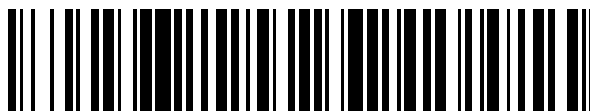


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 755 573**

51 Int. Cl.:

H04N 19/503	(2014.01)	H04N 19/105	(2014.01)
H04N 19/51	(2014.01)		
H04N 19/593	(2014.01)		
H04N 19/70	(2014.01)		
H04N 19/52	(2014.01)		
H04N 19/61	(2014.01)		
H04N 19/176	(2014.01)		
H04N 19/56	(2014.01)		
H04N 19/46	(2014.01)		
H04N 19/573	(2014.01)		

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **26.01.2016 PCT/US2016/014875**
- 87 Fecha y número de publicación internacional: **04.08.2016 WO16123081**
- 96 Fecha de presentación y número de la solicitud europea: **26.01.2016 E 16705364 (4)**
- 97 Fecha y número de publicación de la concesión europea: **14.08.2019 EP 3251361**

54 Título: **Predicción de vector de movimiento temporal avanzada basada en unidades de subpredicción**

30 Prioridad:

26.01.2015 US 201562107933 P
25.01.2016 US 201615005564

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
22.04.2020

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)
ATTN: International IP Administration, 5775
Morehouse Drive
San Diego, CA 92121-1714, US

72 Inventor/es:

CHEN, YING;
LI, XIANG;
LIU, HONGBIN;
CHEN, JIANLE;
ZHANG, LI y
KARCZEWICZ, MARTA

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 755 573 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 de vector de movimiento temporal avanzada basada en unidades de subpredicción

- 5 **[0001]** Esta solicitud reivindica el beneficio de la solicitud provisional estadounidense número 62/107.933 presentada el 26 de enero de 2015.

CAMPO TÉCNICO

- 10 **[0002]** Esta divulgación se refiere a la codificación de vídeo.

ANTECEDENTES

- 15 **[0003]** Las capacidades del vídeo digital se pueden incorporar a una amplia gama de dispositivos, que incluye televisores digitales, sistemas de difusión digital directa, sistemas de difusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores de tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión continua de vídeo y similares. Los dispositivos de vídeo digital implementan técnicas de codificación de vídeo, tales como las descritas en las normas definidas por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificación de Vídeo Avanzada (AVC), ITU-T H.265, también denominada Codificación de Vídeo de Alta Eficacia (HEVC) y las extensiones de dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, descodificar y/o almacenar información de vídeo digital más eficazmente, implementando dichas técnicas de codificación de vídeo.

- 25 **[0004]** Las técnicas de codificación de vídeo incluyen la predicción espacial (intra-imagen) y/o la predicción temporal (inter-imagen) para reducir o eliminar la redundancia intrínseca en las secuencias de vídeo. Para la codificación de vídeo basada en bloques, un corte de vídeo (por ejemplo, una trama de vídeo o una parte de una trama de vídeo) se puede dividir en bloques de vídeo, que para algunas técnicas también se pueden denominar bloques de árbol, unidades de codificación (CU) y/o nodos de codificación. Los bloques de vídeo en un corte intracodificado (I) de una imagen se codifican usando la predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen. Los bloques de vídeo en un corte intercodificado (P o B) de una imagen pueden usar la predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen o la predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes se pueden denominar tramas y las imágenes de referencia se pueden denominar tramas de referencia.

- 35 **[0005]** La predicción espacial o temporal da como resultado un bloque predictivo para un bloque que se va a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original que se va a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intracodificado se codifica de acuerdo con un modo de intracodificación y de acuerdo con los datos residuales. Para una mayor compresión, los datos residuales se pueden transformar desde el dominio de píxel a un dominio de transformada, dando como resultado coeficientes de transformada residuales, que a continuación se pueden cuantificar. Los coeficientes de transformada cuantificados, dispuestos inicialmente en una matriz bidimensional, se pueden explorar a fin de producir un vector unidimensional de coeficientes de transformada, y se puede aplicar la codificación por entropía para lograr aún más compresión.

SUMARIO

- 50 **[0006]** En general, esta divulgación describe técnicas relacionadas con la codificación (por ejemplo, codificación o descodificación) de información de movimiento para un bloque de datos de vídeo. Más en particular, un codificador de vídeo (por ejemplo, un codificador de vídeo o un descodificador de vídeo) se puede configurar para codificar información de movimiento para un bloque actual (por ejemplo, una unidad de predicción (PU) actual) usando la predicción de vector de movimiento temporal avanzada (ATMVP). La ATMVP en general implica el uso de un vector de movimiento temporal para identificar una unidad de predicción correspondiente que se divide en sub-PU. En lugar de dividir la PU actual en sub-PU, el vector de movimiento temporal puede simplemente identificar el bloque correspondiente que se divide en sub-PU, teniendo cada una su propia información de movimiento, y el codificador de vídeo se puede configurar para predecir partes correspondientes del bloque actual usando la información de movimiento de las respectivas sub-PU. Evitando dividir activamente el bloque actual, la información de señalización de sobrecoste se puede reducir para el bloque actual, al tiempo que todavía se logra una predicción de grano fino para partes del bloque actual que de otro modo podrían resultar de dividir el bloque actual en sub-PU.

- 65 **[0007]** En un ejemplo, un procedimiento de codificación de datos de vídeo incluye formar, para un bloque actual de datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales de cuatro bloques vecinos al bloque actual y, de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de

movimiento temporal avanzada (ATMVP), codificar un índice en la lista de candidatos de fusión que identifica un candidato de fusión de la pluralidad de candidatos de fusión en la lista de candidatos de fusión, y codificar el bloque actual de datos de vídeo usando información de movimiento del candidato de fusión identificado.

5 **[0008]** En otro ejemplo, un dispositivo para codificar datos de vídeo incluye una memoria configurada para almacenar datos de vídeo y un codificador de vídeo configurado para formar, para un bloque actual de datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales de cuatro bloques vecinos al bloque actual y, de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de movimiento temporal avanzada
10 (ATMVP), codificar un índice en la lista de candidatos de fusión que identifica un candidato de fusión de la pluralidad de candidatos de fusión en la lista de candidatos de fusión, y codificar el bloque actual de datos de vídeo usando información de movimiento del candidato de fusión identificado.

15 **[0009]** En otro ejemplo, un dispositivo para codificar datos de vídeo incluye medios para formar, para un bloque actual de datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales de cuatro bloques vecinos al bloque actual y, de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de movimiento temporal avanzada (ATMVP), medios para codificar un índice en la lista de candidatos de fusión que identifica un candidato de fusión de la pluralidad de candidatos de fusión en la lista de candidatos de fusión, y
20 medios para codificar el bloque actual de datos de vídeo usando información de movimiento del candidato de fusión identificado.

25 **[0010]** En otro ejemplo, un medio de almacenamiento legible por ordenador ha almacenado en el mismo instrucciones que, cuando se ejecutan, provocan que un procesador forme, para un bloque actual de datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales de cuatro bloques vecinos al bloque actual y, de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de movimiento temporal avanzada (ATMVP), codifique un índice en la lista de candidatos de fusión que identifica un candidato de fusión de la pluralidad de candidatos de fusión en la lista de candidatos de fusión, y codifique el bloque actual de datos de vídeo usando información de movimiento del candidato de fusión identificado.
30

35 **[0011]** Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la descripción a continuación. Otras características, objetos y ventajas resultarán evidentes a partir de la descripción y de los dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0012]

40 La figura 1 es un diagrama de bloques que ilustra un sistema de codificación y descodificación de vídeo de ejemplo que puede utilizar técnicas para implementar la predicción de vector de movimiento temporal avanzada (ATMVP).

45 La figura 2 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo que puede implementar técnicas para la predicción de vector de movimiento temporal avanzada (ATMVP).

La figura 3 es un diagrama de bloques que ilustra un ejemplo de descodificador de vídeo que puede implementar técnicas para la predicción de vector de movimiento temporal avanzada (ATMVP).

50 La figura 4 es un diagrama conceptual que ilustra candidatos vecinos espaciales en la Codificación de Vídeo de Alta Eficacia (HEVC).

La figura 5 es un diagrama conceptual que ilustra la predicción de vector de movimiento temporal (TMVP) en HEVC.

55 La figura 6 es un diagrama conceptual que ilustra una estructura de predicción para 3D-HEVC de ejemplo.

La figura 7 es un diagrama conceptual que ilustra la predicción de movimiento de inter-vista basada en sub-PU en 3D-HEVC.

60 La figura 8 es un diagrama conceptual que ilustra la predicción de movimiento sub-PU a partir de una imagen de referencia.

La figura 9 es un diagrama conceptual que ilustra imágenes relevantes en ATMVP (similar a TMVP).

65 La figura 10 es un diagrama de flujo que ilustra un procedimiento de ejemplo para añadir un candidato de ATMVP a una lista de candidatos durante un proceso de codificación de acuerdo con las técnicas de esta divulgación.

La figura 11 es un diagrama de flujo que ilustra un procedimiento de ejemplo para añadir un candidato de ATMVP a una lista de candidatos durante un proceso de decodificación de acuerdo con las técnicas de esta divulgación.

5 DESCRIPCIÓN DETALLADA

[0013] En general, esta divulgación se refiere a la predicción de vector de movimiento en códecs de vídeo. Más específicamente, la predicción del vector de movimiento temporal avanzada se logra recogiendo los vectores de movimiento en un nivel de subbloque (sub-PU) para un bloque dado (unidad de predicción).

[0014] 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 e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluyendo sus ampliaciones de codificación de vídeo escalable (SVC) y de codificación de vídeo de múltiples vistas (MVC). Otro proyecto conjunto de MVC se describe en "Advanced video coding for generic audiovisual services", Recomendación ITU-T H.264, marzo de 2010.

[0015] Además, existe una norma de codificación de vídeo de nuevo desarrollo, a saber la Codificación de Vídeo de Alta Eficacia (HEVC), desarrollada por el Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) del Grupo de Expertos sobre Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos sobre Imágenes en Movimiento (MPEG) de ISO/IEC. Un proyecto reciente de la HEVC está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip. La norma HEVC también se presenta conjuntamente en la Recomendación ITU-T H.265 y la norma internacional ISO/IEC 23008-2, ambas tituladas "Codificación de vídeo de alta eficacia", y ambas publicadas en octubre de 2014.

[0016] Información de movimiento: Para cada bloque, un conjunto de información de movimiento puede estar disponible. Un conjunto de información de movimiento contiene información de movimiento para las direcciones de predicción hacia delante y hacia atrás. Aquí, las direcciones de predicción hacia delante y hacia atrás son dos direcciones de predicción de un modo de predicción bidireccional y los términos "hacia delante" y "hacia atrás" no tienen necesariamente un significado geométrico; en cambio, corresponden a la lista de imágenes de referencia 0 (RefPicList0) y a la lista de imágenes de referencia 1 (RefPicList1) de una imagen actual. Cuando solo está disponible una lista de imágenes de referencia para una imagen o un corte, solo está disponible la RefPicList0 y la información de movimiento de cada bloque de un corte siempre es hacia delante.

[0017] Para cada dirección de predicción, la información de movimiento debe contener un índice de referencia y un vector de movimiento. En algunos casos, por simplicidad, se puede hacer referencia a un vector de movimiento por sí mismo, de manera que se suponga que tiene un índice de referencia asociado. Se usa un índice de referencia para identificar una imagen de referencia en la lista de imágenes de referencia actuales (RefPicList0 o RefPicList1). Un vector de movimiento tiene un componente horizontal y otro vertical.

[0018] El recuento de orden de imágenes (POC) se usa ampliamente en las normas de codificación de vídeo para identificar el orden de visualización de una imagen. Aunque existen casos en los que dos imágenes dentro de una secuencia de vídeo codificado pueden tener el mismo valor de POC, típicamente no ocurre dentro de una secuencia de vídeo codificado. Cuando están presentes múltiples secuencias de vídeo codificado en un flujo de bits, las imágenes con un mismo valor de POC pueden estar más cercanas entre sí en términos de orden de decodificación. Los valores de POC de las imágenes se usan típicamente para la construcción de la lista de imágenes de referencia, la obtención de un conjunto de imágenes de referencia como en la HEVC y el ajuste a escala del vector de movimiento.

[0019] Estructura de macrobloque (MB) en la codificación de vídeo avanzada (AVC) (H.264): En H.264/AVC, cada inter macrobloque (MB) se puede dividir en cuatro formas diferentes:

- Una división de MB de 16x16
- Dos divisiones de MB de 16x8
- Dos divisiones de MB de 8x16
- Cuatro divisiones de MB de 8x8

[0020] Diferentes divisiones de MB en un MB pueden tener diferentes valores del índice de referencia para cada dirección (RefPicList0 o RefPicList1).

[0021] Cuando un MB no está dividido en cuatro divisiones de MB de 8x8, tiene solo un vector de movimiento para cada división de MB en cada dirección.

[0022] Cuando un MB se divide en cuatro divisiones de MB de 8x8, cada división de MB de 8x8 se puede dividir además en subbloques, cada uno de los cuales puede tener un vector de movimiento diferente en cada dirección. Existen cuatro formas diferentes de obtener subbloques a partir de una división de MB de 8x8:

- 5 • un subbloque de 8x8
- dos subbloques de 8x4
- dos subbloques de 4x8
- 10 • cuatro subbloques de 4x4

[0023] Cada subbloque puede tener un vector de movimiento diferente en cada dirección. Por lo tanto, un vector de movimiento está presente en un nivel igual a mayor que el subbloque.

[0024] Modo directo temporal en AVC: En AVC, el modo directo temporal se puede habilitar bien en el nivel de MB o de división de MB para el modo de salto o directo en los cortes B. Para cada división de MB, los vectores de movimiento del bloque coubicado con la división de MB actual en la RefPicList1[0] del bloque actual se usan para obtener los vectores de movimiento. Cada vector de movimiento en el bloque coubicado se ajusta a escala basado en las distancias de POC.

[0025] Modo directo espacial en AVC: En AVC, un modo directo también puede predecir información de movimiento de los vecinos espaciales.

[0026] Estructura de la unidad de codificación (CU) en la codificación de vídeo de alta eficacia (HEVC): En la HEVC, la unidad de codificación más grande en un corte se denomina un bloque de árbol de codificación (CTB) o unidad de árbol de codificación (CTU). Una CTB contiene un árbol cuaternario cuyos nodos son unidades de codificación.

[0027] El tamaño de una CTB puede variar desde 16x16 a 64x64 en el perfil principal de la HEVC, aunque, técnicamente, se puede prestar soporte a tamaños de CTB de 8x8. Una unidad de codificación (CU) podría ser del mismo tamaño que una CTB y tan pequeña como 8x8. Cada CU está codificada con un modo. Cuando una CU está intercodificada, la CU se puede dividir además en 2 o 4 unidades de predicción (PU) o quedar en una única PU cuando otra división no es aplicable. Cuando dos PU están presentes en una CU, pueden ser rectángulos de mitad de tamaño o de tamaño de dos rectángulos con un $\frac{1}{4}$ o $\frac{3}{4}$ del tamaño de la CU.

[0028] Cuando la CU está inter-codificada, está presente un conjunto de información de movimiento para cada PU. Además, cada PU se codifica con un modo único de interpredicción para obtener el conjunto de información de movimiento.

[0029] Predicción de movimiento en HEVC: En la norma HEVC, hay dos modos de interpredicción, denominados modos de fusión (el salto se considera un caso especial de fusión) y de predicción de vector de movimiento avanzada (AMVP) respectivamente para una unidad de predicción (PU).

[0030] Tanto en el modo AMVP como de fusión, se mantiene una lista de candidatos de vector de movimiento (MV) para predictores de vector de movimiento múltiples. El (los) vector(es) de movimiento, así como los índices de referencia en el modo de fusión, de la PU actual se generan tomando un candidato de la lista de candidatos de MV.

[0031] La lista de candidatos de MV contiene hasta 5 candidatos para el modo de fusión y solo dos candidatos para el modo de AMVP. Un candidato de fusión puede contener un conjunto de información de movimiento, por ejemplo, vectores de movimiento correspondientes a ambas listas de imágenes de referencia (lista 0 y lista 1) y los índices de referencia. Si un candidato de fusión se identifica mediante un índice de fusión, las imágenes de referencia se usan para la predicción de los bloques actuales, así como se determinan los vectores de movimiento asociados. Sin embargo, en el modo de AMVP, para cada dirección de predicción potencial bien de la lista 0 o la lista 1, un índice de referencia necesita ser señalado explícitamente, conjuntamente con un índice de MVP, a la lista de candidatos de MV, ya que el candidato de AMVP solo contiene un vector de movimiento. En el modo de AMVP, los vectores de movimiento predichos se pueden refinar aún más.

[0032] Como se puede observar anteriormente, un candidato de fusión corresponde a un conjunto completo de información de movimiento, mientras que un candidato de AMVP contiene solo un vector de movimiento para una dirección de predicción e índice de referencia específicos.

[0033] Los candidatos para ambos modos se obtienen de forma similar a partir de los mismos bloques vecinos espaciales y temporales.

[0034] El diseño sub-PU para un códec de vídeo 2D, especialmente el referido a la TMVP avanzada, puede encontrar los siguientes problemas. Se puede lograr un proceso de predicción de vector de movimiento temporal basado en sub-

PU definiendo dicho proceso como un candidato adicional, a saber, un candidato de ATMVP. Sin embargo, existen las siguientes cuestiones de diseño para dicho candidato de ATMVP:

- 5 1. Aunque un candidato de ATMVP se puede insertar como un candidato adicional como TMVP, no se conoce la posición de dicho candidato de ATMVP, así como tampoco la interacción con el candidato de TMVP para lograr una mayor eficacia de codificación.
- 10 2. No está claro cómo definir la disponibilidad del candidato de ATMVP; sería de alta complejidad si todos los vectores de movimiento de todas las sub-PU se verificaran para determinar si un candidato de ATMVP no está disponible y, por tanto, se puede insertar en el candidato de ATMVP.
- 15 3. Puede ser necesario el proceso de depuración con un candidato de ATMVP; sin embargo, la depuración con dicho candidato puede ser complicado.
4. Diversos otros detalles de diseño para el candidato de ATMVP para lograr el mejor equilibrio entre la eficacia y la complejidad de codificación siguen siendo desconocidos.

[0035] La figura 1 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo de ejemplo que puede utilizar técnicas para implementar la predicción de vector de movimiento temporal avanzada (ATMVP). Como se muestra en la figura 1, el sistema 10 incluye un dispositivo de origen 12 que proporciona datos de vídeo codificados que se van a decodificar en un momento posterior por un dispositivo de destino 14. En particular, el dispositivo de origen 12 proporciona los datos de vídeo al dispositivo de destino 14 por medio de un medio legible por ordenador 16. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera de una amplia gama de dispositivos, incluyendo ordenadores de escritorio, ordenadores plegables (es decir, portátiles), ordenadores de tableta, decodificadores, equipos telefónicos de mano 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 algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

[0036] El dispositivo de destino 14 puede recibir los datos de vídeo codificados que se van a decodificar por medio del medio legible por ordenador 16. El medio legible por ordenador 16 puede comprender cualquier tipo de medio o dispositivo que puede transportar los datos de vídeo codificados desde el dispositivo de origen 12 al dispositivo de destino 14. En un ejemplo, el medio legible por ordenador 16 puede comprender un medio de comunicación para habilitar el dispositivo de origen 12 para transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. Los datos de vídeo codificados se pueden modular 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 cableada, 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 enrutadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 hasta el dispositivo de destino 14.

[0037] En algunos ejemplos, se pueden emitir datos codificados desde la interfaz de salida 22 hasta un dispositivo de almacenamiento. De forma similar, se puede acceder a los datos codificados desde el dispositivo de almacenamiento mediante una interfaz de entrada. El dispositivo de almacenamiento puede incluir cualquiera de varios medios de almacenamiento de datos, distribuidos o de acceso local, tales como un disco duro, discos Blu-ray, DVD, CD-ROM, memoria flash, memoria volátil o no volátil, o cualquier otro medio de almacenamiento digital adecuado para almacenar datos de vídeo codificados. En un ejemplo adicional, el dispositivo de almacenamiento puede corresponder a un servidor de ficheros o a otro dispositivo de almacenamiento intermedio que pueda almacenar el vídeo codificado generado por el dispositivo de origen 12. El dispositivo de destino 14 puede acceder a datos de vídeo almacenados desde el dispositivo de almacenamiento por medio de transmisión continua o descarga. El servidor de archivos puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al dispositivo de destino 14. Los servidores ejemplares de ficheros incluyen un servidor de la Red (por ejemplo, para una sede de la Red), un servidor del FTP, dispositivos de almacenamiento conectados en red (NAS) o una unidad de disco local. El dispositivo de destino 14 puede acceder a los datos de vídeo codificados a través de cualquier conexión de datos estándar, incluida una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión de Wi-Fi), una conexión cableada (por ejemplo, DSL, módem de cable, etc.) o una combinación de ambas que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de ficheros. La transmisión de datos de vídeo codificados desde el dispositivo de almacenamiento puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas.

[0038] Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas se pueden aplicar a la codificación de vídeo como soporte de cualquiera de una variedad de aplicaciones de multimedia, tales como difusiones de televisión por el aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones continuas de vídeo por internet, tales como la transmisión

continua adaptativa dinámica por HTTP (DASH), vídeo digital que se codifica en un medio de almacenamiento de datos, descodificación de vídeo digital almacenado en un medio de almacenamiento de datos u otras aplicaciones. En algunos ejemplos, el sistema 10 puede estar configurado para prestar soporte a una transmisión de vídeo unidireccional o bidireccional, a fin de prestar soporte a aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

[0039] En el ejemplo de la figura 1, el dispositivo de origen 12 incluye un origen de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. El dispositivo de destino 14 incluye una interfaz de entrada 28, un descodificador de vídeo 30 y un dispositivo de visualización 32. De acuerdo con esta divulgación, el codificador de vídeo 20 del dispositivo de origen 12 puede estar configurado para aplicar las técnicas para la predicción de vector de movimiento temporal avanzada (ATMVP). En otros ejemplos, un dispositivo de origen y un dispositivo de destino pueden incluir otros componentes o disposiciones. Por ejemplo, el dispositivo de origen 12 puede recibir datos de vídeo desde un origen de vídeo externo 18, tal como una cámara externa. Del mismo modo, el dispositivo de destino 14 puede interactuar con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado.

[0040] El sistema 10 ilustrado de la figura 1 es simplemente un ejemplo. Las técnicas para la predicción de vector de movimiento temporal avanzada (ATMVP) se pueden realizar por cualquier dispositivo de codificación y/o descodificación de vídeo digital. Aunque, en general, las técnicas de esta divulgación se realizan por un dispositivo de codificación de vídeo, las técnicas también se pueden realizar por un codificador/descodificador de vídeo, denominado típicamente como un "CÓDEC". Además, las técnicas de esta divulgación también se pueden realizar por un preprocesador de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 son simplemente ejemplos de dichos dispositivos de codificación en los que el dispositivo de origen 12 genera datos de vídeo codificados para su transmisión al dispositivo de destino 14. En algunos ejemplos, los dispositivos 12, 14 pueden funcionar de manera esencialmente simétrica, de modo que cada uno de los dispositivos 12, 14 incluya componentes de codificación y descodificación de vídeo. De ahí que el sistema 10 pueda prestar soporte a la transmisión de vídeo unidireccional o bidireccional entre los dispositivos de vídeo 12, 14, por ejemplo, para la transmisión continua de vídeo, la reproducción de vídeo, la radiodifusión de vídeo o la videotelefonía.

