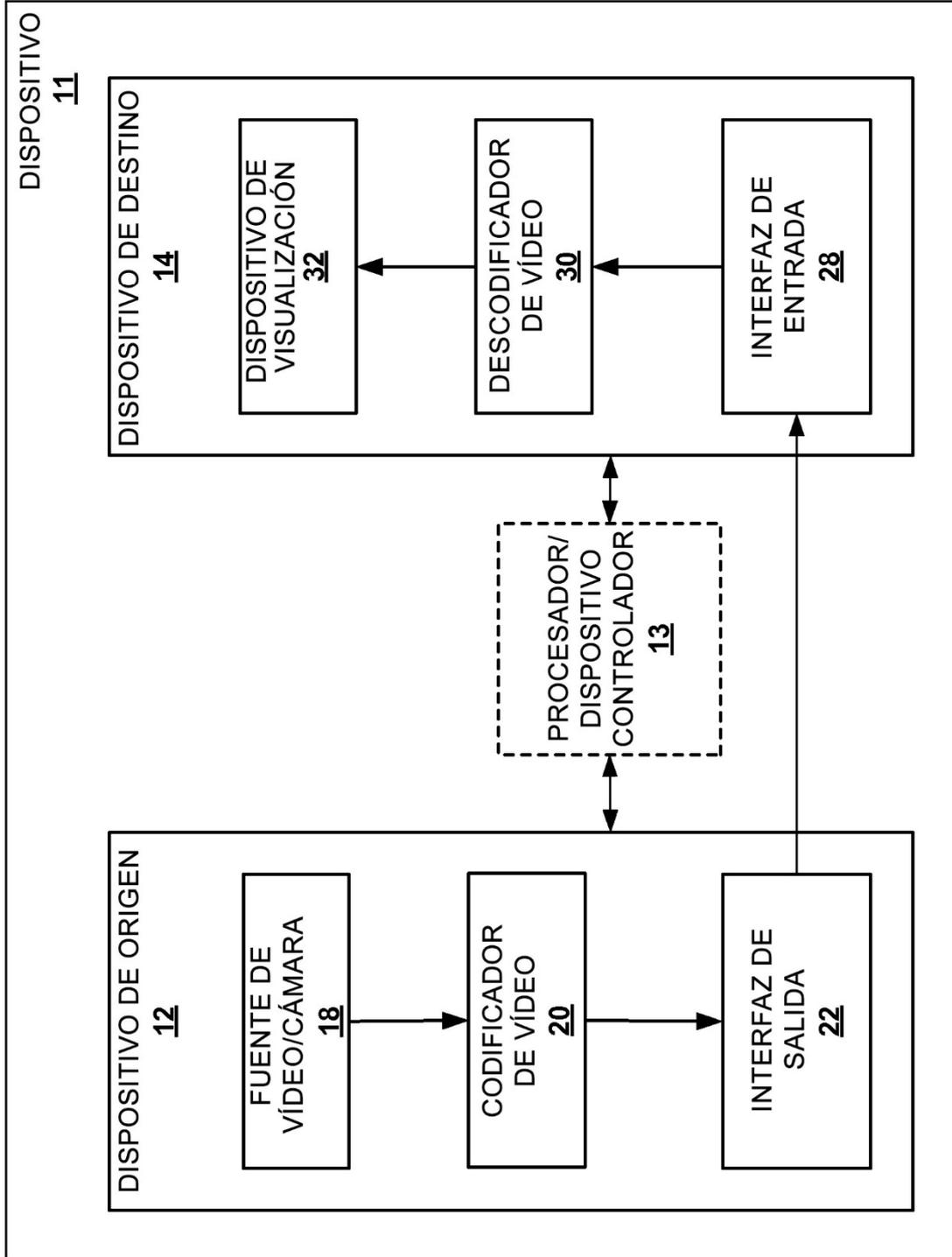


FIG. 1B

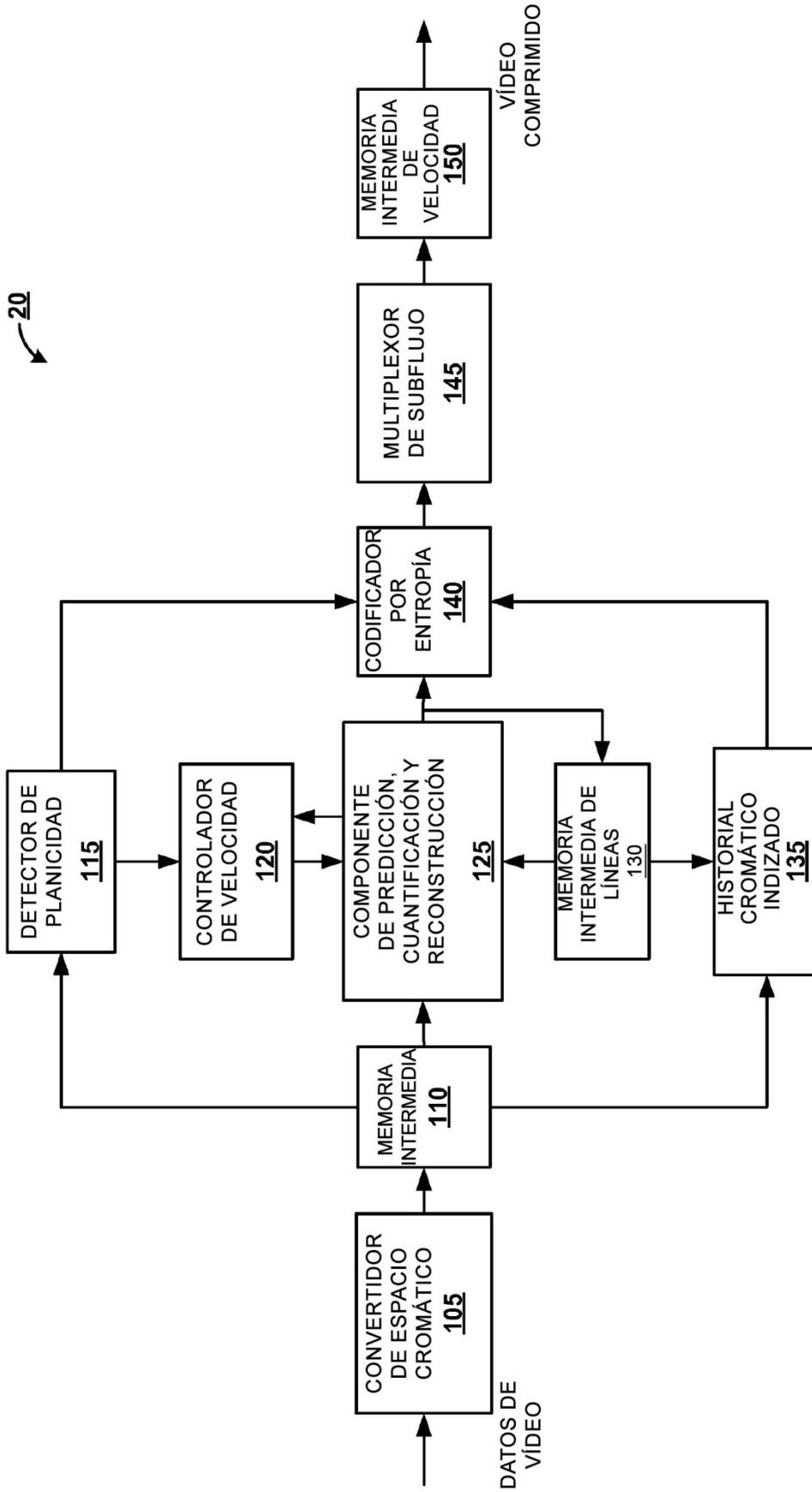


FIG. 2A

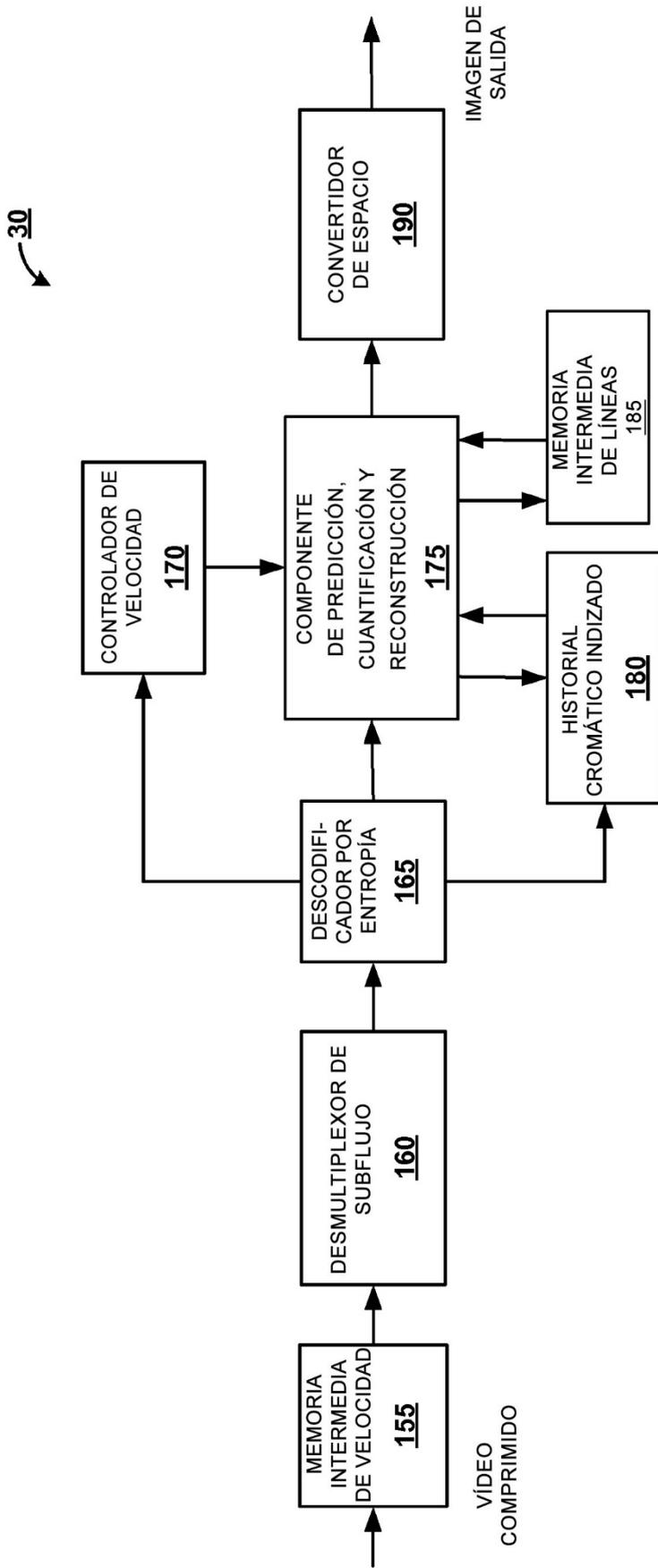


FIG. 2B