[0041] El origen de vídeo 18 del dispositivo de origen 12 puede incluir un dispositivo de captura de vídeo, tal como una cámara de vídeo, un archivo de vídeo que contiene vídeo capturado previamente y/o una interfaz de suministro de vídeo para recibir vídeo desde un proveedor de contenidos de vídeo. Como otra alternativa, el origen de vídeo 18 puede generar datos basados en gráficos de ordenador como el vídeo de origen, o una combinación de vídeo en directo, vídeo archivado y vídeo generado por ordenador. En algunos casos, si el origen de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los llamados teléfonos con cámara o videoteléfonos. Sin embargo, como se menciona anteriormente, 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 cableadas. En cada caso, el codificador de vídeo 20 puede codificar el vídeo capturado, precapturado o generado por ordenador. Después, la información de vídeo codificada se puede emitir por la interfaz de salida 22 a un medio legible por ordenador 16.

[0042] El medio legible por ordenador 16 puede incluir medios transitorios, tales como una difusión inalámbrica o una transmisión de red cableada, o medios de almacenamiento (es decir, medios de almacenamiento no transitorios), tales como un disco duro, una unidad de memoria flash, un disco compacto, un disco de vídeo digital, un disco Blu-ray u otros medios legibles por ordenador. En algunos ejemplos, un servidor de red (no mostrado) puede recibir datos de vídeo codificados desde el dispositivo de origen 12 y proporcionar los datos de vídeo codificados al dispositivo de destino 14, por ejemplo, por medio de transmisión por red. De forma similar, un dispositivo informático de una instalación de producción de un medio, tal como una instalación de acuñación de discos, puede recibir datos de vídeo codificados desde el dispositivo de origen 12 y producir un disco que contiene los datos de vídeo codificados. Por lo tanto, se puede entender que el medio legible por ordenador 16 incluye uno o más medios legibles por ordenador de diversas formas, en diversos ejemplos.

[0043] La interfaz de entrada 28 del dispositivo de destino 14 recibe información desde el medio legible por ordenador 16. La información del medio legible por ordenador 16 puede incluir información sintáctica definida por el codificador de vídeo 20, que también se usa por el descodificador de vídeo 30, que incluye elementos sintácticos que describen características y/o procesamiento de bloques y otras unidades codificadas, por ejemplo, los GOP. 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 un tubo de rayos catódicos (CRT), 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.

[0044] El codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con una norma de codificación de vídeo, tal como la norma de codificación de vídeo de alta eficacia (HEVC), ampliaciones a la norma HEVC, o normas posteriores, tal como ITU-T H. 266. De forma alternativa, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con otras normas privadas o industriales, tales como la norma ITU-T H.264, denominada de forma alternativa MPEG-4, parte 10, codificación de vídeo avanzada (AVC), 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 codificación de vídeo incluyen MPEG-2 e ITU-T H.263. Aunque no se muestra en la figura 1, en algunos aspectos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar cada uno integrados con un codificador y descodificador de audio, y pueden incluir unidades MUX-DEMUX adecuadas, u otro tipo de hardware y programas informáticos, para manejar la codificación tanto de audio como de vídeo en un flujo de datos común o en flujos de datos separados. Si es aplicable, las unidades MUX-DEMUX pueden ser conformes al protocolo multiplexador ITU H.223 o a otros protocolos tales como el protocolo de datagramas de usuario (UDP).

[0045] El codificador de vídeo 20 y el descodificador de vídeo 30 se pueden implementar cada uno como cualquiera de una variedad de circuitos codificadores adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), arreglos de compuertas programables en el campo (FPGA), lógica discreta, programas informáticos, hardware, firmware o cualquier combinación de los mismos. Cuando las técnicas se implementan parcialmente en programas informáticos, un dispositivo puede almacenar instrucciones para el programa informático en un medio legible por ordenador no transitorio adecuado y ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta divulgación. Cada uno del codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar incluidos en uno o más codificadores o descodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/descodificador (CÓDEC) combinado en un dispositivo respectivo.

[0046] El JCT-VC está trabajando en el desarrollo de la norma HEVC. Los esfuerzos de normalización de la HEVC se basan en un modelo en evolución de un dispositivo de codificación de vídeo, denominado modelo de prueba de la HEVC (HM). El HM supone varias capacidades adicionales de los dispositivos de codificación de vídeo respecto a los dispositivos existentes, de acuerdo, por ejemplo, con la norma ITU-T H.264/AVC. Por ejemplo, mientras que H.264 proporciona nueve modos de codificación de intrapredicción, la HEVC HM puede proporcionar hasta treinta y tres modos de codificación de intrapredicción.

[0047] En general, el modelo de explotación del HM describe que una trama o imagen de vídeo se puede dividir en una secuencia de bloques de árbol o unidades de codificación más grandes (LCU), que incluyen tanto muestras de luminancia como de crominancia. Los datos sintácticos dentro de un flujo de bits pueden definir un tamaño para la LCU, que es la unidad de codificación más grande en términos del número de píxeles. Un corte incluye un número de bloques de árbol consecutivos en orden de codificación. Una trama o imagen de vídeo se puede dividir en uno o más cortes. Cada bloque de árbol se puede dividir en unidades de codificación (CU) de acuerdo con un árbol cuaternario. En general, una estructura de datos de árbol cuaternario incluye un nodo por CU, con un nodo raíz correspondiente al bloque de árbol. Si una CU se divide en cuatro sub-CU, el nodo correspondiente a la CU incluye cuatro nodos hoja, cada uno de los cuales corresponde a una de las sub-CU.

[0048] Cada nodo de la estructura de datos de árbol cuaternario puede proporcionar datos sintácticos para la CU correspondiente. Por ejemplo, un nodo en el árbol cuaternario puede incluir un indicador de división, que indica si la CU correspondiente al nodo está dividida en varias sub-CU. Los elementos sintácticos para una CU se pueden definir de manera recursiva y pueden depender de si la CU está dividida en sub-CU. Si una CU no está dividida adicionalmente, se denomina CU hoja. En esta divulgación, cuatro sub-CU de una CU hoja también se denominarán CU hojas, incluso si no existe ninguna división explícita de la CU hoja original. Por ejemplo, si una CU de tamaño 16x16 no se divide adicionalmente, las cuatro sub-CU de 8x8 también se denominarán CU hojas aunque la CU de 16x16 nunca se dividiera.

[0049] Una CU tiene un propósito similar a un macrobloque de la norma H.264, excepto porque una CU no tiene una distinción de tamaño. Por ejemplo, un bloque de árbol se puede dividir en cuatro nodos hijos (también denominados sub-CU) y cada nodo hijo puede a su vez ser un nodo padre y dividirse en otros cuatro nodos hijos. Un nodo hijo final, no dividido, denominado nodo hoja del árbol cuaternario, comprende un nodo de codificación, también denominado una CU hoja. Los datos sintácticos asociados con un flujo de bits codificado pueden definir un número máximo de veces en que se puede dividir un bloque de árbol, denominada profundidad de CU máxima, y también pueden definir un tamaño mínimo de los nodos de codificación. En consecuencia, un flujo de bits también puede definir la unidad de codificación más pequeña (SCU). Esta divulgación usa el término "bloque" para referirse a cualquiera de una CU, PU o TU, en el contexto de la HEVC, o a estructuras de datos similares en el contexto de otras normas (por ejemplo, macrobloques y subbloques de los mismos en la H.264/AVC).

[0050] Una CU incluye un nodo de codificación y unidades de predicción (PU) y unidades de transformada (TU) asociadas con el nodo de codificación. Un tamaño de la CU corresponde a un tamaño del nodo de codificación y debe tener conformación cuadrada. El tamaño de la CU puede variar desde 8x8 píxeles hasta el tamaño del bloque de árbol, con un máximo de 64x64 píxeles o mayor. Cada CU puede contener una o más PU y una o más TU. Los datos sintácticos asociados con una CU pueden describir, por ejemplo, la división de la CU en una o más PU. Los modos de división pueden diferir entre si la CU está codificada en modo de salto o directo, codificada en modo de intrapredicción o codificada en modo de interpretación. Las PU se pueden dividir para tener una conformación no cuadrada. Los datos sintácticos asociados con una CU también pueden describir, por ejemplo, la división de la CU en una o más TU de acuerdo con un árbol cuaternario. Una TU puede tener una conformación cuadrada o no cuadrada (por ejemplo, rectangular).

[0051] La norma HEVC admite transformaciones de acuerdo con las TU, que pueden ser diferentes para CU diferentes. El tamaño de las TU se basa típicamente en el tamaño de las PU dentro de una CU dada definida para una LCU dividida, aunque puede que no sea siempre así. Las TU son típicamente del mismo tamaño o de un tamaño más pequeño que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU se pueden subdividir en unidades más pequeñas usando una estructura de árbol cuaternario conocida como "árbol cuaternario residual" (RQT). Los nodos hoja del RQT se pueden denominar unidades de transformada (TU). Los valores de diferencias de píxeles asociados a las TU se pueden transformar para generar coeficientes de transformada, que se pueden cuantificar.

[0052] Una CU hoja puede incluir una o más unidades de predicción (PU). En general, una PU representa una zona espacial correspondiente a la totalidad, o a una parte, de la CU correspondiente, y puede incluir datos para recuperar una muestra de referencia para la PU. Además, una PU incluye datos relacionados con la predicción. Por ejemplo, cuando la PU está codificada mediante intramodo se pueden incluir datos para la PU en un árbol cuaternario residual (RQT), que puede incluir datos que describen un modo de intrapredicción para una TU correspondiente a la PU. Como otro ejemplo, cuando la PU está codificada mediante intermodo, la PU puede incluir datos que definen uno o más vectores de movimiento para la PU. Los datos que definen el vector de movimiento para una PU pueden describir, por ejemplo, una componente horizontal del vector de movimiento, una componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, precisión de un cuarto de píxel o precisión de un octavo de píxel), una imagen de referencia a la que apunta el vector de movimiento y/o una lista de imágenes de referencia (por ejemplo, lista 0, lista 1 o lista C) para el vector de movimiento.

[0053] Una CU hoja que tiene una o más PU también puede incluir una o más unidades de transformada (TU). Las unidades de transformada se pueden especificar usando un RQT (también denominado estructura de árbol cuaternario de TU), como se analiza anteriormente. Por ejemplo, un indicador de división puede indicar si una CU hoja está dividida en cuatro unidades de transformada. A continuación, cada unidad de transformada se puede dividir adicionalmente en sub-TU adicionales. Cuando una TU no se divide adicionalmente, se puede denominar una TU hoja. En general, para la intracodificación, todas las TU hojas que pertenecen a una CU hoja comparten el mismo modo de intrapredicción. Es decir, el mismo modo de intrapredicción se aplica en general para calcular valores predichos para todas las TU de una CU hoja. Para la intracodificación, un codificador de vídeo puede calcular un valor residual para cada TU hoja usando el modo de intrapredicción, como una diferencia entre la parte de la CU correspondiente a la TU y el bloque original. Una TU no está necesariamente limitada al tamaño de una PU. Por tanto, las TU pueden ser más grandes o más pequeñas que una PU. Para la intracodificación, una PU puede estar ubicada con una TU hoja correspondiente para la misma CU. En algunos ejemplos, el tamaño máximo de una TU hoja puede corresponder al tamaño de la CU hoja correspondiente.

[0054] Además, las TU de las CU hojas también se pueden asociar con respectivas estructuras de datos de árbol cuaternario, denominadas árboles cuaternarios residuales (RQT). Es decir, una CU hoja puede incluir un árbol cuaternario que indica cómo se divide la CU hoja en las TU. El nodo raíz de un árbol cuaternario de una TU corresponde en general a una CU hoja, mientras que el nodo raíz de un árbol cuaternario de CU corresponde en general a un bloque de árbol (o LCU). Las TU del RQT que no están divididas se denominan TU hojas. En general, esta divulgación usa los términos CU y TU para hacer referencia a una CU hoja y a una TU hoja, respectivamente, a no ser que se indique de otro modo.

[0055] Una secuencia de vídeo incluye típicamente una serie de tramas o imágenes de vídeo. Un grupo de imágenes (GOP) comprende, en general, una serie de una o más de las imágenes de vídeo. Un GOP puede incluir datos sintácticos en un encabezado del GOP, un encabezado de una o más de las imágenes, o en otras ubicaciones, que describen una cantidad de imágenes incluidas en el GOP. Cada corte de una imagen puede incluir datos sintácticos de corte que describen un modo de codificación para el corte respectivo. El codificador de vídeo 20 funciona típicamente sobre bloques de vídeo dentro de cortes de vídeo individuales a fin de codificar los datos de vídeo. Un bloque de vídeo puede corresponder a un nodo de codificación dentro de una CU. Los bloques de vídeo pueden tener tamaños fijos o variados y pueden diferir en tamaño de acuerdo con una norma de codificación especificada.

[0056] En un ejemplo, el HM admite la predicción en diversos tamaños de PU. Suponiendo que el tamaño de una CU particular es $2N \times 2N$, el HM admite la intrapredicción en tamaños de PU de $2N \times 2N$ o $N \times N$, y la interpredicción en tamaños de PU simétricos de $2N \times 2N$, $2N \times N$, $N \times 2N$ o $N \times N$. El HM también admite la división asimétrica para la interpredicción en tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$. En la división asimétrica, una dirección de una CU no está dividida, mientras que la otra dirección está dividida entre un 25 % y un 75 %. La parte de la CU correspondiente a la división del 25 % está indicada por una "n" seguida de una indicación de "arriba", "abajo", "izquierda" o "derecha". Por tanto, por ejemplo, " $2N \times nU$ " se refiere a una CU de tamaño $2N \times 2N$ que está dividida horizontalmente, con una PU de $2N \times 0,5N$ encima y una PU de $2N \times 1,5N$ debajo.

[0057] En esta divulgación, " $N \times N$ " y "N por N" se pueden usar de manera intercambiable para hacer referencia a las dimensiones de píxeles de un bloque de vídeo en términos de dimensiones verticales y horizontales, por ejemplo, 16×16 píxeles o 16 por 16 píxeles. En general, un bloque de 16×16 tendrá 16 píxeles en una dirección vertical ($y = 16$) y 16 píxeles en una dirección horizontal ($x = 16$). Asimismo, un bloque de tamaño $N \times N$ tiene, en general, N píxeles en una dirección vertical y N píxeles en una dirección horizontal, donde N representa un valor entero no negativo. Los

píxeles de un bloque se pueden disponer en filas y columnas. Además, no es necesario que los bloques tengan necesariamente el mismo número de píxeles en la dirección horizontal que en la dirección vertical. Por ejemplo, los bloques pueden comprender $N \times M$ píxeles, donde M no es necesariamente igual a N .

5 **[0058]** Tras la codificación intrapredictiva o interpredictiva, usando las PU de una CU, el codificador de vídeo 20 puede calcular datos residuales para las TU de la CU. Las PU pueden comprender datos sintácticos que describen un procedimiento o modo de generación de datos de píxeles predictivos en el dominio espacial (también denominado el dominio de píxel) y las TU pueden comprender coeficientes en el dominio de transformada, tras la aplicación de una transformada, por ejemplo, una transformada de coseno discreta (DCT), una transformada de enteros, una transformada ondícula o una transformada conceptualmente similar, a los datos de vídeo residuales. Los datos residuales pueden corresponder a diferencias de píxeles entre píxeles de la imagen no codificada y valores de predicción correspondientes a las PU. El codificador de vídeo 20 puede formar las TU, incluyendo los datos residuales para la CU y, a continuación, transformar las TU para producir coeficientes de transformada para la CU.

15 **[0059]** Tras cualquier transformada para producir coeficientes de transformada, el codificador de vídeo 20 puede realizar la cuantificación de los coeficientes de transformada. La cuantificación se refiere, en general, a un proceso en el que los coeficientes de transformada se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando una compresión adicional. El proceso de cuantificación puede reducir la profundidad de bits asociada con algunos de, o todos, los coeficientes. Por ejemplo, un valor de n bits se puede redondear hacia abajo hasta un valor de m bits durante la cuantificación, donde n es mayor que m .

25 **[0060]** Después de la cuantificación, el codificador de vídeo puede explorar los coeficientes de transformada, produciendo un vector unidimensional a partir de la matriz bidimensional que incluye los coeficientes de transformada cuantificados. La exploración puede estar diseñada para colocar los coeficientes de energía más alta (y, por lo tanto, de frecuencia más baja) en la parte delantera de la matriz y para colocar los coeficientes de energía más baja (y, por lo tanto, de frecuencia más alta) en la parte trasera de la matriz. En algunos ejemplos, el codificador de vídeo 20 puede utilizar un orden de exploración predefinido para explorar los coeficientes de transformada cuantificados, para producir un vector serializado que se pueda codificar por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar una exploración adaptativa. Después de explorar los coeficientes de transformada cuantificados para formar un vector unidimensional, el codificador de vídeo 20 puede codificar por entropía el vector unidimensional, por ejemplo, de acuerdo con la codificación de longitud variable adaptativa al contexto (CAVLC), la codificación aritmética binaria adaptativa al contexto (CABAC), la codificación aritmética binaria adaptativa al contexto basada en la sintaxis (SBAC), la codificación por entropía por división en intervalos de probabilidad (PIPE) o con otra metodología de codificación por entropía. El codificador de vídeo 20 también puede codificar por entropía los elementos sintácticos asociados con los datos de vídeo codificados, para su uso por el decodificador de vídeo 30 en la decodificación de los datos de vídeo.

35 **[0061]** Para realizar la codificación CABAC, el codificador de vídeo 20 puede asignar un contexto, dentro de un modelo de contexto, a un símbolo que se va a transmitir. El contexto se puede referir, por ejemplo, a si los valores vecinos del símbolo son distintos de cero o no. Para realizar la codificación CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo que se va a transmitir. Las palabras de código en la codificación VLC se pueden construir de modo que los códigos relativamente más cortos correspondan a símbolos más probables, mientras que los códigos más largos correspondan a símbolos menos probables. De esta manera, el uso de la codificación VLC puede lograr un ahorro en bits con respecto, por ejemplo, al uso de palabras de código de igual longitud para cada símbolo que se va a transmitir. La determinación de la probabilidad se puede basar en un contexto asignado al símbolo.

40 **[0062]** De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 se pueden configurar para realizar cualquiera de o todas las siguientes técnicas que se muestran en la lista en numerada a continuación, solas o en cualquier combinación:

1. Posición del candidato de ATMVP, si se inserta, por ejemplo, como una lista de candidatos de fusión

55 a. Supóngase que los candidatos espaciales y el candidato de TMVP se insertan en una lista de candidatos de fusión en un determinado orden. El candidato de ATMVP se puede insertar en cualquier posición relativamente fija de esos candidatos.

60 i. En una alternativa, por ejemplo, el candidato de ATMVP se puede insertar en la lista de candidatos de fusión después de los dos primeros candidatos espaciales, por ejemplo, A1 y B1;

ii. En una alternativa, por ejemplo, el candidato de ATMVP se puede insertar después de los primeros tres candidatos espaciales, por ejemplo, A1 y B1 y B0;

65 iii. En una alternativa, por ejemplo, el candidato de ATMVP se puede insertar después de los primeros cuatro candidatos, por ejemplo, A1, B1, B0 y A0.

iv. En una alternativa, por ejemplo, el candidato de ATMVP se puede insertar justo antes del candidato de TMVP.

5 v. En una alternativa, por ejemplo, el candidato de ATMVP se puede insertar justo después del candidato de TMVP.

b. De forma alternativa, la posición del candidato de ATMVP en la lista de candidatos se puede señalar en el flujo de bits. Las posiciones de otros candidatos, incluyendo el candidato de TMVP, se pueden indicar adicionalmente.

10 2. La verificación de disponibilidad del candidato de ATMVP se puede aplicar accediendo a un solo conjunto de información de movimiento. Cuando dicho conjunto de información no está disponible, por ejemplo, estando un bloque intracodificado, se considera que todo el candidato de ATMVP no está disponible. En ese caso, el ATMVP no se insertará en la lista de fusión.

15 a. Una posición central, o una sub-PU central se usa únicamente para verificar la disponibilidad del candidato de ATMVP. Cuando se usa una sub-PU central, se elige la sub-PU central para que cubra la posición central (por ejemplo, la posición central 3, con una coordenada relativa de $(W/2, H/2)$ a la muestra superior-izquierda de la PU, en la que $W \times H$ es el tamaño de la PU). Dicha posición o sub-PU central se puede usar junto con el vector temporal para identificar un bloque correspondiente en la imagen de origen de movimiento. Se identifica un conjunto de
20 información de movimiento del bloque que cubre la posición central de un bloque correspondiente.

3. Conjunto representativo de información de movimiento para la PU codificada por ATMVP de una sub-PU.

25 a. Para formar el candidato de ATMVP, primero se forma el conjunto representativo de información de movimiento.

b. Dicho conjunto representativo de información de movimiento se puede derivar de una posición fija o una sub-PU fija. Se puede elegir de la misma manera que la del conjunto de información de movimiento usado para determinar la disponibilidad del candidato de ATMVP, como se describe en la viñeta n.º 2.

30 c. Cuando una sub-PU ha identificado su propio conjunto de información de movimiento y no está disponible, se establece para que sea igual al conjunto representativo de información de movimiento.

35 d. Si el conjunto representativo de información de movimiento se establece como una sub-PU, no se necesita almacenamiento de movimiento adicional en el lado del descodificador para la CTU o corte actual en el peor de los casos.

40 e. Dicho conjunto representativo de información de movimiento se usa en todos los escenarios cuando los procesos de descodificación requieren que toda la PU esté representada por un conjunto de información de movimiento, incluyendo la depuración, de modo que el proceso se use para generar candidatos de fusión bipredictivos combinados.

4. El candidato de ATMVP se depura con el candidato de TMVP y se pueden considerar las interacciones entre TMVP y ATMVP; las técnicas detalladas se enumeran a continuación:

45 a. La depuración de un candidato basado en sub-PU, por ejemplo, un candidato de ATMVP con un candidato normal, se puede realizar usando el conjunto representativo de información de movimiento (como en la viñeta n.º 3) para dicho candidato basado en sub-PU. Si dicho conjunto de información de movimiento es el mismo que un candidato de fusión normal, los dos candidatos se consideran iguales.

50 b. De forma alternativa, además, se realiza una verificación para determinar si el ATMVP contiene múltiples conjuntos diferentes de información de movimiento para múltiples sub-PUs; si se identifican al menos dos conjuntos diferentes, el candidato basado en sub-PU no se usa para la depuración, es decir, se considera que es diferente a cualquier otro candidato; de otro modo se puede usar para depurar (por ejemplo, se puede depurar durante el proceso de depuración).
55

c. De forma alternativa, además, el candidato de ATMVP se puede depurar con los candidatos espaciales, por ejemplo, los candidatos izquierdo y superior solamente, con posiciones indicadas como A1 y B1.

60 d. De forma alternativa, solo se forma un candidato a partir de la referencia temporal, bien siendo el candidato de ATMVP o el candidato de TMVP. Cuando ATMVP está disponible, el candidato es ATMVP; de otro modo, el candidato es TMVP. Dicho candidato se inserta en la lista de candidatos de fusión en una posición similar a la posición de TMVP. En este caso, el número máximo de candidatos se puede mantener sin cambios.