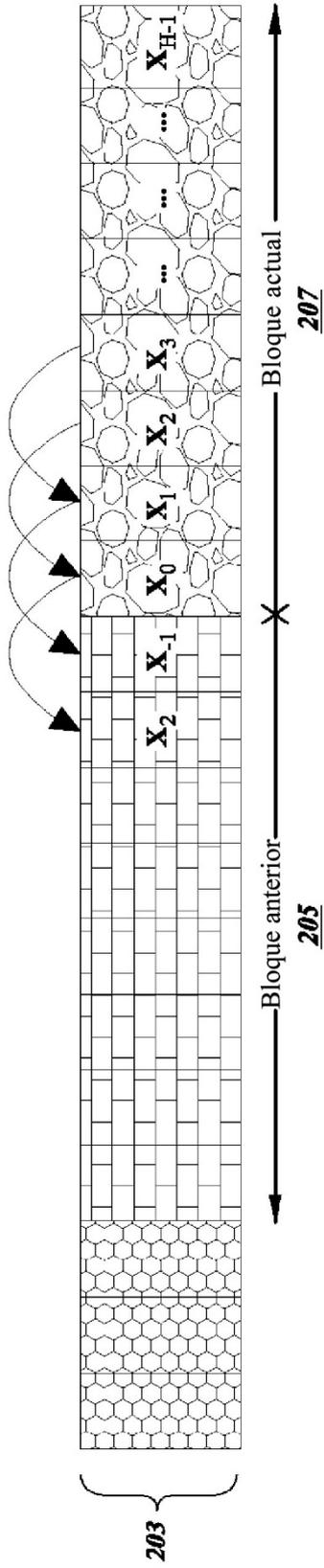


FIG. 3

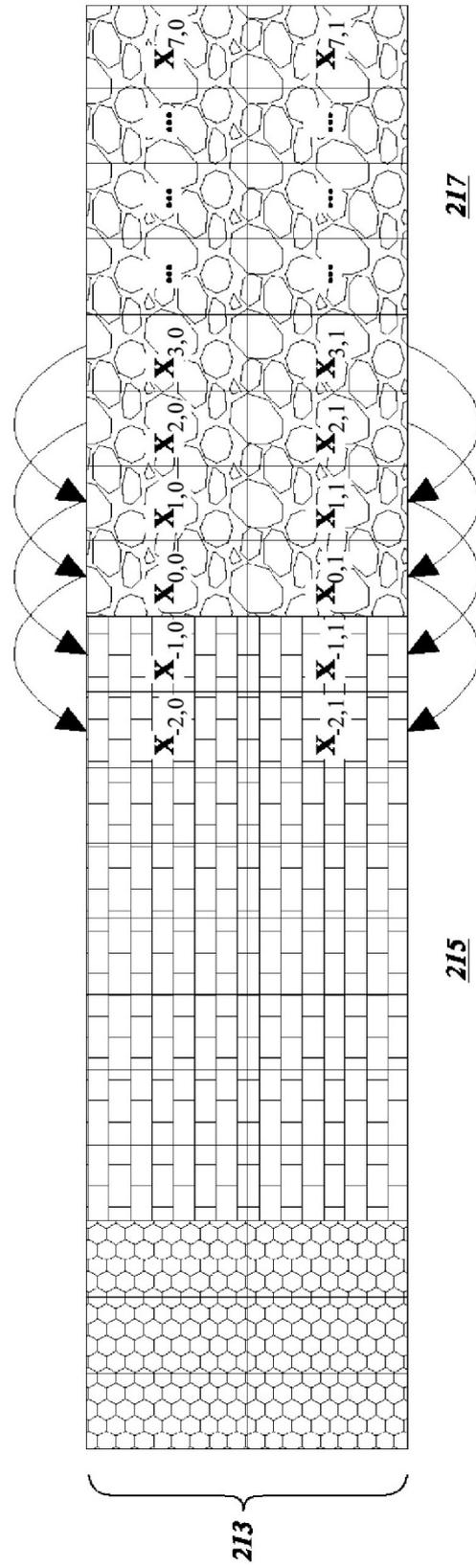


FIG. 4

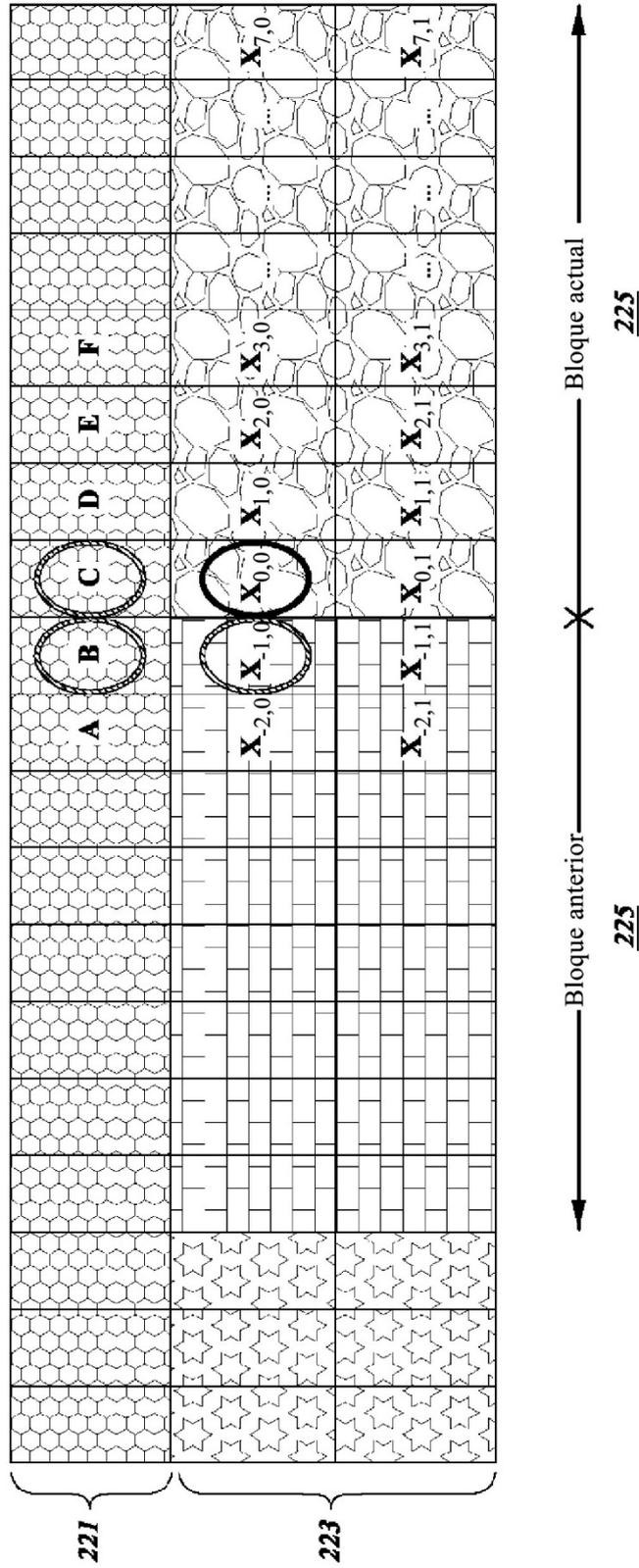


FIG. 5