65 i. De forma alternativa, TMVP siempre está deshabilitado incluso cuando ATMVP no está disponible.

ii. De forma alternativa, TMVP se usa solo cuando ATMVP no está disponible.

e. De forma alternativa, cuando ATMVP está disponible y TMVP no está disponible, se usa un conjunto de información de movimiento de un sub-PU como candidato de TMVP. En este caso, además, no se aplica el proceso de depuración entre ATMVP y TMVP.

5 f. De forma alternativa, o adicionalmente, el vector temporal usado para ATMVP también se puede usar para TMVP, de modo que no sea necesario usar la posición inferior derecha o la posición central 3 como se usa para TMVP actual en HEVC.

10 i. De forma alternativa, la posición identificada por el vector temporal y las posiciones inferior derecha y central 3 se consideran conjuntamente para proporcionar un candidato de TMVP disponible.

15 5. Se admiten múltiples verificaciones de disponibilidad para ATMVP para dar posibilidades más altas de que el candidato de ATMVP sea más exacto y eficaz. Cuando el candidato de ATMVP actual de la imagen de origen de movimiento identificado por el primer vector temporal (por ejemplo, como se muestra en la figura 9) no está disponible, otras imágenes se pueden considerar como imagen de origen de movimiento. Cuando se considera otra imagen, puede estar asociada con un segundo vector temporal diferente, o puede estar asociada simplemente con un segundo vector temporal ajustado a escala a partir del primer vector temporal que apunta al candidato de ATMVP no disponible.

20 a. Un segundo vector temporal puede identificar a un candidato de ATMVP en una segunda imagen de origen de movimiento y se puede aplicar la misma verificación de disponibilidad. Si está disponible el candidato de ATMVP según se derive de la segunda imagen de origen de movimiento, se deriva el candidato de ATMVP y no es necesario verificar ninguna otra imagen; de otro modo, se deben verificar otras imágenes como imágenes de origen de movimiento.

25 b. Las imágenes que se van a verificar pueden ser aquellas en las listas de imágenes de referencia de la imagen actual, con un orden dado. Para cada lista, las imágenes se verifican en el orden ascendente del índice de referencia. La lista X se verifica primero y las imágenes en la lista Y (siendo 1-X) siguen.

30 i. La lista X se elige de modo que la lista X sea la lista que contiene la imagen coubicada usada para TMVP.

ii. De forma alternativa, X simplemente se establece en 1 o 0.

35 c. Las imágenes que se van a verificar son aquellas identificadas por vectores de movimiento de los vecinos espaciales, con un orden dado.

6. Una división de la PU a la que se aplica la ATMVP actual puede ser $2N \times 2N$, $N \times N$, $2N \times N$, $N \times 2N$ o divisiones de división de movimiento asimétrico (AMP), tal como $2N \times N/2$.

40 a. De forma alternativa, además, si se pueden permitir otros tamaños de división, también se puede admitir ATMVP, y dicho tamaño puede incluir, por ejemplo, 64×8 .

b. De forma alternativa, el modo solo se puede aplicar a ciertas divisiones, por ejemplo, $2N \times 2N$.

45 7. El candidato de ATMVP se marca como un tipo diferente de candidato de fusión.

50 8. Al identificar un vector (vector temporal como en la primera fase) a partir de vecinos, se pueden verificar en orden múltiples posiciones vecinas, por ejemplo, aquellas usadas en la construcción de listas de candidatos de fusión. Para cada uno de los vecinos, los vectores de movimiento correspondientes a la lista de imágenes de referencia 0 (lista 0) o la lista de imágenes de referencia 1 (lista 1) se pueden verificar en orden. Cuando hay dos vectores de movimiento disponibles, los vectores de movimiento en la lista X se pueden verificar primero y seguidamente la lista Y (siendo Y igual a 1-X), de modo que la lista X es la lista que contiene la imagen coubicada usada para TMVP. En ATMVP, se usa un vector temporal que se añade como un desplazamiento de cualquier posición central de una sub-PU, en la que puede ser necesario desplazar los componentes del vector temporal a números enteros. Dicha posición central desplazada se usa para identificar una unidad más pequeña a la que se pueden asignar vectores de movimiento, por ejemplo, con un tamaño de 4×4 que cubra la posición central actual.

60 a. De forma alternativa, los vectores de movimiento correspondientes a la lista 0 se pueden verificar antes que los correspondientes a la lista 1;

b. De forma alternativa, los vectores de movimiento correspondientes a la lista 1 se pueden verificar antes que los correspondientes a la lista 0;

65 c. De forma alternativa, todos los vectores de movimiento correspondientes a la lista X en todos los vecinos espaciales se verifican en orden, seguidos de los vectores de movimiento correspondientes a la lista Y (siendo Y

igual a 1-X). Aquí, la lista "X" puede ser la lista que indica dónde pertenece la imagen coubicada, o simplemente establecerse en 0 o 1.

d. El orden de los vecinos espaciales puede ser el mismo que el usado en el modo de fusión HEVC.

9. Cuando en la primera fase de identificación de un vector temporal no se incluye información que identifica una imagen de referencia, la imagen de origen de movimiento como se muestra en la figura 9, se puede establecer simplemente para que sea una imagen fija, por ejemplo, la imagen coubicada usada para TMVP.

a. En dicho caso, el vector solo se puede identificar a partir de los vectores de movimiento que apuntan a dicha imagen fija.

b. En dicho caso, el vector solo se puede identificar a partir de los vectores de movimiento que apuntan a cualquier imagen pero además ajustados a escala hacia la imagen fija.

10. Cuando en la primera fase de identificación de un vector consiste en identificar una imagen de referencia, la imagen de origen de movimiento como se muestra en la figura 9, una o más de las siguientes verificaciones adicionales se pueden aplicar para un vector de movimiento candidato.

a. Si el vector de movimiento está asociado con una imagen o corte que está intracodificado, dicho vector de movimiento se considera no disponible y no se puede usar para convertirlo al vector.

b. Si el vector de movimiento identifica un bloque intra (por ejemplo, añadiendo la coordenada central actual con el vector de movimiento) en la imagen asociada, dicho vector de movimiento se considera no disponible y no se puede usar para convertirlo al vector.

11. Cuando se encuentra en la primera fase de identificación de un vector, los componentes del vector se pueden establecer en (mitad del ancho de la PU actual, mitad de la altura de la PU actual), de modo que identifique una posición de píxel inferior derecha en la imagen de origen de movimiento. Aquí (x, y) indica un componente horizontal y vertical de un vector de movimiento.

a. De forma alternativa, los componentes del vector se pueden establecer en (suma(mitad del ancho de la PU actual, M), suma(mitad de la altura de la PU actual, N)) donde la función suma(a, b) devuelve la suma de a y b. En un ejemplo, cuando la información de movimiento se almacena en la unidad 4x4, M y N se establecen ambas para que sean iguales a 2. En otro ejemplo, cuando la información de movimiento se almacena en la unidad 8x8, M y N se establecen ambas para que sean iguales a 4.

12. El tamaño de subbloque/sub-PU cuando se aplica ATMVP se señala en un conjunto de parámetros, por ejemplo, conjunto de parámetros de secuencia del conjunto de parámetros de imagen. El tamaño varía desde el tamaño de PU menor hasta el tamaño de la CTU. El tamaño también se puede predefinir o señalar. El tamaño puede ser, por ejemplo, tan pequeño como 4x4. De forma alternativa, el tamaño del subbloque/sub-PU se puede derivar basado en el tamaño de PU o CU. Por ejemplo, el subbloque/sub-PU se puede establecer igual a $\max(4 \times 4, (\text{ancho de CU}) \gg M)$. El valor de M se puede predefinir o señalar en el flujo de bits.

13. El número máximo de candidatos a la fusión se puede incrementar en 1 debido al hecho de que ATMVP se puede considerar como un nuevo candidato de fusión. Por ejemplo, en comparación con HEVC, que toma hasta 5 candidatos en una lista de candidatos de fusión después de la depuración, el número máximo de candidatos de fusión se puede incrementar a 6.

a. De forma alternativa, la depuración con el candidato de TMVP convencional o la unificación con el candidato de TMVP convencional se puede realizar para ATMVP de modo que el número máximo de candidatos de fusión se pueda mantener sin cambios.

b. De forma alternativa, cuando se identifica que ATMVP está disponible, un candidato vecino espacial se excluye de la lista de candidatos de fusión, por ejemplo, se excluye el último candidato vecino espacial en orden de obtención.

14. Cuando se considera que múltiples vectores de movimiento vecinos espaciales derivan el vector temporal, se puede calcular una similitud de vector de movimiento basado en los vectores de movimiento vecinos de la PU actual, así como en los vectores de movimiento vecinos identificados por un vector temporal específico que se establece igual a un vector de movimiento. El que dé lugar a la similitud más alta de movimiento se puede elegir como el vector temporal final.

a. En una alternativa, para cada vector de movimiento desde una posición vecina N, el vector de movimiento identifica un bloque (mismo tamaño que la PU actual) en la imagen de origen de movimiento, en el que su posición

vecina N contiene un conjunto de información de movimiento. Este conjunto de vectores de movimiento se compara con el conjunto de información de movimiento como en la posición vecina N del bloque actual.

b. En otra alternativa, para cada vector de movimiento desde una posición vecina N, el vector de movimiento identifica un bloque en la imagen de origen de movimiento, en el que sus posiciones vecinas contienen múltiples conjuntos de información de movimiento. Estos múltiples conjuntos de vectores de movimiento se comparan con los múltiples conjuntos de información de movimiento de las posiciones vecinas de la PU actual en las mismas posiciones relativas. Se calcula una similitud de información de movimiento. Por ejemplo, la PU actual tiene los siguientes conjuntos de información de movimiento de A1, B1, A0 y B0, indicados como MIA1, MIB1, MIA0 y MIB0. Para un vector temporal TV, identifica un bloque correspondiente a la PU en la imagen de origen de movimiento. Dicho bloque tiene información de movimiento de las mismas posiciones relativas A1, B1, A0 y B0, y se indican como TMIA1, TMIB1, TMIA0 y TMIB0. La similitud de movimiento según se determine por TV se calcula como

$MStv = \sum_{N \in \{A1, B1, A0, B0\}} MVSIM(\llbracket MI \rrbracket_{-N}, \llbracket TMI \rrbracket_{-N})$, en la que MVSIM define la similitud entre dos conjuntos de información de movimiento.

c. En los dos casos anteriores, se puede usar la similitud de movimiento MVSIM, en la que los dos parámetros de entrada son los dos conjuntos de información de movimiento, conteniendo cada uno hasta dos vectores de movimiento y dos índices de referencia. Cada par de los vectores de movimiento en la lista X están realmente asociados con imágenes de referencia en diferentes listas X de diferentes imágenes, la imagen actual y la imagen de origen de movimiento. Para cada uno de los dos vectores de movimiento MVXN y TMVXN (siendo X igual a 0 o 1), la diferencia del vector de movimiento MVDXN se puede calcular como MVXN - TMVXN. Después, la

diferencia MVSIMX se calcula como, por ejemplo, $abs(\llbracket MVDX \rrbracket_{-N}[0]) + abs(\llbracket MVDX \rrbracket_{-N}[1])$, o $(\llbracket MVDX \rrbracket_{-N}[0] * \llbracket MVDX \rrbracket_{-N}[0] + \llbracket MVDX \rrbracket_{-N}[1] * \llbracket MVDX \rrbracket_{-N}[1])$. Si ambos conjuntos de información de movimiento contienen vectores de movimiento disponibles, MVSIM se establece igual a MVSIM0 + MVSIM1.

i. A fin de tener un cálculo unificado de la diferencia de movimiento, ambos vectores de movimiento se deben ajustar a escala hacia la misma imagen fija, que puede ser, por ejemplo, la primera imagen de referencia RefPicListX[0] de la lista X de la imagen actual.

ii. Si la disponibilidad del vector de movimiento en la lista X del primer conjunto y la disponibilidad del vector de movimiento en la lista X del segundo conjunto son diferentes, es decir, un índice de referencia es -1 mientras que el otro no, dichos dos conjuntos de movimiento de la información se consideran no similares en la dirección X. Si los dos conjuntos no son similares en ambos conjuntos, la función MVSIM final puede devolver un gran valor T, que se puede, por ejemplo, considerar como infinito.

iii. De forma alternativa, para un par de conjuntos de información de movimiento, si uno se predice de la lista X (siendo X igual a 0 o 1) pero no de la lista Y (siendo Y igual a 1-X) y el otro tiene el mismo estado, se puede usar una ponderación entre 1 y 2 (por ejemplo, MVSIM es igual a MVSIMX *1.5). Cuando un conjunto solo se predice de la lista X y el otro solo se predice de la lista Y, MVSIM se establece en el gran valor T.

iv. De forma alternativa, para cualquier conjunto de información de movimiento, siempre que esté disponible un vector de movimiento, se producirán ambos vectores de movimiento. En el caso de que solo esté disponible un vector de movimiento (correspondiente a la lista X), se ajusta a escala para formar el vector de movimiento correspondiente a la otra lista Y.

d. De forma alternativa, el vector de movimiento se puede medir en base a las diferencias entre los píxeles vecinos de la PU actual y los píxeles vecinos del bloque (mismo tamaño que la PU actual) identificados por el vector de movimiento. El vector de movimiento que da lugar a la diferencia más pequeña se puede elegir como el vector temporal final.

15. Al derivar el vector temporal del bloque actual, los vectores de movimiento y/o vectores temporales de bloques vecinos que están codificados con ATMVP pueden tener una prioridad más alta que los vectores de movimiento de otros bloques vecinos.

a. En un ejemplo, solo los vectores temporales de los bloques vecinos se verifican primero, y el primero disponible se puede establecer en el vector temporal del bloque actual. Solo cuando dichos vectores temporales no están presentes, se verifican además los vectores de movimiento normales. En este caso, los vectores temporales para bloques codificados por ATMVP se deben almacenar.

b. En otro ejemplo, solo los vectores de movimiento de los bloques vecinos codificados por ATMVP se verifican primero, y el primero disponible se puede establecer en el vector temporal del bloque actual. Solo cuando dichos vectores temporales no están presentes, se verifican además los vectores de movimiento normales.

c. En otro ejemplo, solo los vectores de movimiento de los bloques vecinos codificados por ATMVP se verifican primero, y el primero disponible se puede establecer en el vector temporal del bloque actual. Si dichos vectores de movimiento no están disponibles, la verificación del vector temporal continúa de manera similar a la viñeta 15a.

5 d. En otro ejemplo, los vectores temporales de los bloques vecinos se verifican primero, el primero disponible se puede establecer en el vector temporal del bloque actual. Si dichos vectores de movimiento no están disponibles, la verificación del vector temporal continúa de manera similar a la viñeta 15b.

10 e. En otro ejemplo, los vectores temporales y los vectores de movimiento de los bloques vecinos codificados por ATMVP se verifican primero, el primero disponible se puede establecer en el vector temporal del bloque actual. Solo cuando dichos vectores temporales y vectores de movimiento no están presentes, se verifican además los vectores de movimiento normales.

15 16. Cuando se considera que múltiples vectores de movimiento vecinos espaciales derivan el vector temporal, se puede elegir un vector de movimiento de modo que minimice la distorsión que se calcula a partir del dominio de píxel, por ejemplo, se puede usar la equiparación entre plantillas para derivar el vector temporal de modo que se selecciona el que dé lugar a un coste de equiparación mínimo como el vector temporal final.

20 17. La derivación de un conjunto de información de movimiento a partir de un bloque correspondiente (en la imagen de origen de movimiento) se realiza de manera que cuando un vector de movimiento esté disponible en el bloque correspondiente para cualquier lista X (indica que el vector de movimiento sea MVX), para la sub-PU actual del candidato de ATMVP, el vector de movimiento se considera como disponible para la lista X (ajustando a escala el MVX). Si el vector de movimiento no está disponible en el bloque correspondiente para cualquier lista X, se considera que el vector de movimiento no está disponible para la lista X.

25 a. De forma alternativa, cuando el vector de movimiento en el bloque correspondiente no está disponible para la lista X pero está disponible para la lista 1 - X (indicado 1 - X por Y e indica que el vector de movimiento sea MVY), el vector de movimiento todavía se considera disponible para la lista X (ajustando a escala el MVY hacia la imagen de referencia diana en la lista X).

30 b. De forma alternativa, o además, cuando ambos vectores de movimiento en el bloque correspondiente para la lista X y la lista Y (igual a 1-X) están disponibles, los vectores de movimiento de la lista X y de la lista Y no se usan necesariamente para ajustar a escala y generar los dos vectores de movimiento de una sub-PU actual mediante ajuste a escala.

35 i. En un ejemplo, al formular el candidato de ATMVP, la verificación de bajo retardo como se hace en TMVP se aplica a cada sub-PU. Si para cada imagen (indicada por refPic) en cada lista de imágenes de referencia del corte actual, el valor del recuento de orden de imagen (POC) de refPic es más pequeño que el POC del corte actual, el corte actual se considera con modo de bajo retardo. En este modo de bajo retardo, los vectores de movimiento de la lista X y de la lista Y se ajustan a escala para generar los vectores de movimiento de una sub-PU actual para la lista X y la lista Y, respectivamente. Cuando no está en el modo de bajo retardo, solo se elige un vector de movimiento MVZ de MVX o MVY y se ajusta a escala para generar los dos vectores de movimiento para una sub-PU actual. Similar a TMVP, en dicho caso Z se establece igual a `collocated_from_10_flag`, lo que significa que depende de si la imagen coubicada como en TMVP está en la lista X o en la lista Y de la imagen actual. De forma alternativa, Z se establece como sigue: si la imagen de origen de movimiento se identifica a partir de la lista X, se establece Z en X. De forma alternativa, además, cuando las imágenes de origen de movimiento pertenecen a ambas listas de imágenes de referencia, y `RefPicList0[idx0]` es la imagen de origen de movimiento que está presente primero en la lista 0 y `RefPicList(1)[idx1]` es la imagen de origen de movimiento que está presente primero en la lista 1, se establece Z en 0 si `idx0` es más pequeño o igual que `idx1`, y de otro modo se establece en 1.

40 18. La imagen de origen de movimiento se puede señalar, por ejemplo, generada por el codificador de vídeo 20 en un flujo de bits codificado. En detalle, un indicador que indica si la imagen de origen de movimiento es de la lista 0 o de la lista 1 se señala para un corte B. De forma alternativa, además, se puede señalar un índice de referencia a una lista 0 o a una lista 1 de la imagen actual para identificar la imagen de origen de movimiento.

55 19. Al identificar un vector temporal, se considera que un vector no está disponible (por tanto, se pueden considerar otros) si apunta a un bloque intracodificado en la imagen de origen de movimiento asociada.

60 **[0063]** La implementación de las diversas técnicas de esta divulgación se analiza a continuación. Se supone que la ATMVP se implementa encima de la versión 1 de HEVC. Puede que no se aplique la compresión de movimiento a imágenes de referencia y se pueden habilitar bloques más pequeños con compensación de movimiento bidireccional.

65 **[0064]** Señalización de ATMVP en SPS:

`atmvp_sub_pu_size` puede estar presente en SPS.

atmvp_sub_pu_size puede especificar el tamaño de las sub-PU de una PU codificada con el modo ATMVP. Es en el intervalo de 2 a 6, inclusive. El tamaño de sub-PU para ATMVP, (spuWidth, spuHeight) se deriva como $\min(w, 1 << \text{atmvp_sub_pu_size})$ by $\min(h, 1 << \text{atmvp_sub_pu_size})$, en la que $w \times h$ es el tamaño de una PU actual.

5 **[0065]** De forma alternativa, tanto el ancho como la altura de los tamaños de sub-PU se señalizan por separado en SPS.

10 **[0066]** De forma alternativa, los tamaños de sub-PU se señalizan frente al tamaño de la CTU o el tamaño de unidad de codificación más pequeño.

[0067] Se deriva una variable atmvpEnableFlag para que sea igual a 1 si atmvp_sub_pu_size es más pequeño que un tamaño de CTU (por ejemplo, 6 como en la versión 1 de HEVC), y 0, de otro modo.

15 **[0068]** Señalización de ATMVP en el encabezado del corte: five_minus_max_num_merge_cand especifica el número máximo de candidatos MVP de fusión admitidos en el corte restado de 5. El número máximo de candidatos MVP de fusión, MaxNumMergeCand se deriva como:

$$\text{MaxNumMergeCand} = (\text{atmvpEnableFlag? } 6: 5) - \text{five_minus_max_num_merge_cand} \quad (7-41)$$

20 **[0069]** El valor de five_minus_max_num_merge_cand se limitará de modo que MaxNumMergeCand esté en el intervalo de 1 a (atmvpEnableFlag? 6: 5), inclusive.

25 **[0070]** De forma alternativa, five_minus_max_num_merge_cand se cambia a six_minus_max_num_merge_cand y la semántica es como sigue:

six_minus_max_num_merge_cand especifica el número máximo de candidatos MVP de fusión admitidos en el corte restado de 6. El número máximo de candidatos MVP de fusión, MaxNumMergeCand se deriva como

$$\text{MaxNumMergeCand} = 6 - \text{six_minus_max_num_merge_cand} \quad (7-41)$$

[0071] De forma alternativa, max_num_merge_cand_minus1 se señala directamente.

35 **[0072]** En algunos ejemplos, otros cambios sintácticos son innecesarios, y un candidato de ATMVP se identifica mediante un idx de fusión, que puede estar en el intervalo de 0 a 5, inclusive.

[0073] Procesos de descodificación relacionados con ATMVP: Los siguientes procesos de descodificación se pueden implementar, por ejemplo, mediante el descodificador de vídeo 30, para formular un candidato de ATMVP e incluirlo como parte de la lista de candidatos de fusión:

40 **[0074]** Identificación del vector temporal de la primera fase:

[0075] Establecer la variable mtSrcPOC en el valor POC de la imagen coubicada usada en TMVP, tV en un vector cero y atmvpAvaFlag se establece en 0.

45 **[0076]** Para cada una de las posiciones N de las posiciones vecinas espaciales, siendo A1, B1, B0, A0 y B2, se aplica lo siguiente:

- dir se establece igual a collocated_from_10_flag;
- Para X siendo igual a dir a través de (1-dir), inclusive, si el corte actual es un corte B, o simplemente X siendo igual a 0 si el corte actual no es un corte B, se aplica lo siguiente:
 - Cuando el bloque vecino N está disponible y no está intracodificado, y RefIdxX[N] es mayor o igual a 0 (indica que MVLX[N] y RefIdxX[N] son el vector de movimiento y el índice de referencia del bloque vecino N correspondiente a RefPicListX), se aplican las siguientes etapas en orden:
 - mtSrcPOC se establece igual al valor POC de RefPicListX[RefIdxX[N]];
 - tV se establece igual a MVLX[N];
 - atmvpAvaFlag se establece en 1;

- terminar este proceso.

[0077] Identificación de un origen de movimiento disponible:

5 Establecer una lista de imágenes para que sea CanPicATMVP para que sea una lista vacía.

CanPicATMVP[0] se establece para que sea la imagen con un valor POC igual a mtSrcPOC. i se establece igual a 1;

10 MotionSrcPic es una imagen vacía y se puede actualizar como se especifica a continuación.

[0078] Para cada una de la lista X de imágenes de referencia disponibles, se aplica lo siguiente:

- dir se establece igual a collocated_from_10_flag;
 - Para X siendo igual a dir a través de (1-dir), inclusive, si el corte actual es un corte B, o simplemente X siendo igual a 0 si el corte actual no es un corte B, se aplica lo siguiente:
 - Para cada idx desde 0 hasta num_ref_active_1X_minus1;
 - CanPicATMVP[i++] = RefPicListX[idx];

[0079] Dejar que (CurrPosX, CurrPosY) sea la coordenada de la posición de píxel superior izquierda de la PU actual.