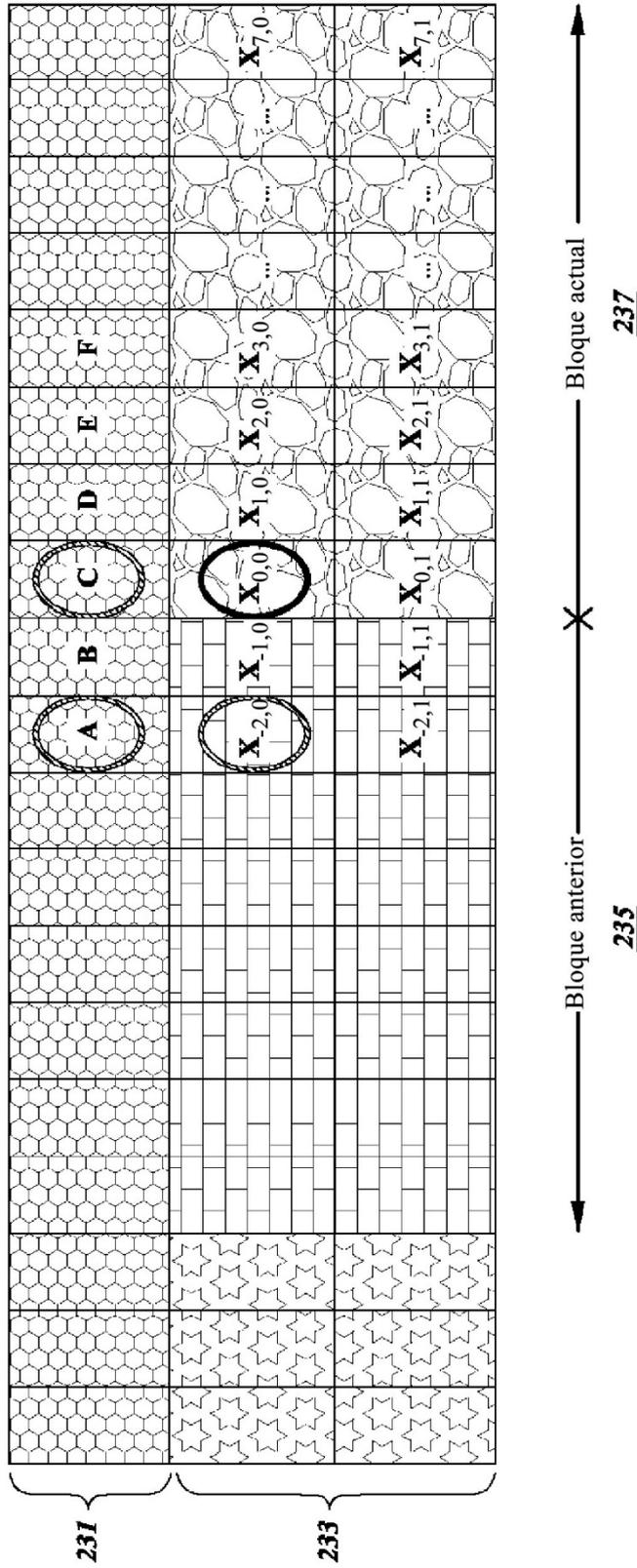


FIG. 6

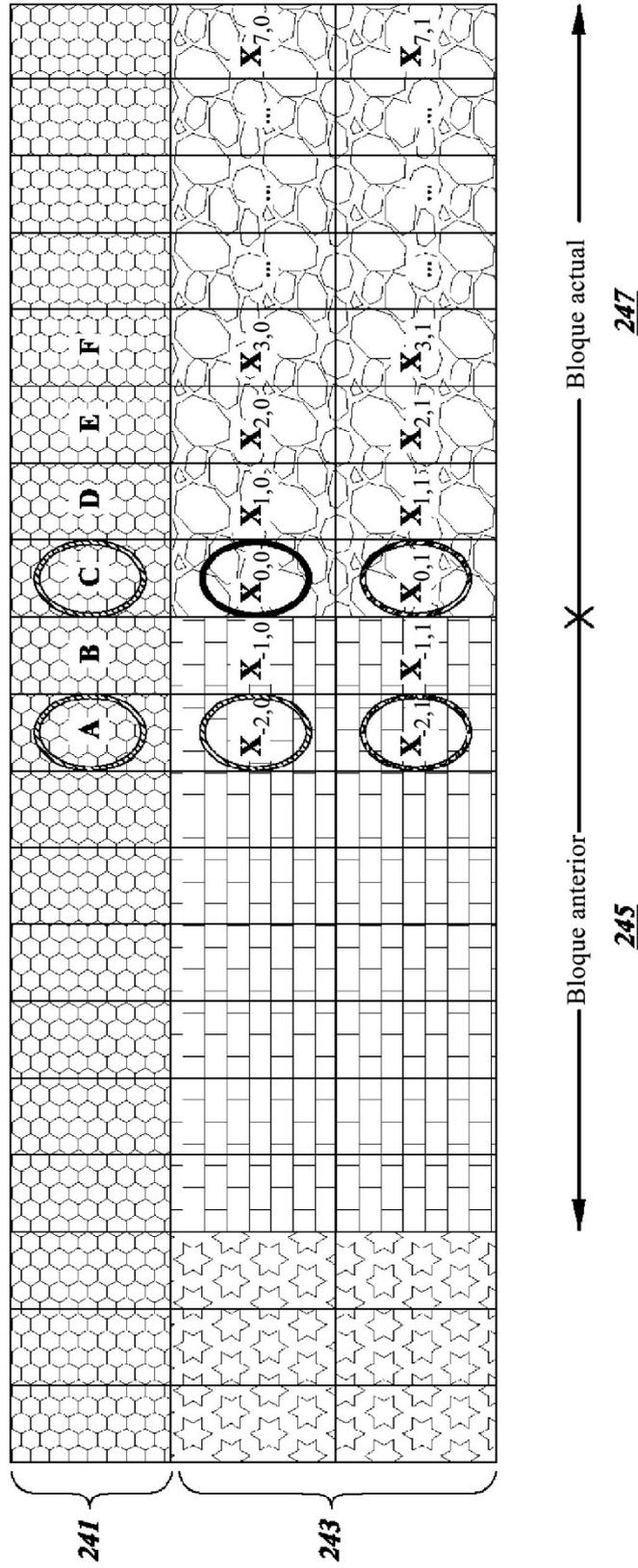