25 **[0080]** Para n siendo igual a 0 hasta i, inclusive, se aplica lo siguiente.

- Si n no es igual a 0, ajustar a escala el tV hacia la imagen CanPicATMVP[n] para obtener un tScaledVector, en el que las imágenes pertinentes para el tV son la imagen actual y la CanPicATMVP[0], y las imágenes pertinentes para el vector de destino tScaledVector son la imagen actual y la CanPicATMVP[i];
- De otro modo (n es igual a 0), tScaledVector se establece igual a tV.
- Obtener la información de movimiento del bloque correspondiente a la sub-PU central de CanPicATMVP[n] como sigue:
 - $centerPosX = CurrPosX + ((tScaledVector[0] + 2) \gg 2)$;
 - $centerPosY = CurrPosY + ((tScaledVector[1] + 2) \gg 2)$;
 - Dejar que (centerPosX, centerPosY) sea la posición que identifica el bloque correspondiente de la sub-PU central y el tamaño de la PU actual para que sea ancho por alto.
 - $centerPosX += CurrPosX + ((width / spuWidth) \gg 1) * spuWidth + (min(spuWidth, width) \gg 1)$;
 - $centerPosY += CurrPosY + ((height/spuHeight) \gg 1) * spuHeight + (min(spuHeight, height) \gg 1)$;
 - Invocar el proceso de obtención de información de movimiento que toma la información de movimiento, con una imagen mtnSrcPic siendo igual a CanPicATMVP[n], y una posición (posX, posY) siendo igual a (centerPosX, centerPosY) como entrada y un indicador disponible de movimiento de sub-PU SubPuMtnAvaFlag, un par de índices de referencia sColRefIdx0 y sColRefIdx1, y un par de vectores de movimiento, sColMVO, sColMV1 como salida.
 - Si SubPuMtnAvaFlag es igual a 1, se aplica lo siguiente:
 - MotionSrcPic se establece en CanPicATMVP[n]
 - tV se establece para ser tScaledVector
 - terminar este bucle.

60 **[0081]** Proceso de obtención de información de movimiento:

La entrada de este proceso es una imagen mtnSrcPic y una posición (posX, posY) con la imagen, y la salida de este proceso es el indicador disponible de movimiento mtnAvaFlag, un par de índices de referencia refIdx0 y refIdx1, y un par de vectores de movimiento, mv0, mv1.

La (posX, posY) se recorta primero para estar dentro de la imagen mtnSrcPic.

Se identifica el bloque 4x4 (u otro tamaño más pequeño que almacena información de movimiento) blkT, que contiene la posición (posX, posY).

5 mtnAvaFlag se establece igual a 0.

Si blkT no está intracodificado y su información de movimiento contiene blkTRefIdxO, blkTRefIdx1, blkTMvO y blkTMv1, se aplica lo siguiente.

10 • Cuando blkTRefIdxO o blkTRefIdx1 es mayor o igual a 0, mtnAvaFlag se establece igual a 1 y lo siguiente se aplica para X siendo igual a 0 y 1.

15 ◦ refIdxX se establece igual a blkTRefIdxX

◦ mvX se establece igual a blkTMvX

[0082] Generación de movimiento sub-PU para ATMVP:

20 Si SubPuMtnAvaFlag es igual a 1, se invoca el siguiente proceso:

• Para cada uno de los sub-PU (por ejemplo, en orden de exploración ráster), se aplica lo siguiente.

25 ◦ Indicar el índice horizontal y el índice vertical de la sub-PU actual como k y l respectivamente, en el que k varía de 0 a width/spuWidth-1, inclusive y l varía de 0 a height/spuHeight-1, inclusive. Por ejemplo, si una PU de 16x16 se divide en cuatro sub-PU de 8x8, los valores (k, l) de las cuatro sub-PU en orden de exploración ráster son (0, 0), (1, 0), (0, 1) y (1, 1) respectivamente.

30 ◦ Las coordenadas de Sub-PU (tempPosX, tempPosY) se calculan como (tempPosX, tempPosY) = (CurrPosX, CurrPosY) + (k * spuWidth, l * spuHeight).

◦ $tempPosX += ((tV[0] + 2) >> 2);$

35 ◦ $tempPosY += ((tV[1] + 2) >> 2);$

40 ◦ Invocar el proceso de obtención de información de movimiento que toma la información de movimiento, con una imagen mtnSrcPic siendo igual a MotionSrcPic, y una posición (posX, posY) siendo igual a (tempPosX, tempPosY) como entrada y un indicador de movimiento sub-PU disponible currSubPuMtnAvaFlag, un par de índices de referencia currSubRefIdx0 y currSubRefIdx1, y un par de vectores de movimiento, currSubMV0, currSubMV1 como salida.

◦ Cuando currSubPuMtnAvaFlag es igual a 0, para X igual a 0 y 1, inclusive, currSubRefIdxX se establece igual a cColRefIdxX y currSubMVX se establece igual a cColMVX.

45 ◦ Para X siendo igual a 0 y 1, inclusive, ajustar a escala el vector de movimiento currSubMVX hacia la imagen de referencia diana predeterminada de la imagen actual, que es RefPicListX[0], similar a TMVP. Indicar el índice de referencia derivado y el vector de movimiento para la sub-PU actual como cSpuRefIdxX y cSpuMVX y se derivan como sigue:

50 ◦ $cSpuRefIdxX = (currSubRefIdxX >= 0 ? 0 : -1);$

◦ cSpuMVX se establece para ser el vector ajustado a escala de currSubMVX, similar a TMVP.

55 **[0083]** El conjunto representativo de información de movimiento, aRefIdxX y aMVX (para X siendo igual a 0 o 1) para este candidato de ATMVP se deriva como sigue:

• $aRefIdxX = (cColRefIdxX >= 0 ? 0 : -1);$

• aMVX se establece para ser el vector ajustado a escala de cColMVX, similar a TMVP.

60 *De forma alternativa, el proceso de obtención de movimiento y el proceso de ajuste a escala de movimiento son los mismos (o similares) que en TMVP de HETIC versión 1, es decir, la subcláusula 8.5.3.2.8 de HETIC versión 1: "Derivation process for collocated motion vectors" se aplica para reemplazar el texto resaltado en esta subsección. En este caso, el proceso de obtención de movimiento y ajuste a escala de movimiento como en TMVP (subcláusula 8.5.3.2.8) reemplaza el proceso de obtención de movimiento y el proceso de ajuste a escala de movimiento definidos anteriormente (incluyendo lo indicado por el texto en cursiva).*

65

[0084] Inserción del candidato de ATMVP en una lista de candidatos de fusión:

[0085] Cuando SubPuMtnAvaFlag es igual a 1, el candidato de ATMVP se inserta en el candidato de fusión después de que el candidato A0 (o de forma alternativa B0) se someta a prueba y posiblemente se inserte en la lista de candidatos de fusión.

[0086] Se considera que la información de movimiento para este candidato está formada por aRefldxX y aMVX (con X siendo igual a 0 o 1).

[0087] Cuando el candidato de TMVP está disponible, se compara además con la información representativa del candidato de ATMVP (aRefldxX y aMVX); solo si el candidato de TMVP tiene un refldxX que no es igual a aRefldxX o un vector de movimiento que no es igual a aMVX (siendo X igual a 0 o 1), se inserta además en la lista de candidatos a la fusión.

[0088] Cuando todos los candidatos, incluyendo el candidato de ATMVP, se consideran de forma transparente como representados por un solo conjunto de información de movimiento (hasta dos índices de referencia y dos vectores de movimiento asociados), se usa la información representativa para el candidato de ATMVP. Por ejemplo, en el proceso de derivación para candidatos de fusión bipredictivos combinados, se usa la información de movimiento representativa de la ATMVP.

[0089] Además, cada candidato de fusión se adjunta con una marca (puede ser un indicador o un tipo) que indica si dicho candidato es un candidato de ATMVP.

[0090] Además, para un candidato de ATMVP, los conjuntos de información de movimiento, indicados anteriormente como cSpuRefldxX y cSpuMVX para X siendo igual a 0 y 1 y para cada sub-PU se deben almacenar para que el PU actual se descodifique.

[0091] Compensación de movimiento basada en el candidato de ATMVP: Cuando una PU actual se codifica con el modo de fusión y merge_idx especificó un candidato indicado como candidato de ATMVP, para cada sub-PU, se deriva la información de movimiento cSpuRefldxX y cSpuMVX (para X siendo igual a 0 y 1) y se usa para realizar compensación de movimiento por el sub-PU actual. Después de realizar la compensación de movimiento, la descodificación residual y otros procesos se realizan de la misma manera que otros modos inter.

[0092] Además, el codificador de vídeo 20 puede enviar datos sintácticos, tales como datos sintácticos basados en bloques, datos sintácticos basados en tramas y datos sintácticos basados en GOP, al descodificador de vídeo 30, por ejemplo, en un encabezado de trama, un encabezado de bloque, un encabezado de corte o un encabezado de GOP. Los datos sintácticos de GOP pueden describir una cantidad de tramas en el GOP respectivo, y los datos sintácticos de trama pueden indicar un modo de codificación/predicción usado para codificar la trama correspondiente.

[0093] Tanto el codificador de vídeo 20 como el descodificador de vídeo 30 se pueden implementar como cualquiera de una variedad de circuitos codificadores o descodificadores adecuados, según sea aplicable, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), arreglos de compuertas programables en el campo (FPGA), circuitos de lógica discreta, programas informáticos, hardware, firmware o cualquier combinación de los mismos. Tanto el codificador de vídeo 20 como el descodificador de vídeo 30 pueden estar incluidos en uno o más codificadores o descodificadores, cada uno de los cuales puede estar integrado como parte de un codificador/descodificador (CÓDEC) de vídeo combinado. Un dispositivo que incluye el codificador de vídeo 20 y/o el descodificador de vídeo 30 puede comprender un circuito integrado, un microprocesador y/o un dispositivo de comunicación inalámbrica, tal como un teléfono celular.

[0094] La figura 2 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo 20 que puede implementar técnicas para la predicción de vector de movimiento temporal avanzada (ATMVP). El codificador de vídeo 20 puede realizar la intracodificación y la intercodificación de bloques de vídeo dentro de cortes de vídeo. La intracodificación se basa en la predicción espacial para reducir o retirar la redundancia espacial en el vídeo dentro de una trama o imagen de vídeo dado. La intercodificación se basa en la predicción temporal para reducir o retirar la redundancia temporal en el vídeo dentro de tramas o imágenes contiguos de una secuencia de vídeo. Intramodo (modo I) se puede referir a cualquiera de varios modos de codificación de base espacial. Los intermodos, tales como la predicción unidireccional (modo P) o la bipredicción (modo B), se pueden referir a cualquiera de varios modos de codificación de base temporal.

[0095] Como se muestra en la figura 2, el codificador de vídeo 20 recibe un bloque de vídeo actual dentro de una trama de vídeo que se va a codificar. En el ejemplo de la figura 2, el codificador de vídeo 20 incluye una unidad de selección de modo 40, una memoria de imágenes de referencia 64, un sumador 50, una unidad de procesamiento de transformada 52, una unidad de cuantificación 54 y una unidad de codificación por entropía 56. A su vez, la unidad de selección de modo 40 incluye la unidad de compensación de movimiento 44, la unidad de estimación de movimiento 42, la unidad de intrapredicción 46 y la unidad de división 48. Para la reconstrucción de bloques de vídeo, el codificador

de vídeo 20 incluye también una unidad de cuantificación inversa 58, una unidad de transformada inversa 60 y un sumador 62. También se puede incluir un filtro de eliminación de bloques (no mostrado en la figura 2) para filtrar fronteras de bloques, para eliminar distorsiones de efecto pixelado del vídeo reconstruido. Si se desea, el filtro de eliminación de bloques filtrará habitualmente la salida del sumador 62. También se pueden usar filtros adicionales (en el bucle o tras el bucle), además del filtro de eliminación de bloques. Dichos filtros no se muestran por razones de brevedad pero, si se desea, pueden filtrar la salida del sumador 50 (como un filtro en el bucle).

[0096] Durante el proceso de codificación, el codificador de vídeo 20 recibe una trama o un corte de vídeo que se va a codificar. La trama o el corte se pueden dividir en múltiples bloques de vídeo. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 realizan la codificación interpredictiva del bloque de vídeo recibido con respecto a uno o más bloques en una o más tramas de referencia para proporcionar predicción temporal. La unidad de intrapredicción 46, de forma alternativa, puede llevar a cabo la codificación intrapredictiva del bloque de vídeo recibido con respecto a uno o más bloques vecinos en la misma trama o corte que el bloque que se va a codificar para proporcionar predicción espacial. El codificador de vídeo 20 puede realizar múltiples pases de codificación, por ejemplo, para seleccionar un modo adecuado de codificación para cada bloque de datos de vídeo.

[0097] Además, la unidad de división 48 puede dividir bloques de datos de vídeo en subbloques, basado en la evaluación de los esquemas de división previos en los pases de codificación previos. Por ejemplo, la unidad de división 48 puede dividir inicialmente una trama o un corte en varias LCU, y dividir cada una de las LCU en varias sub-CU, basado en un análisis de velocidad-distorsión (por ejemplo, una optimización de velocidad-distorsión). La unidad de selección de modo 40 puede producir además una estructura de datos de árbol cuaternario, indicativa de la división de una LCU en las sub-CU. Las CU de nodos hojas del árbol cuaternario pueden incluir una o más PU y una o más TU.

[0098] La unidad de selección de modo 40 puede seleccionar uno de los modos de codificación, intra o inter, por ejemplo, basado en los resultados de errores, y proporciona el bloque intracodificado o intercodificado resultante al sumador 50 para generar datos de bloques residuales, y al sumador 62 para reconstruir el bloque codificado para su uso como una trama de referencia. La unidad de selección de modo 40 también proporciona elementos sintácticos, tales como vectores de movimiento, indicadores de intramodo, información de división y otra información sintáctica de este tipo, a la unidad de codificación por entropía 56.

[0099] La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar altamente integradas, pero se ilustran por separado con propósitos conceptuales. La estimación del movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generación de vectores de movimiento, que estiman el movimiento para los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo dentro de una trama o imagen de vídeo actual, con respecto a un bloque predictivo dentro de una trama de referencia (u otra unidad codificada), con respecto al bloque actual que se está codificando dentro de la trama actual (u otra unidad codificada). Un bloque predictivo es un bloque que se encuentra como estrechamente coincidente con el bloque que se va a codificar, en lo que respecta a la diferencia de píxeles, lo cual se puede determinar mediante la suma de diferencia absoluta (SAD), la suma de diferencia al cuadrado (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para posiciones fraccionarias de píxeles de imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones fraccionarias de píxel de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento con respecto a las posiciones de píxel completo y las posiciones de píxel fraccionario, y emitir un vector de movimiento con una precisión de píxel fraccionario.

[0100] La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de vídeo en un corte intercodificado, comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia se puede seleccionar a partir de una primera lista (lista 0) de imágenes de referencia o una segunda lista (lista 1) de imágenes de referencia, cada una de las cuales identifica una o más imágenes de referencia almacenadas en una memoria de imágenes de referencia 64. La unidad de estimación de movimiento 42 envía el vector de movimiento calculado a la unidad de codificación por entropía 56 y a la unidad de compensación de movimiento 44.

[0101] La compensación de movimiento, llevada a cabo por la unidad de compensación de movimiento 44, puede implicar obtener o generar el bloque predictivo basado en el vector de movimiento determinado por la unidad de estimación de movimiento 42. De nuevo, la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 se pueden integrar funcionalmente, en algunos ejemplos. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, la unidad de compensación de movimiento 44 puede localizar el bloque predictivo al que apunta el vector de movimiento en una de las listas de imágenes de referencia. El sumador 50 forma un bloque de vídeo residual restando los valores de píxel del bloque predictivo a los valores de píxel del bloque de vídeo actual que se está codificando, generando valores de diferencias de píxel, como se expone más adelante. En general, la unidad de estimación de movimiento 42 realiza la estimación de movimiento con respecto a los componentes de luma, y la unidad de compensación de movimiento 44 usa los vectores de movimiento calculados en base a los componentes de luma, tanto para los componentes de croma como para los componentes de luma. La unidad de selección de modo

40 también puede generar elementos sintácticos asociados a los bloques de vídeo y al corte de vídeo para su uso por el decodificador de vídeo 30 en la decodificación de los bloques de vídeo del corte de vídeo.

5 **[0102]** El codificador de vídeo 20 se puede configurar para realizar cualquiera de las diversas técnicas de esta divulgación analizadas anteriormente con respecto a la figura 1. Por ejemplo, la unidad de compensación de movimiento 44 se puede configurar para codificar información de movimiento para un bloque de datos de vídeo usando AMVP o modo de fusión de acuerdo con las técnicas de esta divulgación.

10 **[0103]** Suponiendo que la unidad de compensación de movimiento 44 elige realizar el modo de fusión, la unidad de compensación de movimiento 44 puede formar una lista de candidatos que incluye un conjunto de candidatos de fusión. La unidad de compensación de movimiento 44 puede añadir candidatos a la lista de candidatos basado en un orden predeterminado particular. En un ejemplo, la unidad de compensación de movimiento 44 añade los candidatos a la lista de candidatos en el orden de A1, B1, B0, A0, a continuación un candidato de predicción de vector de movimiento temporal avanzada (ATMVP). La unidad de compensación de movimiento 44 también puede añadir
15 candidatos adicionales y realizar la depuración de la lista de candidatos, como se analiza anteriormente. Finalmente, la unidad de selección de modo 40 puede determinar cuál de los candidatos se usará para codificar información de movimiento del bloque actual, y codificar un índice de fusión que representa el candidato seleccionado.

20 **[0104]** Además, en algunos ejemplos, la unidad de compensación de movimiento 44 puede determinar primero si el candidato de ATMVP está disponible. Por ejemplo, la unidad de compensación de movimiento puede determinar un bloque correspondiente al bloque actual en una imagen de referencia y determinar si está disponible la información de movimiento para el bloque correspondiente. La unidad de compensación de movimiento 44 puede a continuación determinar que el candidato de ATMVP (es decir, el bloque correspondiente) está disponible cuando la información de movimiento está disponible para el bloque correspondiente. En algunos ejemplos, la unidad de compensación de
25 movimiento 44 puede determinar que la información de movimiento está disponible para el bloque correspondiente cuando se predice todo el bloque correspondiente (por ejemplo, un bloque de posición central, como se muestra en la figura 5a a continuación) sin el uso de la intrapredicción, pero no está disponible cuando al menos parte del bloque correspondiente se predice usando la intrapredicción.

30 **[0105]** De manera similar, en algunos ejemplos, la unidad de compensación de movimiento 44 puede determinar cuál de los dos candidatos de ATMVP potenciales se debe usar como el candidato de ATMVP finalmente añadido a la lista de candidatos. Por ejemplo, la unidad de compensación de movimiento 44 puede formar un primer vector de movimiento temporal con respecto al bloque actual que identifica un primer candidato de ATMVP en una primera imagen de origen de movimiento, es decir, una primera imagen de referencia. Si la información de movimiento no está
35 disponible para el primer candidato de ATMVP, la unidad de compensación de movimiento 44 puede determinar si la información de movimiento está disponible para un segundo candidato de ATMVP diferente. El segundo candidato de ATMVP se puede identificar usando el mismo vector de movimiento temporal que se refiere a una segunda imagen de referencia diferente, un vector de movimiento temporal diferente que se refiere a la misma (es decir, la primera) imagen de referencia, o un vector de movimiento temporal diferente que se refiere a la segunda, imagen de referencia
40 diferente. Las imágenes de referencia que se van a verificar, como se analiza anteriormente, pueden estar en orden ascendente de índices de referencia en una lista de imágenes de referencia. Del mismo modo, si se usan diferentes vectores de movimiento temporal, los vectores de movimiento temporal se pueden seleccionar en un orden predeterminado a partir de vectores temporales de bloques vecinos al bloque actual.

45 **[0106]** Además, la unidad de compensación de movimiento 44 puede determinar si un vector de movimiento está disponible para una sub-PU en el candidato de ATMVP para una lista de imágenes de referencia particular. Si es así, se considera que el vector de movimiento está disponible para esa lista de imágenes de referencia. De otro modo, se considera que el vector de movimiento no está disponible para esa lista de imágenes de referencia. De forma
50 alternativa, si hay un vector de movimiento disponible para la otra lista de imágenes de referencia, la unidad de compensación de movimiento 44 puede modificar la información de movimiento ajustando a escala el vector de movimiento para apuntar a una imagen de referencia diana en la primera lista de imágenes de referencia, como se analiza anteriormente.

55 **[0107]** La unidad de intrapredicción 46 puede intrapredecir un bloque actual, como alternativa a la interpredicción realizada por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se analiza anteriormente. En particular, la unidad de intrapredicción 46 puede determinar un modo de intrapredicción a usar para codificar un bloque actual. En algunos ejemplos, la unidad de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante pases de codificación independientes, y la
60 unidad de intrapredicción 46 (o la unidad de selección de modo 40, en algunos ejemplos) puede seleccionar un modo de intrapredicción adecuada a usar de los modos sometidos a prueba.

65 **[0108]** Por ejemplo, la unidad de intrapredicción 46 puede calcular valores de velocidad-distorsión usando un análisis de velocidad-distorsión para los diversos modos de intrapredicción sometidos a prueba, y seleccionar el modo de intrapredicción que tenga las mejores características de velocidad-distorsión de los modos sometidos a prueba. El análisis de velocidad-distorsión determina, en general, una cantidad de distorsión (o de errores) entre un bloque codificado y un bloque original, no codificado, que se codificó para producir el bloque codificado, así como una

velocidad de bits (es decir, un número de bits) usada para producir el bloque codificado. La unidad de intrapredicción 46 puede calcular proporciones a partir de las distorsiones y velocidades para los diversos bloques codificados, para determinar qué modo de intrapredicción presenta el mejor valor de velocidad-distorsión para el bloque.

5 **[0109]** Después de seleccionar un modo de intrapredicción para un bloque, la unidad de intrapredicción 46 puede proporcionar información, indicativa del modo de intrapredicción seleccionado para el bloque, a la unidad de codificación por entropía 56. La unidad de codificación por entropía 56 puede codificar la información que indica el modo de intrapredicción seleccionado. El codificador de vídeo 20 puede incluir, en el flujo de bits transmitido, datos de configuración, que pueden incluir una pluralidad de tablas de índices de el modo de intrapredicción y una pluralidad de tablas de índices de el modo de intrapredicción modificadas (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para diversos bloques e indicaciones de un modo de intrapredicción más probable, una tabla de índices de el modo de intrapredicción y una tabla de índices del modo de intrapredicción modificada, a usar para cada uno de los contextos.

15 **[0110]** El codificador de vídeo 20 forma un bloque de vídeo residual restando los datos de predicción de la unidad de selección de modo 40 del bloque de vídeo original que se está codificando. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de procesamiento de transformada 52 aplica una transformada, tal como una transformada discreta de coseno (DCT) o una transformada conceptualmente similar, al bloque residual, produciendo un bloque de vídeo que comprende valores de coeficientes de transformada residuales. La unidad de procesamiento de transformada 52 puede realizar otras transformadas que son conceptualmente similares a la DCT. También se podrían usar transformadas de ondículas, transformadas de números enteros, transformadas de subbandas u otros tipos de transformadas.

25 **[0111]** En cualquier caso, la unidad de procesamiento de transformada 52 aplica la transformada al bloque residual, produciendo un bloque de coeficientes de transformada residuales. La transformada puede convertir la información residual desde un dominio de valores de píxel a un dominio de transformada, tal como un dominio de frecuencia. La unidad de procesamiento de transformada 52 puede enviar los coeficientes de transformada resultantes a la unidad de cuantificación 54. La unidad de cuantificación 54 cuantifica los coeficientes de transformada para reducir más la velocidad de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada con algunos de, o todos, los coeficientes. El grado de cuantificación se puede modificar ajustando un parámetro de cuantificación. En algunos ejemplos, la unidad de cuantificación 54 puede realizar, a continuación, una exploración de la matriz que incluye los coeficientes de transformada cuantificados. De forma alternativa, la unidad de codificación por entropía 56 puede realizar la exploración.

35 **[0112]** Tras la cuantificación, la unidad de codificación por entropía 56 codifica por entropía los coeficientes de transformada cuantificados. Por ejemplo, la unidad de codificación por entropía 56 puede llevar a cabo la codificación de longitud variable adaptativa al contexto (CAVLC), la codificación aritmética binaria adaptativa el contexto (CABAC), la codificación aritmética binaria adaptativa el contexto basada en sintaxis (SBAC), la codificación por entropía por división en intervalos de probabilidad (PIPE) u otra técnica de codificación por entropía. En el caso de la codificación por entropía basada en el contexto, el contexto se puede basar en bloques vecinos. Tras la codificación por entropía por parte de la unidad de codificación por entropía 56, el flujo de bits codificado se puede transmitir a otro dispositivo (por ejemplo, el descodificador de vídeo 30) o archivar para su posterior transmisión o recuperación.

45 **[0113]** La unidad de cuantificación inversa 58 y la unidad de transformada inversa 60 aplican la cuantificación inversa y la transformación inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxel, por ejemplo, para su uso posterior como bloque de referencia. La unidad 44 de compensación de movimiento puede calcular un bloque de referencia añadiendo el bloque residual a un bloque predictivo de una de las tramas de la memoria de imágenes de referencia 64. La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores fraccionarios de píxeles, para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque de predicción compensado por movimiento, producido por la unidad de compensación de movimiento 44, para producir un bloque de vídeo reconstruido para su almacenamiento en la memoria de imágenes de referencia 64. El bloque de vídeo reconstruido puede ser usado por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como bloque de referencia para intercodificar un bloque en una trama de vídeo posterior.

55 **[0114]** De esta manera, el codificador de vídeo 20 de la figura 2 representa un ejemplo de un codificador de vídeo configurado para formar, para un bloque actual de los datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales desde cuatro bloques vecinos al bloque actual y, de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de movimiento temporal avanzada (ATMVP), codificar un índice en la lista de candidatos de fusión que identifica un candidato de fusión de la pluralidad de candidatos de fusión en la lista de candidatos de fusión, y codificar el bloque actual de datos de vídeo usando información de movimiento del candidato de fusión identificado.

65 **[0115]** La figura 3 es un diagrama de bloques que ilustra un ejemplo de descodificador de vídeo 30 que puede implementar técnicas para la predicción de vector de movimiento temporal avanzada (ATMVP). En el ejemplo de la

figura 3, el descodificador de vídeo 30 incluye una unidad de descodificación por entropía 70, una unidad de compensación de movimiento 72, una unidad de intrapredicción 74, una unidad de cuantificación inversa 76, una unidad de transformación inversa 78, una memoria de imágenes de referencia 82 y un sumador 80. En algunos ejemplos, el descodificador de vídeo 30 puede realizar un pase de descodificación, en general, recíproco al pase de codificación descrito con respecto al codificador de vídeo 20 (figura 2). La unidad de compensación de movimiento 72 puede generar datos de predicción basado en vectores de movimiento recibidos desde la unidad de descodificación por entropía 70, mientras que la unidad de intrapredicción 74 puede generar datos de predicción basado en indicadores de modo de intrapredicción, recibidos desde la unidad de descodificación por entropía 70.

[0116] Durante el proceso de descodificación, el descodificador de vídeo 30 recibe un flujo de bits de vídeo codificado, que representa bloques de vídeo de un corte de vídeo codificado y elementos sintácticos asociados, desde el codificador de vídeo 20. La unidad de descodificación por entropía 70 del descodificador de vídeo 30 descodifica por entropía el flujo de bits para generar coeficientes cuantificados, vectores de movimiento o indicadores de modo de intrapredicción y otros elementos sintácticos. La unidad de descodificación por entropía 70 reenvía los vectores de movimiento y otros elementos sintácticos a la unidad de compensación de movimiento 72. El descodificador de vídeo 30 puede recibir los elementos sintácticos en el nivel de corte de vídeo y/o el nivel de bloque de vídeo.

[0117] Cuando el corte de vídeo se codifica como un corte intracodificado (I), la unidad de intrapredicción 74 puede generar datos de predicción para un bloque de vídeo del corte de vídeo actual basado en un modo de intrapredicción señalado y en datos de bloques previamente descodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un corte intercodificado (es decir, B, P o GPB), la unidad de compensación de movimiento 72 genera bloques predictivos para un bloque de vídeo del corte de vídeo actual basado en los vectores de movimiento y en otros elementos sintácticos recibidos desde la unidad de descodificación por entropía 70. Los bloques predictivos se pueden generar a partir de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El descodificador de vídeo 30 puede construir las listas de tramas de referencia, Lista 0 y Lista 1, usando técnicas de construcción predeterminadas, basado en imágenes de referencia almacenadas en la memoria de imágenes de referencia 82.

[0118] La unidad de compensación de movimiento 72 determina la información de predicción para un bloque de vídeo del corte de vídeo actual, analizando sintácticamente los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para producir los bloques predictivos para el bloque de vídeo actual que se está descodificando. Por ejemplo, la unidad de compensación de movimiento 72 usa algunos de los elementos sintácticos recibidos para determinar un modo de predicción (por ejemplo, intrapredicción o interpredicción) usado para codificar los bloques de vídeo del corte de vídeo, un tipo de corte de interpredicción (por ejemplo, un corte B, un corte P o un corte GPB), información de construcción para una o más de las listas de imágenes de referencia para el corte, vectores de movimiento para cada bloque de vídeo intercodificado del corte, el estado de interpredicción para cada bloque de vídeo intercodificado del corte y otra información para descodificar los bloques de vídeo en el corte de vídeo actual.

[0119] La unidad de compensación de movimiento 72 también puede realizar la interpolación basada en filtros de interpolación. La unidad de compensación de movimiento 72 puede usar filtros de interpolación como se usan por el codificador de vídeo 20 durante la codificación de los bloques de vídeo, para calcular valores interpolados para píxeles fraccionarios de los bloques de referencia. En este caso, la unidad de compensación de movimiento 72 puede determinar los filtros de interpolación usados por el codificador de vídeo 20 a partir de los elementos sintácticos recibidos y usar los filtros de interpolación para producir bloques predictivos.

[0120] El descodificador de vídeo 30 se puede configurar para realizar cualquiera de las diversas técnicas de esta divulgación analizadas anteriormente con respecto a la figura 1. Por ejemplo, la unidad de compensación de movimiento 72 se puede configurar para determinar si la información de movimiento para un bloque de datos de vídeo se codifica usando AMVP o modo de fusión de acuerdo con las técnicas de esta divulgación. Más en particular, la unidad de descodificación por entropía 70 puede descodificar uno o más elementos sintácticos que representan cómo se codifica la información de movimiento para el bloque actual.

[0121] Suponiendo que los elementos sintácticos indican que se realiza el modo de fusión, la unidad de compensación de movimiento 72 puede formar una lista de candidatos que incluye un conjunto de candidatos de fusión. La unidad de compensación de movimiento 72 puede añadir candidatos a la lista de candidatos basado en un orden predeterminado particular. En un ejemplo, la unidad de compensación de movimiento 72 añade los candidatos a la lista de candidatos en el orden de A1, B1, B0, A0, a continuación un candidato de predicción de vector de movimiento temporal avanzada (ATMVP). La unidad de compensación de movimiento 72 también puede añadir candidatos adicionales y realizar la depuración de la lista de candidatos, como se analiza anteriormente. Finalmente, la unidad de compensación de movimiento 72 puede descodificar un índice de fusión que representa qué candidato se usa para codificar información de movimiento para el bloque actual.

[0122] Además, en algunos ejemplos, la unidad de compensación de movimiento 72 puede determinar primero si está disponible el candidato de ATMVP. Por ejemplo, la unidad de compensación de movimiento puede determinar un bloque correspondiente al bloque actual en una imagen de referencia y determinar si está disponible la información de movimiento para el bloque correspondiente. La unidad de compensación de movimiento 72 puede a continuación

determinar que está disponible el candidato de ATMVP (es decir, el bloque correspondiente) cuando está disponible la información de movimiento para el bloque correspondiente. En algunos ejemplos, la unidad de compensación de movimiento 72 puede determinar que está disponible la información de movimiento para el bloque correspondiente cuando se predice todo el bloque correspondiente sin el uso de intrapredicción, pero no está disponible cuando al menos parte del bloque correspondiente se predice usando la intrapredicción.

[0123] De forma similar, en algunos ejemplos, la unidad de compensación de movimiento 72 puede determinar cuál de los dos candidatos de ATMVP potenciales se debe usar como el candidato de ATMVP finalmente añadido a la lista de candidatos. Por ejemplo, la unidad de compensación de movimiento 72 puede formar un primer vector de movimiento temporal con respecto al bloque actual que identifica un primer candidato de ATMVP en una primera imagen de origen de movimiento, es decir, una primera imagen de referencia. Si no está disponible la información de movimiento para el primer candidato de ATMVP, la unidad de compensación de movimiento 72 puede determinar si está disponible la información de movimiento para un segundo candidato de ATMVP diferente. El segundo candidato de ATMVP se puede identificar usando el mismo vector de movimiento temporal que se refiere a una segunda imagen de referencia diferente, un vector de movimiento temporal diferente que se refiere a la misma (es decir, la primera) imagen de referencia, o un vector de movimiento temporal diferente que se refiere a la segunda, imagen de referencia diferente. Las imágenes de referencia que se van a verificar, como se analiza anteriormente, pueden estar en orden ascendente de índices de referencia en una lista de imágenes de referencia. Del mismo modo, si se usan diferentes vectores de movimiento temporal, los vectores de movimiento temporal se pueden seleccionar en un orden predeterminado a partir de vectores temporales de bloques vecinos al bloque actual.

[0124] Además, la unidad de compensación de movimiento 72 puede determinar si un vector de movimiento está disponible para una sub-PU en el candidato de ATMVP para una lista de imágenes de referencia particular. Si es así, se considera que el vector de movimiento está disponible para esa lista de imágenes de referencia. De otro modo, se considera que el vector de movimiento no está disponible para esa lista de imágenes de referencia. De forma alternativa, si un vector de movimiento está disponible para la otra lista de imágenes de referencia, la unidad de compensación de movimiento 72 puede modificar la información de movimiento ajustando a escala el vector de movimiento para apuntar a una imagen de referencia diana en la primera lista de imágenes de referencia, como se analiza anteriormente.

[0125] La unidad de cuantificación inversa 76 cuantifica de manera inversa, es decir, descuantifica, los coeficientes de transformada cuantificados proporcionados en el flujo de bits y descodificados por entropía por la unidad de descodificación por entropía 70. El proceso de cuantificación inversa puede incluir el uso de un parámetro de cuantificación QP_{γ} , calculado por el descodificador de vídeo 30 para cada bloque de vídeo en el corte de vídeo, para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que se debería aplicar.

[0126] La unidad de transformada inversa 78 aplica una transformada inversa, por ejemplo, una DCT inversa, una transformada de enteros inversa, o un proceso de transformada inversa conceptualmente similar, a los coeficientes de transformada con el fin de generar bloques residuales en el dominio de píxel.

[0127] Después de que la unidad de compensación de movimiento 72 genere el bloque predictivo para el bloque de vídeo actual basado en los vectores de movimiento y otros elementos sintácticos, el descodificador de vídeo 30 forma un bloque de vídeo descodificado sumando los bloques residuales de la unidad de transformada inversa 78 a los bloques predictivos correspondientes generados por la unidad de compensación de movimiento 72. El sumador 80 representa el componente o los componentes que realizan esta operación de suma. Si se desea, también se puede aplicar un filtro de eliminación de bloques para filtrar los bloques descodificados, a fin de retirar distorsiones de efecto pixelado. También se pueden usar otros filtros de bucle (bien en el bucle de codificación o bien después del bucle de codificación) para allanar las transiciones de píxeles o mejorar de otro modo la calidad del vídeo. Los bloques de vídeo descodificados en una trama o imagen dada se almacenan a continuación en la memoria de imágenes de referencia 82, que almacena imágenes de referencia usadas para una compensación de movimiento posterior. La memoria de imágenes de referencia 82 almacena también vídeo descodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la figura 1.

[0128] De esta manera, el descodificador de vídeo 30 representa un ejemplo de un codificador de vídeo configurado para formar, para un bloque actual de los datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales desde cuatro bloques vecinos al bloque actual y, de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de movimiento temporal avanzada (ATMVP), codificar un índice en la lista de candidatos de fusión que identifica un candidato de fusión de la pluralidad de candidatos de fusión en la lista de candidatos de fusión, y codificar el bloque actual de datos de vídeo usando información de movimiento del candidato de fusión identificado.

[0129] La figura 4 es un diagrama conceptual que ilustra candidatos vecinos espaciales en HEVC. Los candidatos de MV espaciales se derivan de los bloques vecinos que se muestran en la figura 4, para una PU específica (PU0), aunque los procedimientos para generar los candidatos a partir de los bloques difieren para los modos de fusión y de AMVP.

- 5 **[0130]** En el modo de fusión, se pueden derivar hasta cuatro candidatos de MV espaciales con los órdenes mostrados en la figura 4(a) con números, y el orden es el siguiente: izquierda (0, A1), superior (1, B1), superior derecha (2, B0), inferior izquierda (3, A0) y arriba izquierda (4 B2), como se muestra en la figura 4(a). Es decir, en la figura 4(a), el bloque 100 incluye PUO 104A y PU1 104B. Cuando un codificador de vídeo codifica información de movimiento para PUO 104A usando el modo de fusión, el codificador de vídeo añade información de movimiento de los bloques vecinos espaciales 108A, 108B, 108C, 108D y 108E a una lista de candidatos, en ese orden. Los bloques 108A, 108B, 108C, 108D y 108E también se pueden denominar, respectivamente, bloques A1, B1, B0, A0 y B2, como en HEVC.
- 10 **[0131]** En el modo de AMVP, los bloques vecinos se dividen en dos grupos: un grupo izquierdo que incluye los bloques 0 y 1, y un grupo superior que incluye los bloques 2, 3 y 4, como se muestra en la figura 4(b). Estos bloques están marcados, respectivamente, como bloques 110A, 110B, 110C, 110D y 110E en la figura 4(b). En particular, en la figura 4(b), el bloque 102 incluye PUO 106A y PU1 106B, y los bloques 110A, 110B, 110C, 110D y 110E representan vecinos espaciales a PUO 106A. Para cada grupo, el candidato potencial en un bloque vecino, que se refiere a la misma imagen de referencia que la indicada por el índice de referencia señalado, tiene la prioridad más alta para ser elegido para formar un candidato final del grupo. Es posible que todos los bloques vecinos no contengan un vector de movimiento que apunte a la misma imagen de referencia. Por lo tanto, si no se puede encontrar dicho candidato, el primer candidato disponible se ajustará a escala para formar el candidato final; por tanto, las diferencias de distancia temporal se pueden compensar.
- 15 **[0132]** La figura 5A es un diagrama conceptual que ilustra la predicción de vector de movimiento temporal en HEVC. En particular, la figura 5(a) ilustra un ejemplo CU 120 que incluye PUO 122A y PU 1 122B. PUO 122A incluye un bloque central 126 para PU 122A y un bloque inferior derecho 124 para PUO 122A. La figura 5(a) también muestra un bloque externo 128 para el que se puede predecir información de movimiento a partir de la información de movimiento de PUO 122A, como se analiza a continuación. La figura 5(b) ilustra una imagen actual 130 que incluye un bloque actual 138 para el que se va a predecir información de movimiento. En particular, la figura 5(b) ilustra una imagen coubicada 134 a la imagen actual 130 (incluyendo el bloque coubicado 140 al bloque actual 138), una imagen de referencia actual 132 y una imagen de referencia coubicada 136. El bloque coubicado 140 se predice usando el vector de movimiento 144, que se usa como un predictor de vector de movimiento temporal (TMVP) 142 para la información de movimiento del bloque 138.
- 20 **[0133]** Un codificador de vídeo puede añadir un candidato de TMVP (por ejemplo, candidato 142 TMVP) a la lista de candidatos MV después de cualquier candidato de vector de movimiento espacial si TMVP está habilitada y el candidato de TMVP está disponible. El proceso de derivación de vectores de movimiento para un candidato de TMVP es el mismo tanto para el modo de fusión como de AMVP. Sin embargo, el índice de referencia diana para el candidato de TMVP en el modo de fusión se establece en 0, de acuerdo con HEVC.
- 25 **[0134]** La localización del bloque primario para la derivación de candidato de TMVP es el bloque inferior derecho fuera de la PU coubicada, como se muestra en la figura 5(a) como bloque 124 a PUO 122A, para compensar el sesgo a los bloques superiores e izquierdos usados para generar candidatos vecinos espaciales. Sin embargo, si el bloque 124 se encuentra fuera de la fila de CTB actual o la información de movimiento no está disponible para el bloque 124, el bloque se sustituye con el bloque central 126 de la PU como se muestra en la figura 5(a).
- 30 **[0135]** Un vector de movimiento para un candidato de TMVP 142 se deriva del bloque coubicado 140 de la imagen coubicada 134, como se indica en la información a nivel de corte.
- 35 **[0136]** De manera similar al modo directo temporal en AVC, un vector de movimiento del candidato de TMVP puede estar sujeto al ajuste a escala del vector de movimiento, que se realiza para compensar las diferencias de distancia del recuento de orden de imagen (POC) entre la imagen actual 130 y la imagen de referencia actual 132, y la imagen coubicada 134 y la imagen de referencia coubicada 136. Es decir, el vector de movimiento 144 se puede ajustar a escala para producir el candidato de TMVP 142, en base a estas diferencias de POC.
- 40 **[0137]** A continuación se analizan varios aspectos de los modos de fusión y AMVP de HEVC.
- 45 **[0138]** Ajuste a escala de vector de movimiento: Se supone que el valor del vector de movimiento es proporcional a la distancia entre las imágenes en tiempo de presentación. Un vector de movimiento asocia dos imágenes: la imagen de referencia y la imagen que contiene el vector de movimiento (a saber, la imagen contenedora). Cuando el codificador de vídeo 20 o el descodificador de vídeo 30 usan un vector de movimiento para predecir otro vector de movimiento, la distancia entre la imagen contenedora y la imagen de referencia se calcula en base a los valores de recuento de orden de imagen (POC).
- 50 **[0139]** Para que un vector de movimiento sea predicho, su imagen contenedora y la imagen de referencia asociadas son diferentes. Es decir, hay dos valores de diferencia de POC para dos vectores de movimiento distintos: un primer vector de movimiento que se va a predecir y un segundo vector de movimiento usado para predecir el primer vector de movimiento. Además, la primera diferencia de POC es la diferencia entre la imagen actual y la imagen de referencia del primer vector de movimiento, y la segunda diferencia de POC es la diferencia entre la imagen que contiene el
- 55 **[0138]** Ajuste a escala de vector de movimiento: Se supone que el valor del vector de movimiento es proporcional a la distancia entre las imágenes en tiempo de presentación. Un vector de movimiento asocia dos imágenes: la imagen de referencia y la imagen que contiene el vector de movimiento (a saber, la imagen contenedora). Cuando el codificador de vídeo 20 o el descodificador de vídeo 30 usan un vector de movimiento para predecir otro vector de movimiento, la distancia entre la imagen contenedora y la imagen de referencia se calcula en base a los valores de recuento de orden de imagen (POC).
- 60 **[0139]** Para que un vector de movimiento sea predicho, su imagen contenedora y la imagen de referencia asociadas son diferentes. Es decir, hay dos valores de diferencia de POC para dos vectores de movimiento distintos: un primer vector de movimiento que se va a predecir y un segundo vector de movimiento usado para predecir el primer vector de movimiento. Además, la primera diferencia de POC es la diferencia entre la imagen actual y la imagen de referencia del primer vector de movimiento, y la segunda diferencia de POC es la diferencia entre la imagen que contiene el
- 65 **[0139]** Para que un vector de movimiento sea predicho, su imagen contenedora y la imagen de referencia asociadas son diferentes. Es decir, hay dos valores de diferencia de POC para dos vectores de movimiento distintos: un primer vector de movimiento que se va a predecir y un segundo vector de movimiento usado para predecir el primer vector de movimiento. Además, la primera diferencia de POC es la diferencia entre la imagen actual y la imagen de referencia del primer vector de movimiento, y la segunda diferencia de POC es la diferencia entre la imagen que contiene el

- segundo vector de movimiento y la imagen de referencia a la que se refiere el segundo vector de movimiento. El segundo vector de movimiento se puede ajustar a escala en base a estas dos distancias de POC. Para un candidato vecino espacial, las imágenes contenedoras para los dos vectores de movimiento son las mismas, mientras que las imágenes de referencia son diferentes. En la HEVC, el ajuste a escala del vector de movimiento se aplica tanto a la TMVP como a la AMVP para candidatos vecinos espaciales y temporales.
- [0140]** Generación de candidato de vector de movimiento artificial: Si una lista de candidatos de vector de movimiento no está completa, se pueden generar e insertar candidatos de vector de movimiento artificiales al final de la lista hasta que la lista incluya un número predeterminado de candidatos.
- [0141]** En el modo de fusión, hay dos tipos de candidatos de MV artificiales: candidatos combinados derivados solo para cortes B y candidatos cero usados solo para la AMVP si el primer tipo no proporciona suficientes candidatos artificiales.
- [0142]** Para cada par de candidatos que ya están en la lista de candidatos y tienen la información de movimiento necesaria, los candidatos de vector de movimiento combinados bidireccionales se derivan por una combinación del vector de movimiento del primer candidato que se refiere a una imagen en la lista 0 y el vector de movimiento de un segundo candidato que se refiere a una imagen en la lista 1.
- [0143]** Proceso de depuración para la inserción del candidato: Los candidatos de diferentes bloques pueden ser los mismos, lo que disminuye la eficacia de una lista de candidatos de fusión/AMVP. Se puede aplicar un proceso de depuración para resolver este problema. De acuerdo con el proceso de depuración, un codificador de vídeo compara a un candidato con los otros en la lista de candidatos actuales para evitar la inserción de un candidato idéntico, hasta un punto determinado. Para reducir la complejidad, solo se aplica un número limitado de procesos de depuración, en lugar de comparar cada candidato potencial con todos los otros candidatos existentes que ya están en la lista.
- [0144]** La figura 6 ilustra una estructura de predicción para 3D-HEVC de ejemplo. 3D-HEVC es una extensión de vídeo 3D de HEVC en desarrollo por JCT-3V. Ciertas técnicas relacionadas con las técnicas de esta divulgación se describen con respecto a las figuras 6 y 7 a continuación.
- [0145]** La figura 6 muestra una estructura de predicción de múltiples vistas para un caso de tres vistas. V3 indica la vista base y se puede predecir una imagen en una vista no base (V1 o V5) a partir de imágenes en una vista (base) dependiente de la misma instancia de tiempo.
- [0146]** La predicción de muestra inter-vista (a partir de muestras reconstruidas) se admite en MV-HEVC, cuya estructura de predicción típica se muestra en la figura 8.
- [0147]** Tanto MV-HEVC como 3D-HEVC son compatibles con HEVC de manera que la vista (de textura) base es descodificable por el decodificador HEVC (versión 1). Un modelo de prueba para MV-HEVC y 3D-HEVC se describe en Zhang et al., "Test Model 6 of 3D-HEVC and MV-HEVC", el documento JCT-3V ISO/IEC JTC1/SC29/WG11 N13940, disponible en el sitio web mpeg.chiariglione.org/standards/mpeg-h/high-efficiency-video-coding/test-model-6-3d-hevc-and-mv-hevc a partir del 26 de enero de 2015.
- [0148]** En MV-HEVC, una imagen actual en una vista no base se puede predecir, tanto por imágenes en la misma vista como por imágenes en una vista de referencia de la misma instancia de tiempo, poniendo todas estas imágenes en listas de imágenes de referencia de la imagen. Por lo tanto, una lista de imágenes de referencia de la imagen actual, contiene tanto imágenes de referencia temporal como imágenes de referencia inter-vista.
- [0149]** Un vector de movimiento asociado con un índice de referencia correspondiente a una imagen de referencia temporal se indica como un vector de movimiento temporal.
- [0150]** Un vector de movimiento asociado con un índice de referencia correspondiente a una imagen de referencia inter-vista se indica como un vector de movimiento de disparidad.
- [0151]** 3D-HEVC admite todas las funciones de MV-HEVC. Por lo tanto, la predicción de muestra inter-vista como se mencionó anteriormente está habilitada.
- [0152]** Además, se admiten herramientas de codificación de solo textura más avanzadas y herramientas de codificación dependientes/relacionadas con la profundidad.
- [0153]** Las herramientas de codificación de solo textura a menudo requieren la identificación de los bloques correspondientes (entre vistas) que pueden pertenecer al mismo objeto. Por lo tanto, la derivación del vector de disparidad es una tecnología básica en 3D-HEVC.
- [0154]** La figura 7 es un diagrama conceptual que ilustra la predicción de movimiento de inter-vista basada en sub-PU en 3D-HEVC. La figura 7 muestra la imagen actual 160 de una vista actual (V1) y una imagen coubicada 162 en

una vista de referencia (V0). La imagen actual 160 incluye una PU actual 164 que incluye cuatro sub-Pu 166A-166D (sub-PU 166). Los vectores de disparidad 174A-174D respectivos (vectores de disparidad 174) identifican las sub-PU 168A-168D correspondientes a las sub-PU 166 en la imagen coubicada 162. En 3D-HEVC, un procedimiento de predicción de movimiento inter-vista nivel de sub-PU para el candidato de fusión inter-vista, es decir, el candidato derivado de un bloque de referencia en la vista de referencia.