FIG. 7

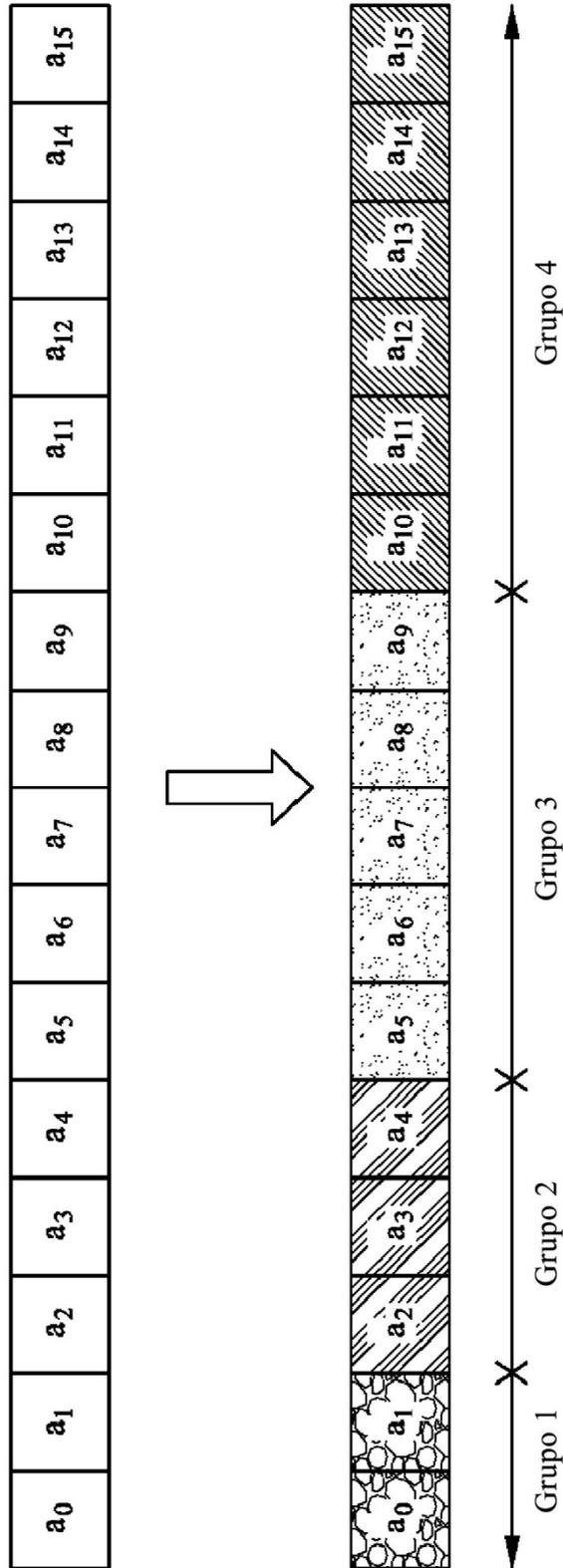


FIG. 8