[0155] Cuando dicho modo está habilitado, la PU actual 164 puede corresponder a un área de referencia (con el mismo tamaño que la PU actual identificada por el vector de disparidad) en la vista de referencia y el área de referencia puede tener información de movimiento más rica que la necesaria para la generación de un conjunto de información de movimiento típicamente para una PU. Por lo tanto, se puede usar un procedimiento de predicción de movimiento inter-vista a nivel de sub-PU (SPIVMP), como se muestra en la figura 7.

[0156] Este modo también se puede señalar como un candidato de fusión especial. Cada una de las sub-PU contiene un conjunto completo de información de movimiento. Por lo tanto, una PU puede contener múltiples conjuntos de información de movimiento.

[0157] Herencia de parámetros de movimiento (MPI) basada en sub-PU en 3D-HEVC: De forma similar, en 3D-HEVC, el candidato MPI también se puede extender de una manera similar a la predicción de movimiento inter-vista a nivel de sub-PU. Por ejemplo, si la PU de profundidad actual tiene una región coubicada que contiene múltiples PU, la PU de profundidad actual se puede separar en sub-PU, cada una de las cuales puede tener un conjunto diferente de información de movimiento. Este procedimiento se llama sub-PU MPI. Es decir, los vectores de movimiento 172A-172D de las sub-PU 168A-168D correspondientes se pueden heredar por las sub-PU 166A-166D, como vectores de movimiento 170A-170D, como se muestra en la figura 7.

[0158] Información relacionada con Sub-PU para codificación de vídeo 2D: En la solicitud de patente de los Estados Unidos N° de serie 61/883,111, se describe un diseño TMVP avanzado basado en sub-PU. En la codificación de capa única, se propone un diseño de predicción de vector de movimiento temporal avanzada de dos fases.

[0159] Una primera fase es derivar un vector que identifica el bloque correspondiente de la unidad de predicción (PU) actual en una imagen de referencia, y una segunda fase es extraer múltiples conjuntos de información de movimiento del bloque correspondiente y asignarlos a sub-PU de la PU. Cada sub-PU de la PU, por lo tanto, se compensa en movimiento por separado. El concepto de ATMVP se resume como sigue:

1. El vector de movimiento en la primera fase se puede derivar de bloques vecinos espaciales y temporales de la PU actual.

2. Este proceso se puede lograr activando un candidato de fusión entre todos los otros candidatos de fusión.

[0160] Aplicable a la codificación de capa única y a la predicción de vector de movimiento temporal de sub-PU, una PU o CU puede tener datos de refinamiento de movimiento que se van a transmitir encima de los predictores.

[0161] Varios aspectos de diseño de la solicitud 61/883,111 se destacan como sigue:

1. La primera fase de la derivación de vector también se puede simplificar con solo un vector cero.

2. La primera fase de la derivación de vector puede incluir la identificación conjunta del vector de movimiento y su imagen asociada. Se han propuesto diversas formas de seleccionar la imagen asociada y decidir además qué vector de movimiento será el vector de la primera fase.

3. Si la información de movimiento durante el proceso anterior no está disponible, se usa el "vector de la primera fase" para la sustitución.

4. Un vector de movimiento identificado a partir de un vecino temporal se tiene que ajustar a escala para ser usado para la sub-PU actual, de manera similar al ajuste a escala del vector de movimiento en TMVP. Sin embargo, se puede diseñar a qué imagen de referencia se puede ajustar a escala dicho vector de movimiento de una de las siguientes maneras:

a. La imagen se identifica mediante un índice de referencia fijo de la imagen actual.

b. La imagen se identifica como la imagen de referencia del vecino temporal correspondiente, si también está disponible en una lista de imágenes de referencia de la imagen actual.

c. La imagen se establece como la imagen coubicada identificada en la primera fase y desde donde se toman los vectores de movimiento.

[0162] La figura 8 es un diagrama conceptual que ilustra la predicción de movimiento sub-PU a partir de una imagen de referencia. En este ejemplo, la imagen actual 180 incluye una PU actual 184 (por ejemplo, una PU). En este ejemplo, el vector de movimiento 192 identifica la PU 186 de la imagen de referencia 182 en relación con la PU 184. La PU 186 se divide en sub-PU 188A-188D, cada una con sus respectivos vectores de movimiento 190A-190D. Por tanto, aunque la PU actual 184 no se divide realmente en sub-PU separadas, en este ejemplo, la PU actual 184 se puede predecir usando información de movimiento de las sub-PU 188A-188D. En particular, un codificador de vídeo puede codificar sub-PU de la PU actual 184 usando los respectivos vectores de movimiento 190A-190D. Sin embargo, el codificador de vídeo no necesita codificar elementos sintácticos que indiquen que la PU actual 184 se divide en sub-PU. De esta manera, la PU actual 184 se puede predecir eficazmente usando múltiples vectores de movimiento 190A-190D, heredados de las respectivas sub-PU 188A-188D, sin el sobrecoste de señalización de los elementos sintácticos usados para dividir la PU actual 184 en múltiples sub-PU.

[0163] La figura 9 es un diagrama conceptual que ilustra imágenes relevantes en ATMVP (similar a TMVP). En particular, la figura 9 ilustra la imagen actual 204, la imagen de origen de movimiento 206 y las imágenes de referencia 200, 202. Más en particular, la imagen actual 204 incluye el bloque actual 208. El vector de movimiento temporal 212 identifica el bloque 210 correspondiente de la imagen de origen de movimiento 206 con respecto al bloque actual 208. El bloque 210 correspondiente, a su vez, incluye el vector de movimiento 214, que se refiere a la imagen de referencia 202 y actúa como un predictor de vector de movimiento temporal avanzado para al menos una parte del bloque actual 208, por ejemplo, una sub-PU del bloque actual 208. Es decir, el vector de movimiento 214 se puede añadir como un predictor de vector de movimiento candidato para el bloque actual 208. Si se selecciona, al menos una parte del bloque actual 208 se puede predecir usando un vector de movimiento correspondiente, a saber, el vector de movimiento 216, que se refiere a la imagen de referencia 200.

[0164] La figura 10 es un diagrama de flujo que ilustra un procedimiento de ejemplo para añadir un candidato de ATMVP a una lista de candidatos durante un proceso de codificación de acuerdo con las técnicas de esta divulgación. El procedimiento de la figura 10 se describe como realizado por el codificador de vídeo 20 (figuras 1 y 2). Sin embargo, se debe entender que otros dispositivos de codificación se pueden configurar para realizar este o un procedimiento similar.

[0165] Inicialmente, el codificador de vídeo 20 obtiene un bloque de datos de vídeo que se van a codificar (no mostrado en la figura 10). El bloque puede incluir un conjunto de vecinos espaciales, tales como los mostrados en la figura 4(a) y la figura 4(b). La unidad de compensación de movimiento 44 puede construir la lista de candidatos añadiendo primero un candidato espacial izquierdo a la lista de candidatos (250). Es decir, con respecto a la figura 4(a), suponiendo que PUO 104A es el bloque que se va a codificar, el codificador de vídeo 20 puede insertar primero el vecino espacial 108A en la lista de candidatos.

[0166] A continuación, la unidad de compensación de movimiento 44 puede añadir el candidato espacial superior a la lista de candidatos (252). Con respecto a la figura 4(a), el codificador de vídeo 20 puede insertar el vecino espacial 108B en la lista de candidatos.

[0167] A continuación, la unidad de compensación de movimiento 44 puede añadir el candidato espacial superior derecho a la lista de candidatos (254). Con respecto a la figura 4(a), el codificador de vídeo 20 puede insertar el vecino espacial 108C en la lista de candidatos.

[0168] A continuación, la unidad de compensación de movimiento 44 puede añadir el candidato espacial inferior izquierdo a la lista de candidatos (256). Con respecto a la figura 4(a), el codificador de vídeo 20 puede insertar el vecino espacial 108D en la lista de candidatos.

[0169] A continuación, la unidad de compensación de movimiento 44 puede añadir un candidato de predictor de vector de movimiento temporal avanzado (ATMVP) en la lista de candidatos (258). Como se analiza anteriormente, el candidato de ATMVP puede representar un bloque correspondiente identificado por un vector temporal como se muestra y analiza con respecto a, por ejemplo, las figuras 8 y 9. Además, en algunos ejemplos, la unidad de compensación de movimiento 44 puede determinar primero si el candidato de ATMVP está disponible. Por ejemplo, la unidad de compensación de movimiento puede determinar un bloque correspondiente al bloque actual en una imagen de referencia y determinar si está disponible la información de movimiento para el bloque correspondiente. La unidad de compensación de movimiento 44 puede a continuación determinar que el candidato de ATMVP (es decir, el bloque correspondiente) está disponible cuando la información de movimiento está disponible para el bloque correspondiente. En algunos ejemplos, la unidad de compensación de movimiento 44 puede determinar que la información de movimiento está disponible para el bloque correspondiente cuando se predice todo el bloque correspondiente sin el uso de intrapredicción, pero no está disponible cuando al menos parte del bloque correspondiente se predice usando intrapredicción.

[0170] De manera similar, en algunos ejemplos, la unidad de compensación de movimiento 44 puede determinar cuál de los dos candidatos de ATMVP potenciales se debe usar como el candidato de ATMVP finalmente añadido a la lista de candidatos. Por ejemplo, la unidad de compensación de movimiento 44 puede formar un primer vector de movimiento temporal con respecto al bloque actual que identifica un primer candidato de ATMVP en una primera

imagen de origen de movimiento, es decir, una primera imagen de referencia. Si la información de movimiento no está disponible para el primer candidato de ATMVP, la unidad de compensación de movimiento 44 puede determinar si la información de movimiento está disponible para un segundo candidato de ATMVP diferente. El segundo candidato de ATMVP se puede identificar usando el mismo vector de movimiento temporal que se refiere a una segunda imagen de referencia diferente, un vector de movimiento temporal diferente que se refiere a la misma (es decir, la primera) imagen de referencia o un vector de movimiento temporal diferente que se refiere a la segunda imagen de referencia diferente. Las imágenes de referencia que se van a verificar, como se analiza anteriormente, pueden estar en orden ascendente de índices de referencia en una lista de imágenes de referencia. Del mismo modo, si se usan diferentes vectores de movimiento temporal, los vectores de movimiento temporal se pueden seleccionar en un orden predeterminado a partir de vectores temporales de bloques vecinos al bloque actual.

[0171] Además, la unidad de compensación de movimiento 44 puede determinar si un vector de movimiento está disponible para una sub-PU en el candidato de ATMVP para una lista de imágenes de referencia particular. Si es así, se considera que el vector de movimiento está disponible para esa lista de imágenes de referencia. De otro modo, se considera que el vector de movimiento no está disponible para esa lista de imágenes de referencia. De forma alternativa, si hay un vector de movimiento disponible para la otra lista de imágenes de referencia, la unidad de compensación de movimiento 44 puede modificar la información de movimiento ajustando a escala el vector de movimiento para apuntar a una imagen de referencia diana en la primera lista de imágenes de referencia, como se analiza anteriormente.

[0172] El codificador de vídeo 20 puede seleccionar a continuación uno de los candidatos de la lista de candidatos (260). Por ejemplo, el codificador de vídeo 20 puede probar la codificación del bloque usando cualquiera o todos los candidatos en la lista de candidatos. Además o de forma alternativa, la unidad de estimación de movimiento 42 del codificador de vídeo 20 puede realizar una búsqueda de movimiento y determinar un vector de movimiento para el bloque, y determinar si codifica el vector de movimiento usando la predicción de vector de movimiento avanzada (AMVP) o el modo de fusión. En el ejemplo de la figura 10, se supone que el codificador de vídeo 20 ha elegido codificar información de movimiento usando el modo de fusión. En general, el codificador de vídeo 20 (más en particular, la unidad de selección de modo 40) puede determinar cuál de los candidatos en la lista de candidatos proporciona las mejores características de velocidad-distorsión, y seleccionar ese candidato para su uso para predecir el bloque.

[0173] En consecuencia, el codificador de vídeo 20 puede predecir el bloque actual usando el candidato seleccionado (262). Es decir, la unidad de compensación de movimiento 44 puede recuperar uno o más bloques de referencia identificados por la información de movimiento del candidato seleccionado, y en algunos ejemplos puede interpolar valores para píxeles fraccionarios, si la información de movimiento tiene una precisión de subpíxeles.

[0174] A continuación, el codificador de vídeo 20 puede formar un bloque residual para el bloque actual (264). Como se analiza anteriormente, el sumador 50 puede calcular las diferencias píxel por píxel entre el bloque actual y el bloque predicho, formando el bloque residual. El codificador de vídeo 20 puede a continuación codificar información residual del bloque residual y codificar un índice de fusión (266). Es decir, la unidad de procesamiento de transformada 52 puede transformar el bloque residual, para producir coeficientes de transformada que representan la información residual. La unidad de cuantificación 54 puede cuantificar a continuación los coeficientes de transformada. La unidad de codificación por entropía 56 puede a continuación codificar por entropía los coeficientes de transformada cuantificados, así como los elementos sintácticos representativos del modo de codificación de información de movimiento (modo de fusión, en este ejemplo), y el índice de fusión que representa al candidato seleccionado de la lista de candidatos.

[0175] De esta manera, el procedimiento de la figura 10 representa un ejemplo de un procedimiento que incluye formar, para un bloque actual de datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales desde cuatro bloques vecinos al bloque actual y, de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de movimiento temporal avanzada (ATMVP), codificar un índice en la lista de candidatos de fusión que identifica un candidato de fusión de la pluralidad de candidatos de fusión en la lista de candidatos de fusión, y codificar el bloque actual de datos de vídeo usando información de movimiento del candidato de fusión identificado.

[0176] La figura 11 es un diagrama de flujo que ilustra un procedimiento de ejemplo para añadir un candidato de ATMVP a una lista de candidatos durante un proceso de decodificación de acuerdo con las técnicas de esta divulgación. El procedimiento de la figura 11 se describe como realizado por el descodificador de vídeo 30 (figuras 1 y 3). Sin embargo, se debe entender que otros dispositivos de decodificación se pueden configurar para realizar este o un procedimiento similar.

[0177] Inicialmente, el descodificador de vídeo 30 obtiene un bloque de datos de vídeo que se van a codificar (no mostrado en la figura 11). El bloque puede incluir un conjunto de vecinos espaciales, tales como los mostrados en la figura 4(a) y la figura 4(b). La unidad de compensación de movimiento 72 puede construir la lista de candidatos añadiendo primero un candidato espacial izquierdo a la lista de candidatos (270). Es decir, con respecto a la figura

4(a), suponiendo que PUO 104A es el bloque que se va a codificar, el descodificador de vídeo 30 puede insertar primero el vecino espacial 108A en la lista de candidatos.

5 **[0178]** A continuación, la unidad de compensación de movimiento 72 puede añadir el candidato espacial superior a la lista de candidatos (272). Con respecto a la figura 4(a), el descodificador de vídeo 30 puede insertar el vecino espacial 108B en la lista de candidatos.

10 **[0179]** A continuación, la unidad de compensación de movimiento 72 puede añadir el candidato espacial superior derecho a la lista de candidatos (274). Con respecto a la figura 4(a), el descodificador de vídeo 30 puede insertar el vecino espacial 108C en la lista de candidatos.

15 **[0180]** A continuación, la unidad de compensación de movimiento 72 puede añadir el candidato espacial inferior izquierdo a la lista de candidatos (276). Con respecto a la figura 4(a), el descodificador de vídeo 30 puede insertar el vecino espacial 108D en la lista de candidatos.

20 **[0181]** A continuación, la unidad de compensación de movimiento 72 puede añadir un candidato de predictor de vector de movimiento temporal avanzado (ATMVP) en la lista de candidatos (278). Como se analiza anteriormente, el candidato de ATMVP puede representar un bloque correspondiente identificado por un vector temporal como se muestra y analiza con respecto a, por ejemplo, las figuras 8 y 9. Además, en algunos ejemplos, la unidad de compensación de movimiento 72 puede determinar primero si está disponible el candidato de ATMVP. Por ejemplo, la unidad de compensación de movimiento puede determinar un bloque correspondiente al bloque actual en una imagen de referencia y determinar si está disponible la información de movimiento para el bloque correspondiente. La unidad de compensación de movimiento 72 puede a continuación determinar que está disponible el candidato de ATMVP (es decir, el bloque correspondiente) cuando está disponible la información de movimiento para el bloque correspondiente. En algunos ejemplos, la unidad de compensación de movimiento 72 puede determinar que está disponible la información de movimiento para el bloque correspondiente cuando se predice todo el bloque correspondiente sin el uso de intrapredicción, pero no está disponible cuando al menos parte del bloque correspondiente se predice usando la intrapredicción.

30 **[0182]** De forma similar, en algunos ejemplos, la unidad de compensación de movimiento 72 puede determinar cuál de los dos candidatos de ATMVP potenciales se debe usar como el candidato de ATMVP finalmente añadido a la lista de candidatos. Por ejemplo, la unidad de compensación de movimiento 72 puede formar un primer vector de movimiento temporal con respecto al bloque actual que identifica un primer candidato de ATMVP en una primera imagen de origen de movimiento, es decir, una primera imagen de referencia. Si no está disponible la información de movimiento para el primer candidato de ATMVP, la unidad de compensación de movimiento 72 puede determinar si está disponible la información de movimiento para un segundo candidato de ATMVP diferente. El segundo candidato de ATMVP se puede identificar usando el mismo vector de movimiento temporal que se refiere a una segunda imagen de referencia diferente, un vector de movimiento temporal diferente que se refiere a la misma (es decir, la primera) imagen de referencia o un vector de movimiento temporal diferente que se refiere a la segunda imagen de referencia diferente. Las imágenes de referencia que se van a verificar, como se analiza anteriormente, pueden estar en orden ascendente de índices de referencia en una lista de imágenes de referencia. Del mismo modo, si se usan diferentes vectores de movimiento temporal, los vectores de movimiento temporal se pueden seleccionar en un orden predeterminado a partir de vectores temporales de bloques vecinos al bloque actual.

45 **[0183]** Además, la unidad de compensación de movimiento 72 puede determinar si un vector de movimiento está disponible para una sub-PU en el candidato de ATMVP para una lista de imágenes de referencia particular. Si es así, se considera que el vector de movimiento está disponible para esa lista de imágenes de referencia. De otro modo, se considera que el vector de movimiento no está disponible para esa lista de imágenes de referencia. De forma alternativa, si un vector de movimiento está disponible para la otra lista de imágenes de referencia, la unidad de compensación de movimiento 72 puede modificar la información de movimiento ajustando a escala el vector de movimiento para apuntar a una imagen de referencia diana en la primera lista de imágenes de referencia, como se analiza anteriormente.

55 **[0184]** El descodificador de vídeo 30 puede a continuación descodificar un índice de fusión para seleccionar uno de los candidatos de la lista de candidatos (280). Más en particular, la unidad de descodificación por entropía 70 del descodificador de vídeo 30 puede descodificar por entropía uno o más elementos sintácticos que representan si la información de movimiento de un bloque actual está codificada usando el modo de fusión, así como un índice de fusión que representa un candidato seleccionado de la lista de candidatos.

60 **[0185]** En consecuencia, el descodificador de vídeo 30 puede predecir el bloque actual usando el candidato seleccionado (282). Es decir, la unidad de compensación de movimiento 72 puede recuperar uno o más bloques de referencia identificados por la información de movimiento del candidato seleccionado, y en algunos ejemplos puede interpolar valores para píxeles fraccionarios, si la información de movimiento tiene una precisión de subpíxeles.

65 **[0186]** El codificador de vídeo 30 también puede descodificar un bloque residual para el bloque actual (284). En particular, la unidad de descodificación por entropía 70 puede descodificar coeficientes de transformada cuantificados,

cuya unidad de cuantificación inversa 76 puede cuantificar de manera inversa para formar un bloque de transformada. La unidad de transformada inversa 78 puede a continuación transformar de manera inversa el bloque de transformada para reproducir el bloque residual. El sumador 80 puede a continuación combinar el bloque predicho con el bloque residual para descodificar el bloque actual (286), en particular, reconstruyendo el bloque actual.

[0187] De esta manera, el procedimiento de la figura 11 representa un ejemplo de un procedimiento que incluye formar, para un bloque actual de datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales desde cuatro bloques vecinos al bloque actual y, de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de movimiento temporal avanzada (ATMVP), codificar un índice en la lista de candidatos de fusión que identifica un candidato de fusión de la pluralidad de candidatos de fusión en la lista de candidatos de fusión, y codificar el bloque actual de datos de vídeo usando información de movimiento del candidato de fusión identificado.

[0188] Ha de reconocerse que, según el ejemplo, ciertos actos o sucesos de cualquiera de las técnicas descritas en el presente documento pueden realizarse en una secuencia distinta, pueden añadirse, combinarse u omitirse por completo (por ejemplo, no todos los actos o sucesos descritos son necesarios para la puesta en práctica de las técnicas). Asimismo, en determinados ejemplos, los actos o sucesos se pueden realizar simultáneamente, por ejemplo, mediante procesamiento de múltiples hilos, procesamiento de interrupciones o múltiples procesadores, en lugar de secuencialmente.

[0189] En uno o más ejemplos, las funciones descritas se pueden implementar en hardware, programas informáticos, firmware o cualquier combinación de los mismos. Si se implementan en programas informáticos, las funciones se pueden almacenar en, o transmitir por, un medio legible por ordenador, como una o más instrucciones o código, y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como medios de almacenamiento de datos, o medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático desde un lugar a otro, por ejemplo, de acuerdo a un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder, en general, a (1) medios de almacenamiento tangibles legibles por ordenador que sean no transitorios o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se pueda acceder desde uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

[0190] A modo de ejemplo, y no de limitación, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe adecuadamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde un sitio web, un servidor u otro origen remoto usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debería entenderse que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que, en cambio, están orientados a medios de almacenamiento tangibles no transitorios. El término disco, como se usa en el presente documento, incluye el disco compacto (CD), el disco láser, el disco óptico, el disco versátil digital (DVD), el disco flexible y el disco Blu-ray, donde algunos discos normalmente reproducen datos de manera magnética, mientras que otros discos reproducen datos ópticamente con láseres. Las combinaciones de lo anterior se deberían incluir también dentro del alcance de los medios legibles por ordenador.

[0191] Las instrucciones pueden ser ejecutadas por 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), arreglos de compuertas programables en el campo (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. En consecuencia, el término "procesador", como se usa en el presente documento, se puede referir a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento se puede proporcionar en módulos de hardware y/o programas informáticos dedicados configurados para la codificación y la descodificación, o incorporados en un códec combinado. Además, las técnicas se podrían implementar totalmente en uno o más circuitos o elementos lógicos.

[0192] Las técnicas de esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluyendo un equipo manual inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). Diversos componentes, módulos o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no requieren necesariamente su

realización mediante diferentes unidades de hardware. En cambio, como se ha descrito anteriormente, diversas unidades se pueden combinar en una unidad de hardware de códec o proporcionarse mediante un grupo de unidades de hardware interoperativas, incluyendo uno o más procesadores, como se ha descrito anteriormente, conjuntamente con programas informáticos y/o firmware adecuados.

- 5 **[0193]** Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un dispositivo para codificar datos de vídeo, comprendiendo el dispositivo:

5 una memoria configurada para almacenar datos de vídeo; y
un codificador de vídeo configurado para:

10 formar, para un bloque actual de los datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales de cuatro bloques vecinos al bloque actual,

caracterizado por que el codificador de vídeo está configurado además para:

15 formar la lista de candidatos de fusión incluyendo, de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de movimiento temporal avanzada (ATMVP), en el que el candidato de ATMVP indica que el bloque actual se debe predecir usando un bloque identificado por el candidato de ATMVP que se divide en una pluralidad de subbloques, teniendo cada uno de la pluralidad de subbloques conjuntos respectivos de información de movimiento;

codificar un índice en la lista de candidatos de fusión que identifique al candidato de ATMVP de la pluralidad de candidatos de fusión en la lista de candidatos de fusión; y

25 basado en el índice que identifica al candidato de ATMVP, codificar el bloque actual de datos de vídeo, en el que para codificar el bloque actual de datos de vídeo, el codificador de vídeo está configurado para codificar subbloques del bloque actual usando la información de movimiento respectiva de los subbloques del bloque identificado por el candidato de ATMVP.

30 2. El dispositivo de la reivindicación 1, en el que para formar la lista de candidatos de fusión, el codificador de vídeo está configurado para:

determinar, para el bloque actual, un bloque correspondiente en una imagen de referencia;

35 determinar si la información de movimiento está disponible para el bloque correspondiente; y

formar la lista de candidatos de fusión para incluir el candidato de ATMVP después de determinar que la información de movimiento está disponible para el bloque correspondiente.

40 3. El dispositivo de la reivindicación 2, en el que para determinar si la información de movimiento está disponible para el bloque correspondiente, el codificador de vídeo está configurado para determinar si una parte del bloque correspondiente está intrapredicha.

45 4. El dispositivo de la reivindicación 1, en el que para formar la lista de candidatos de fusión, el codificador de vídeo está configurado para formar el candidato de ATMVP a partir de un conjunto representativo de información de movimiento para un bloque correspondiente al bloque actual en una imagen de referencia.

50 5. El dispositivo de la reivindicación 4, en el que para formar el candidato de ATMVP a partir del conjunto representativo de información de movimiento, el codificador de vídeo está configurado para formar el candidato de ATMVP a partir de información de movimiento para una posición predeterminada del bloque correspondiente o para una unidad de subpredicción predeterminada (sub-PU) del bloque correspondiente.

6. El dispositivo de la reivindicación 1, en el que para formar la lista de candidatos de fusión, el codificador de vídeo está configurado para:

55 usar un primer vector de movimiento temporal, relativo al bloque actual, para identificar un primer candidato de predicción de vector de movimiento temporal avanzada (ATMVP) en una primera imagen de origen de movimiento;

60 cuando esté disponible el primer candidato de ATMVP, añadir el primer candidato de ATMVP a la lista de candidatos de fusión como el candidato de ATMVP;

cuando no esté disponible el primer candidato de ATMVP:

65 usar un segundo vector de movimiento temporal, relativo al bloque actual, para identificar un segundo candidato de ATMVP en una segunda imagen de origen de movimiento; y

añadir el segundo candidato de ATMVP a la lista de candidatos de fusión como el candidato de ATMVP.

- 5 7. El dispositivo de la reivindicación 6, en el que el primer vector de movimiento temporal y el segundo vector de movimiento temporal comprenden el mismo vector de movimiento temporal, y en el que la primera imagen de origen de movimiento y la segunda imagen de origen de movimiento comprenden diferentes imágenes de origen de movimiento.
- 10 8. El dispositivo de la reivindicación 6, en el que el primer vector de movimiento temporal y el segundo vector de movimiento temporal comprenden diferentes vectores de movimiento temporal.
- 15 9. El dispositivo de la reivindicación 6, en el que el codificador de vídeo está configurado además para seleccionar el primer vector de movimiento temporal y el segundo vector de movimiento temporal de acuerdo con un orden predeterminado a partir de vectores temporales de los bloques vecinos.
- 20 10. El dispositivo de la reivindicación 1, en el que para formar la lista de candidatos de fusión, el codificador de vídeo está configurado para:
 determinar si un vector de movimiento está disponible para un subbloque del candidato de ATMVP para una lista de imágenes de referencia X; y
 añadir el candidato de ATMVP a la lista de candidatos después de determinar que el vector de movimiento está disponible.
- 25 11. El dispositivo de la reivindicación 10, en el que cuando el vector de movimiento no está disponible para la lista de imágenes de referencia X pero está disponible para la lista de imágenes de referencia Y, en el que Y comprende una lista de imágenes de referencia distinta de la lista de imágenes de referencia X, el codificador de vídeo está configurado para fijar el vector de movimiento para estar disponible para la lista de imágenes de referencia X y ajustar a escala el vector de movimiento a una imagen de referencia en la lista de imágenes de referencia X.
- 30 12. El dispositivo de la reivindicación 1, en el que el codificador de vídeo comprende un decodificador de vídeo configurado para decodificar el índice, y para codificar el bloque actual, el decodificador de vídeo está configurado para:
 predecir el bloque actual usando la información de movimiento de los subbloques del bloque identificado por el candidato de ATMVP para formar un bloque predicho;
 decodificar información residual para el bloque actual; y
 decodificar el bloque actual usando la información residual decodificada y el bloque predicho.
- 35 40 13. El dispositivo de la reivindicación 1, en el que el codificador de vídeo comprende un codificador de vídeo configurado para codificar el índice, y en el que para codificar el bloque actual, el codificador de vídeo está configurado para:
 predecir el bloque actual usando la información de movimiento de los subbloques del bloque identificado por el candidato de ATMVP para formar un bloque predicho;
 formar un bloque residual que representa diferencias entre el bloque actual y el bloque predicho; y
 codificar la información residual.
- 45 50 14. El dispositivo de la reivindicación 1, en el que el codificador de vídeo es un codificador de vídeo, y en el que el dispositivo comprende además un transmisor para transmitir los datos de vídeo codificados.
- 55 15. Un procedimiento de codificación de datos de vídeo, comprendiendo el procedimiento:
 formar, para un bloque actual de datos de vídeo, una lista de candidatos de fusión que incluye una pluralidad de candidatos de fusión, incluyendo la pluralidad de candidatos de fusión cuatro candidatos vecinos espaciales de cuatro bloques vecinos al bloque actual y **caracterizado por que** de inmediato después de los cuatro candidatos vecinos espaciales, un candidato de predicción de vector de movimiento temporal avanzada (ATMVP), en el que el candidato de ATMVP indica que el bloque actual se debe predecir usando un bloque identificado por el candidato de ATMVP que se divide en una pluralidad de subbloques, teniendo cada uno de la pluralidad de subbloques conjuntos respectivos de información de movimiento;
- 60 65

codificar un índice en la lista de candidatos de fusión que identifica al candidato de ATMVP de la pluralidad de candidatos de fusión en la lista de candidatos de fusión; y

5 basado en el índice que identifica al candidato de ATMVP, que codifica el bloque actual de datos de vídeo, en el que la codificación del bloque actual comprende codificar subbloques del bloque actual usando la información de movimiento respectiva de los subbloques del bloque identificado por el candidato de ATMVP.

10 **16.** Un medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo que, cuando se ejecutan, provocan que un procesador realice el procedimiento de cualquiera de la reivindicación 15.