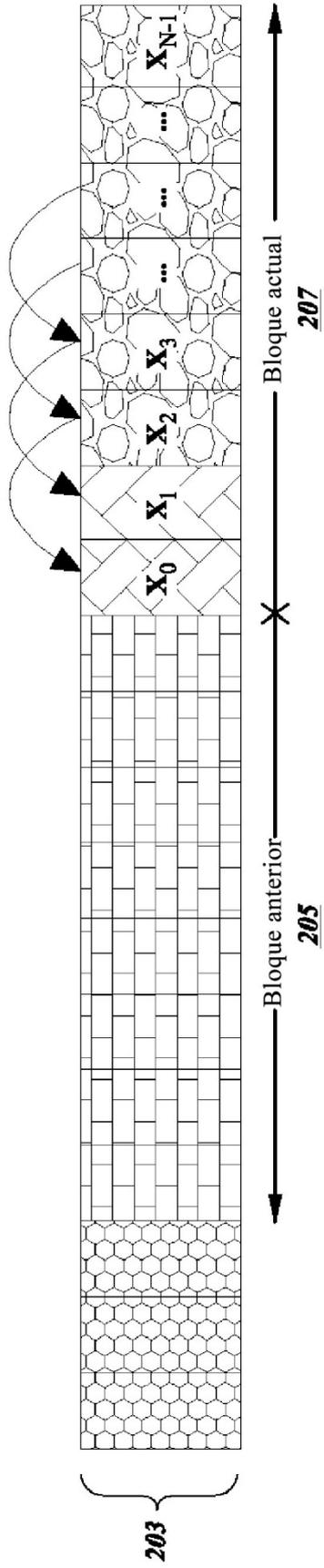


FIG. 9

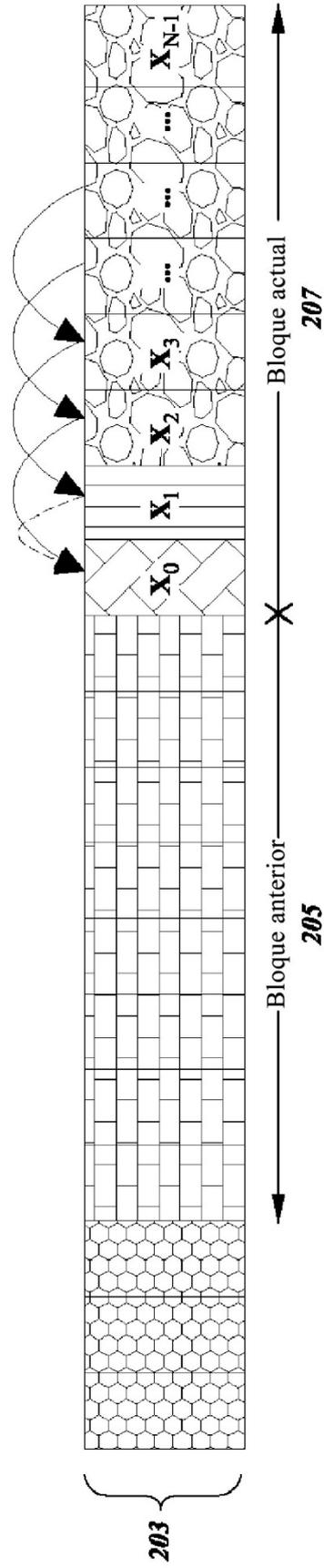