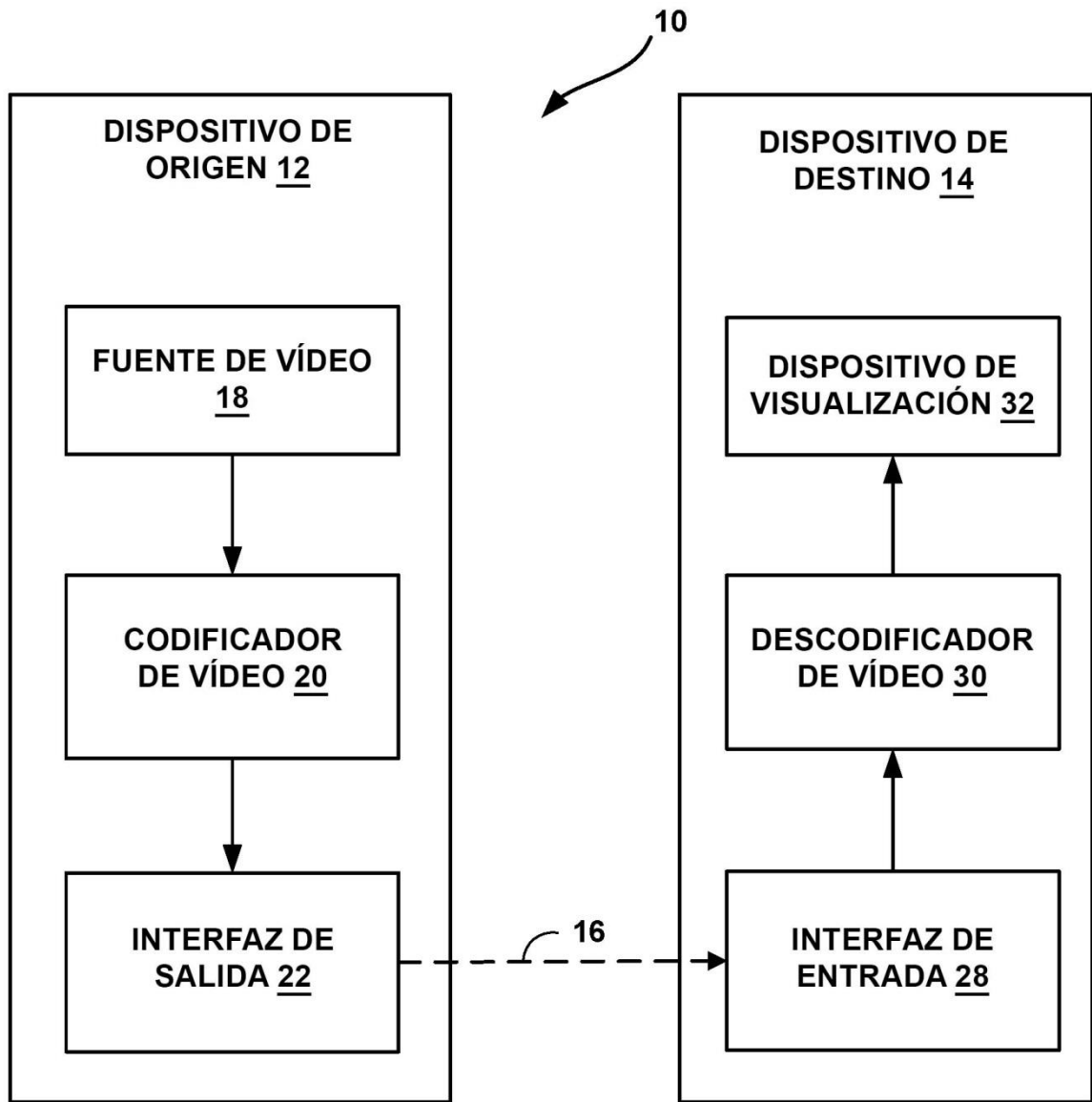


FIG. 1

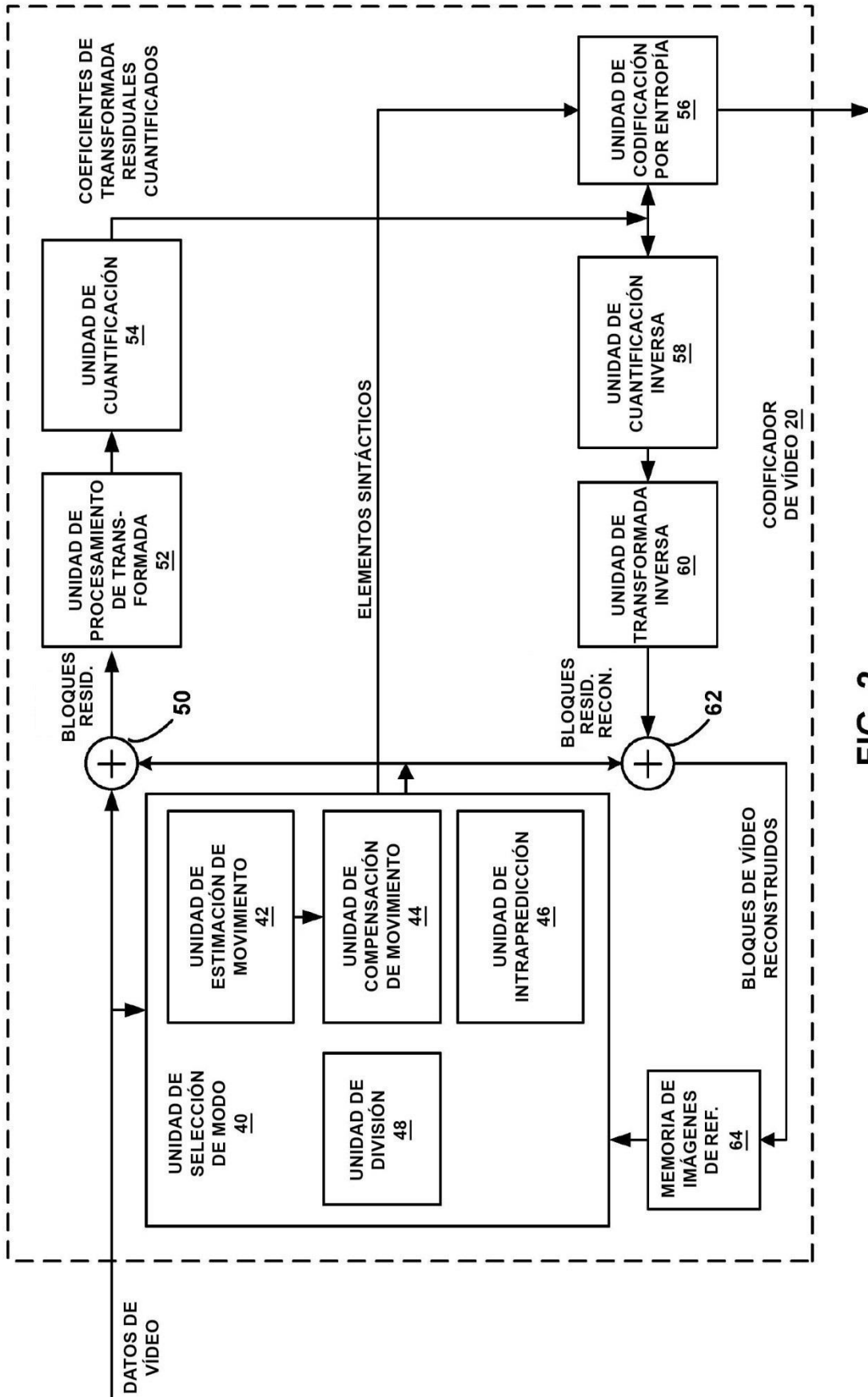


FIG. 2

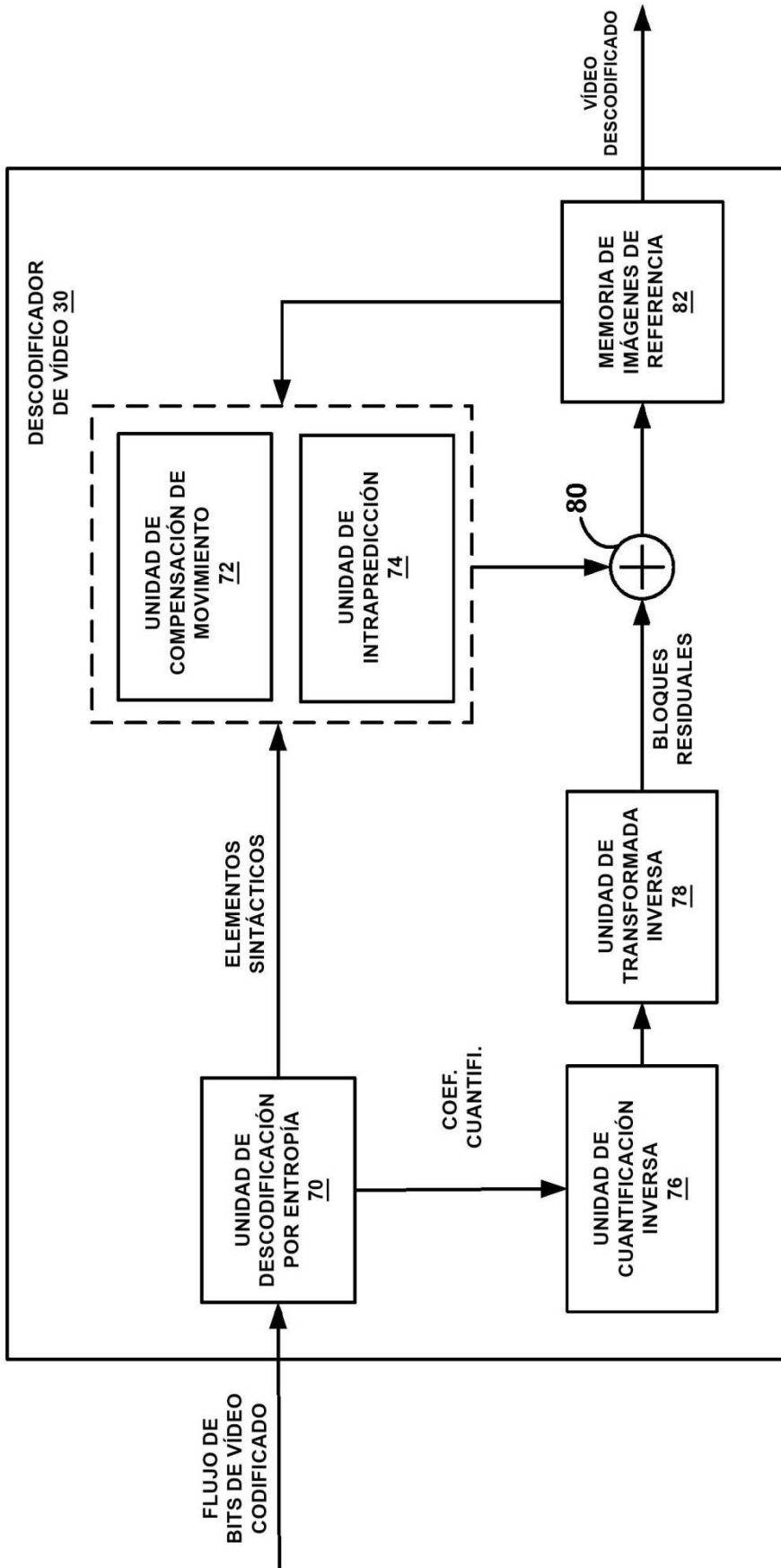


FIG. 3

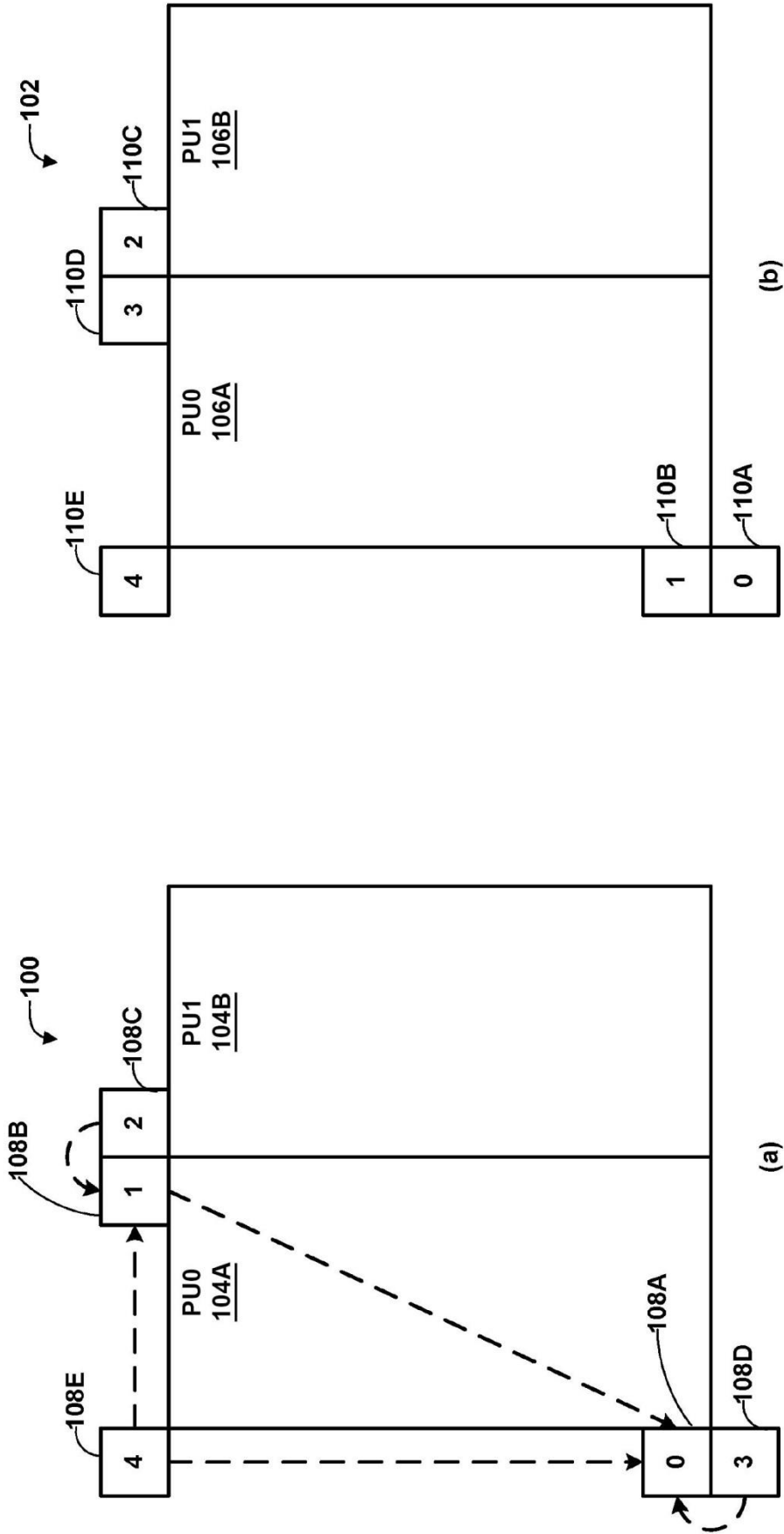


FIG. 4

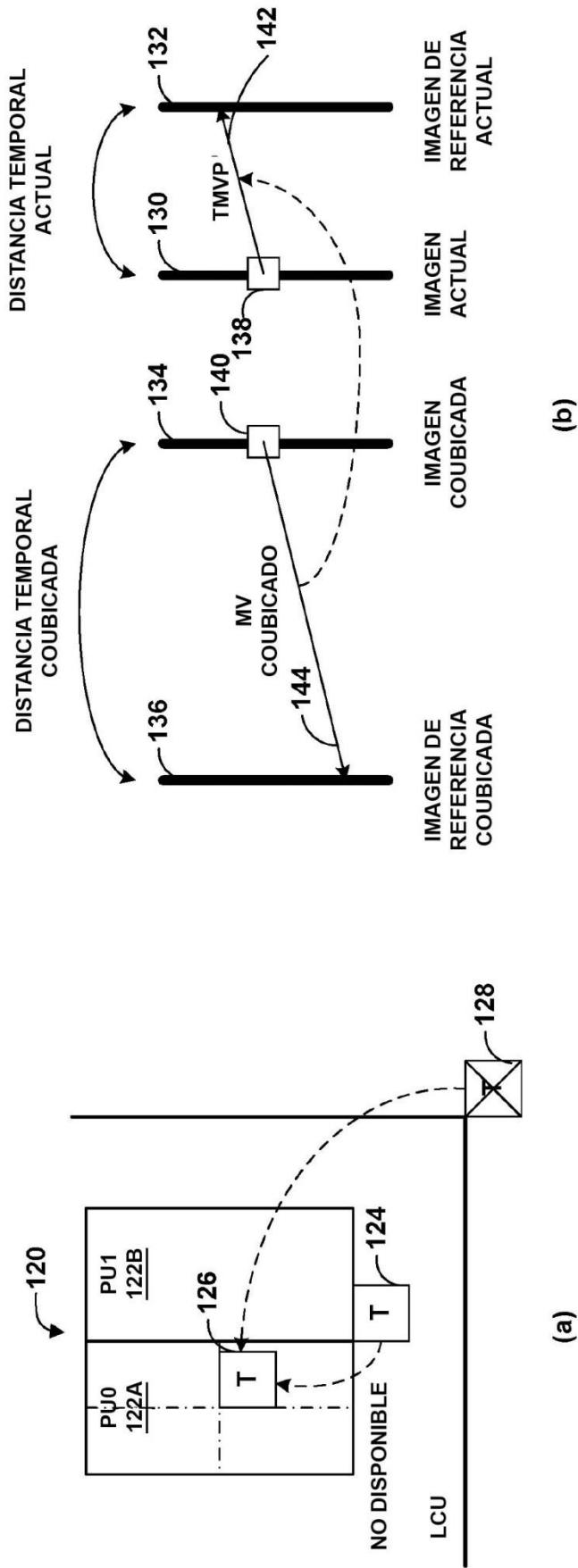


FIG. 5

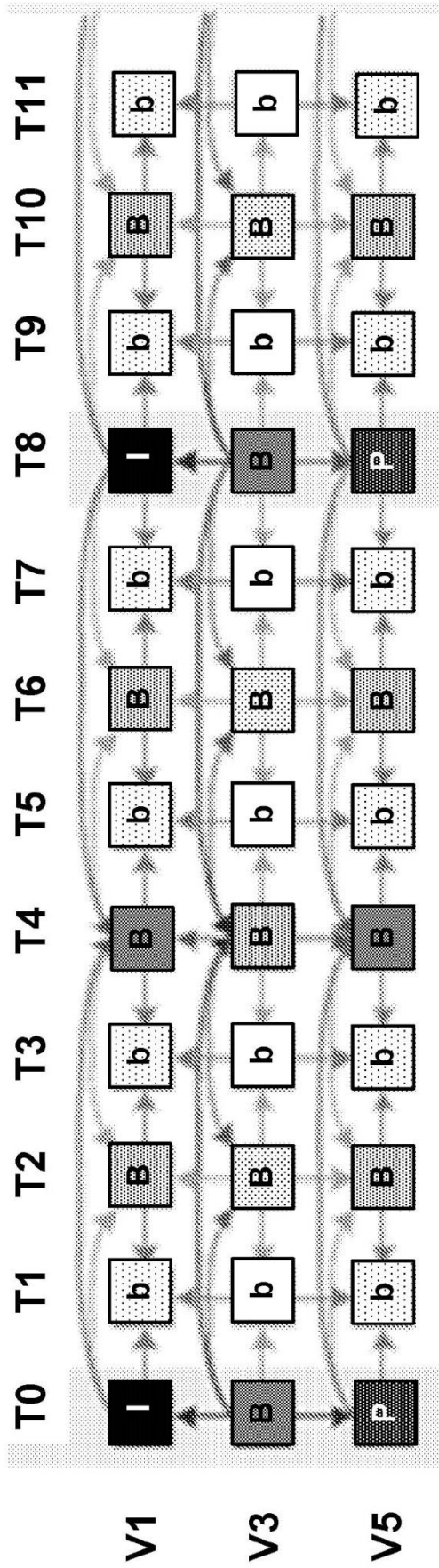
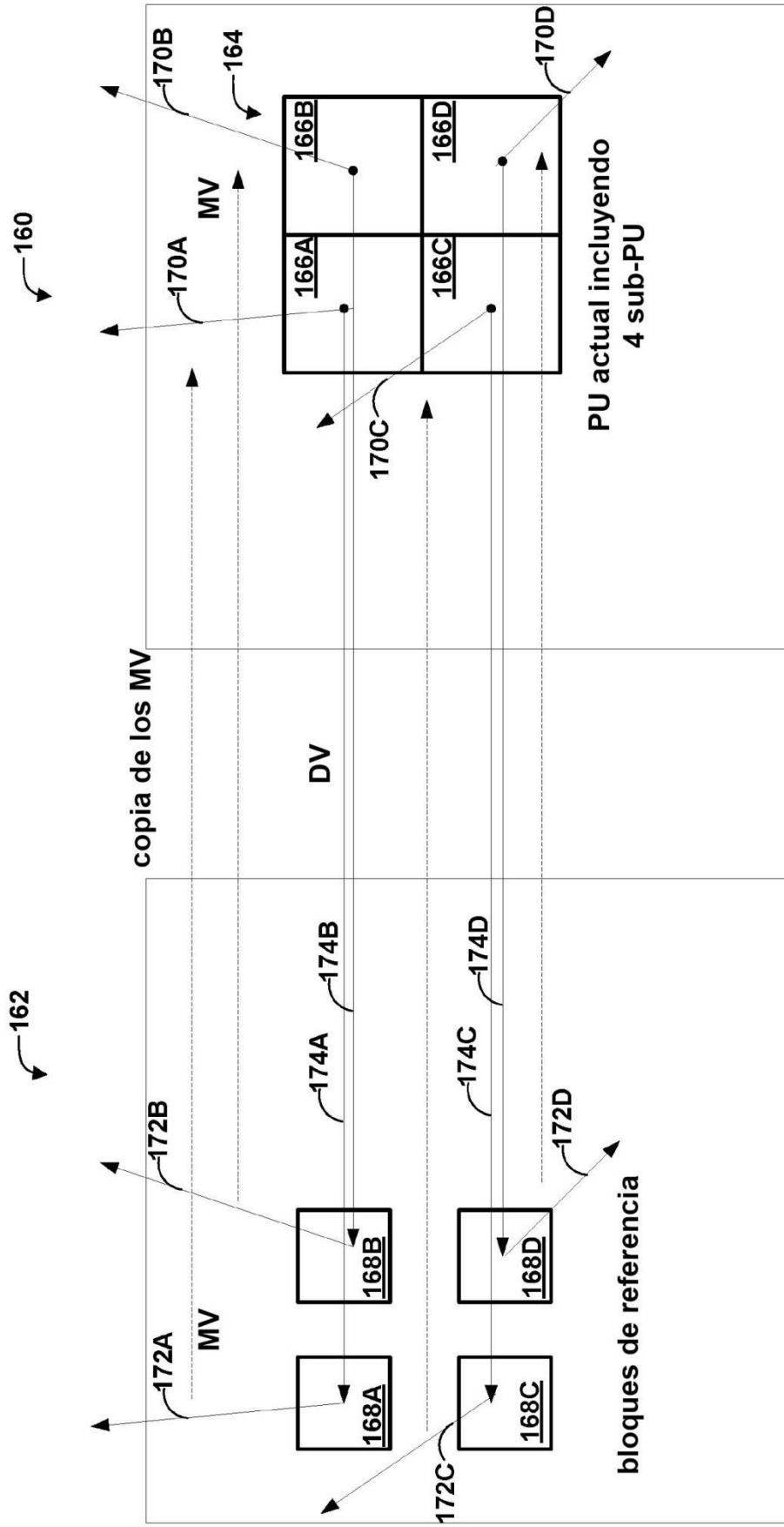


FIG. 6



V1 (vista actual)

V0 (vista de referencia)

FIG. 7

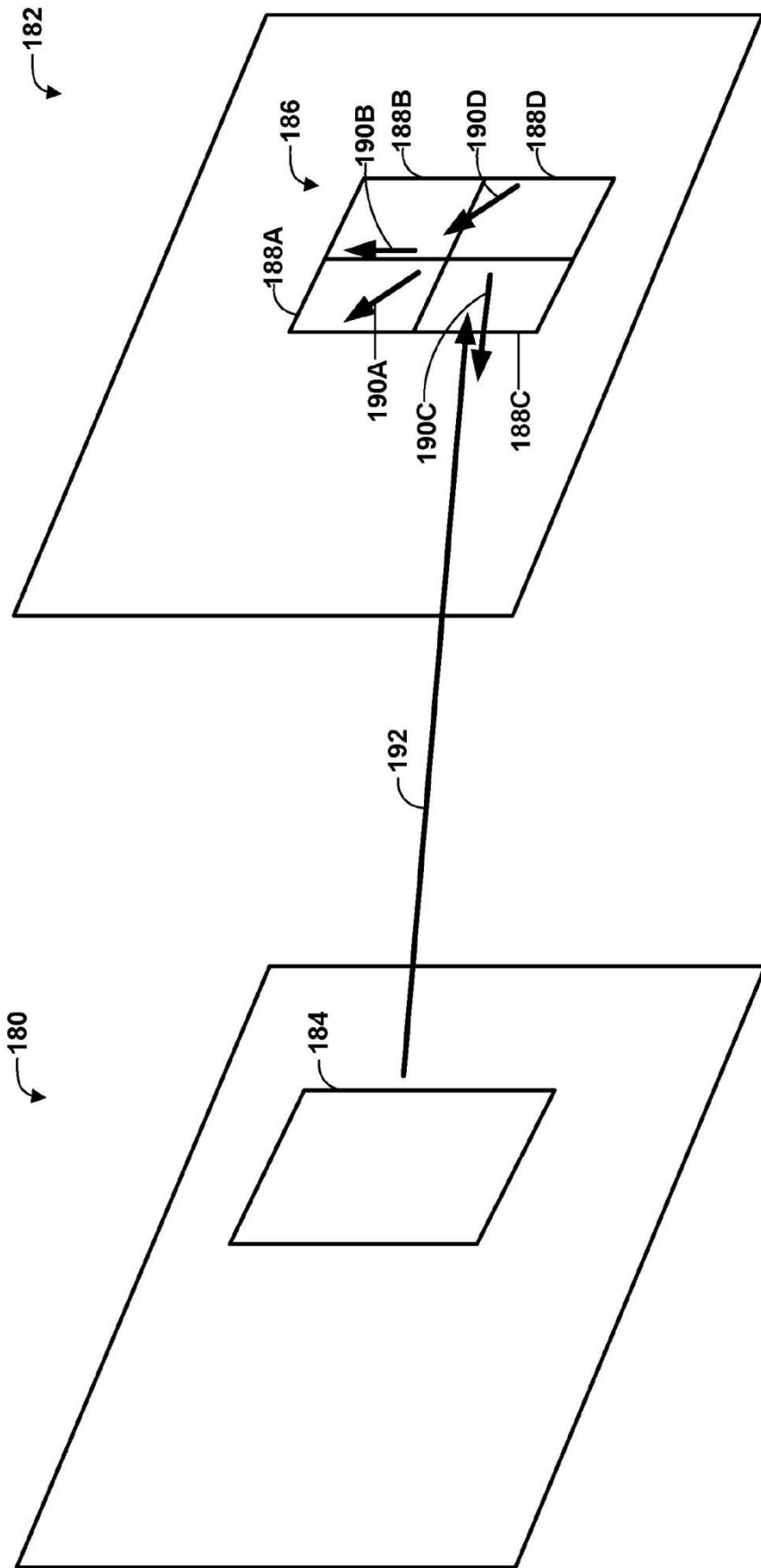


FIG. 8

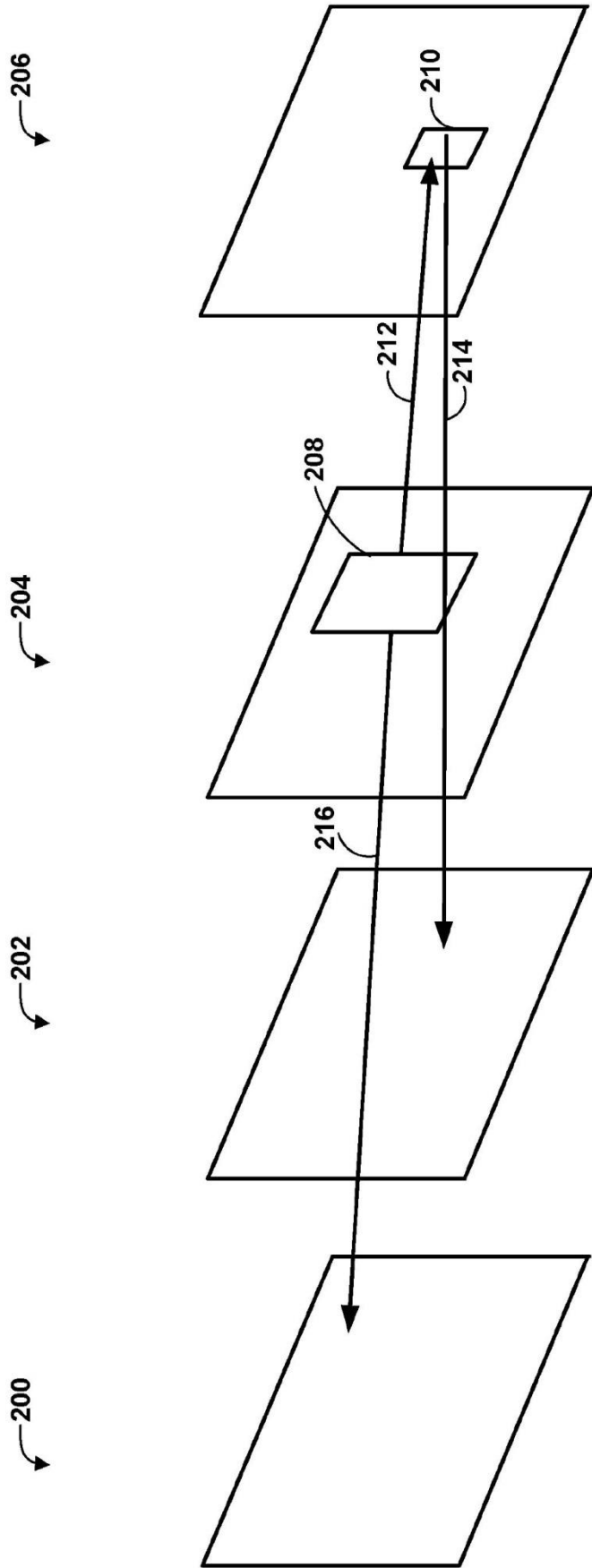


FIG. 9

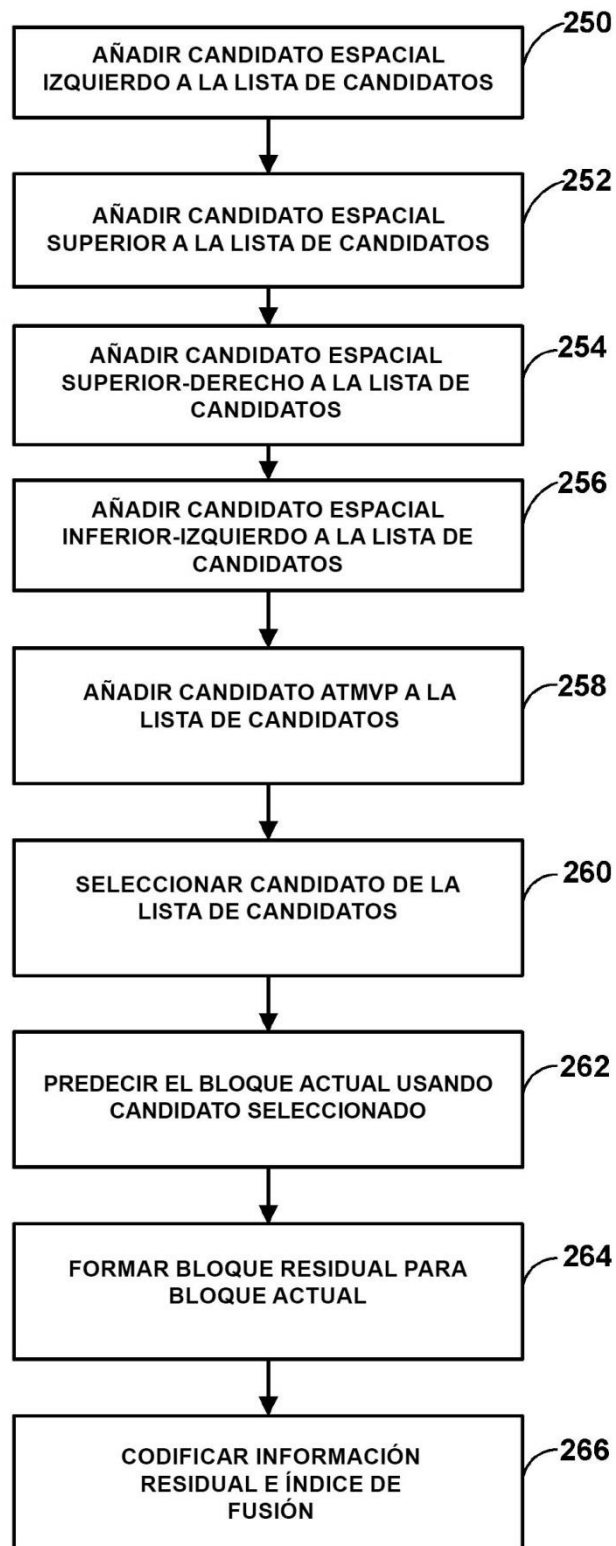


FIG. 10

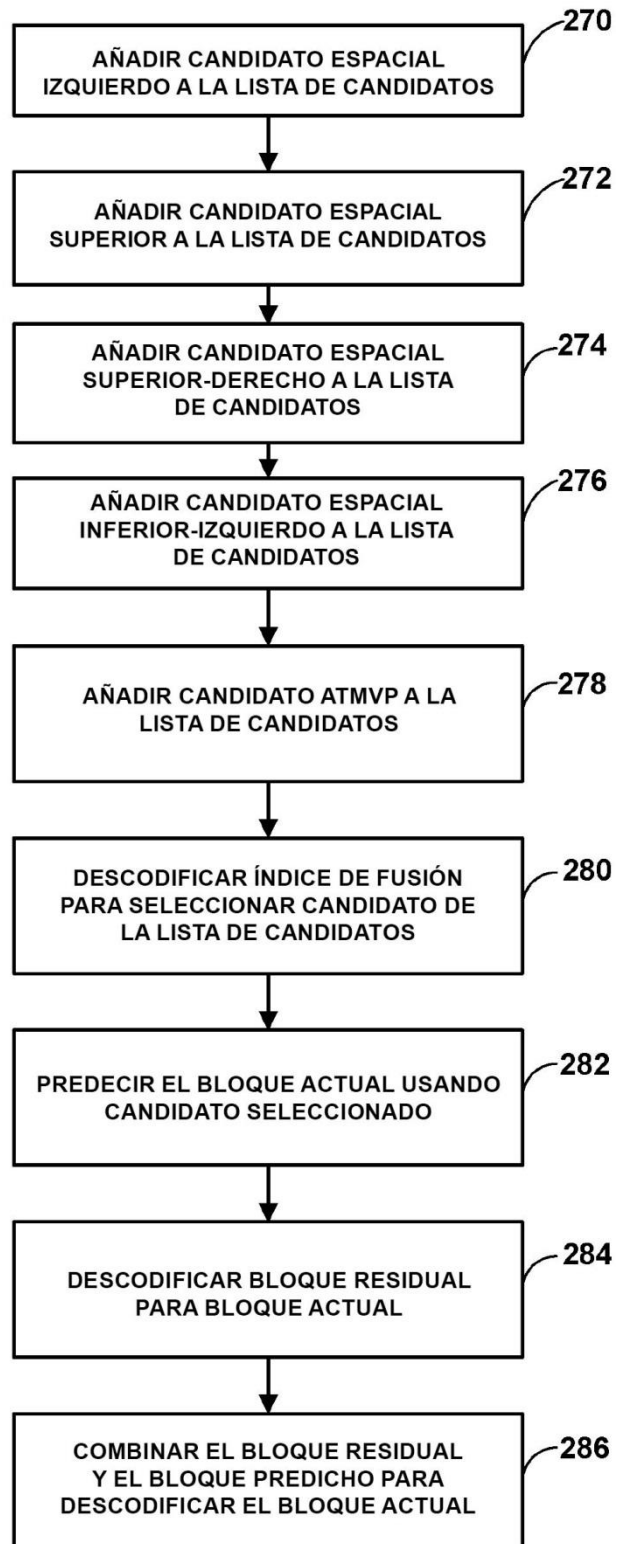


FIG. 11