FIG. 10

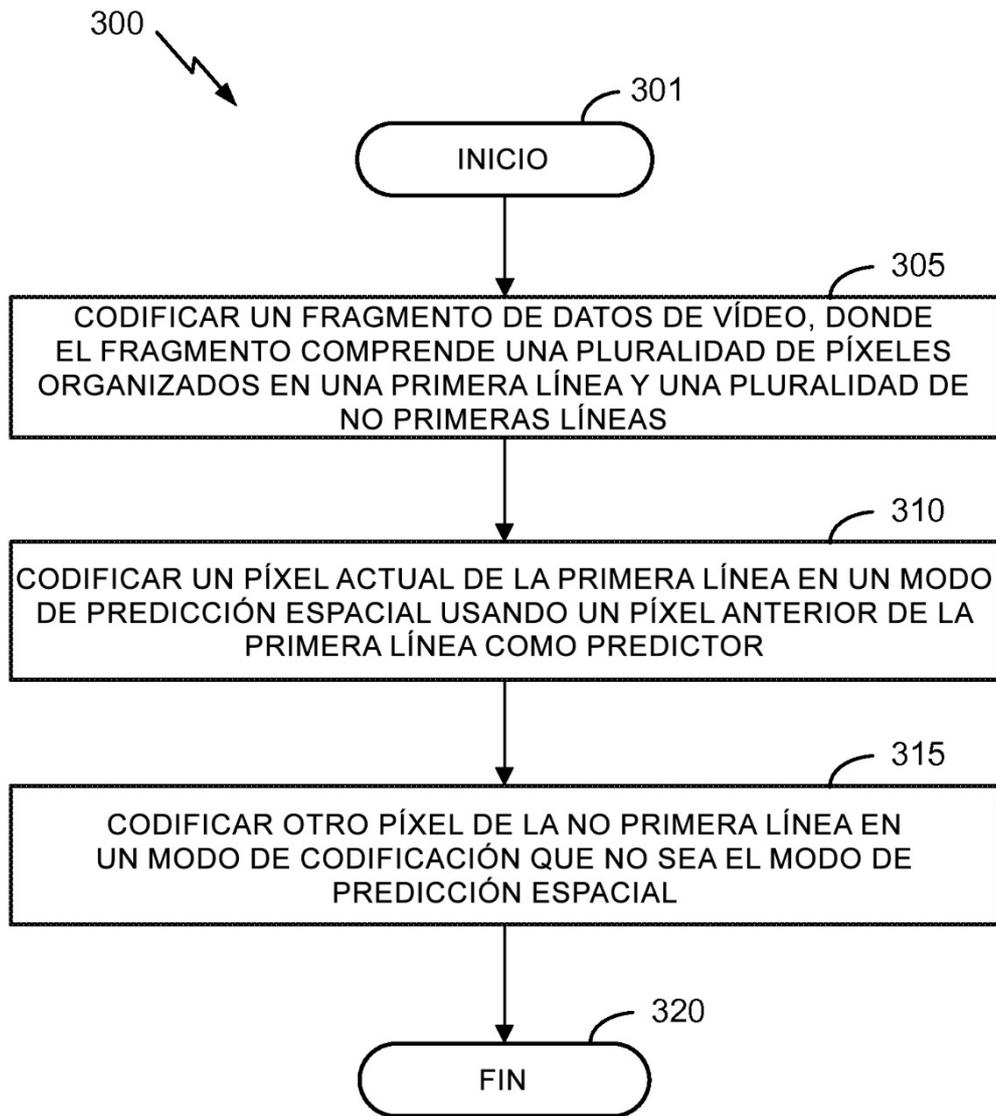


FIG. 11