

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 992 659**

51 Int. Cl.:

**H04N 19/105** (2014.01)  
**H04N 19/119** (2014.01)  
**H04N 19/174** (2014.01)  
**H04N 19/513** (2014.01)  
**H04N 19/52** (2014.01)  
**H04N 19/55** (2014.01)  
**H04N 19/577** (2014.01)  
**H04N 19/70** (2014.01)

12

## TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **20.03.2020** **PCT/CN2020/080412**  
87 Fecha y número de publicación internacional: **24.09.2020** **WO20187316**  
96 Fecha de presentación y número de la solicitud europea: **20.03.2020** **E 20774646 (2)**  
97 Fecha y número de publicación de la concesión europea: **24.04.2024** **EP 3932054**

54 Título: **Método y aparato para refinamiento de predicción con flujo óptico para un bloque codificado afín**

30 Prioridad:

**20.03.2019 US 201962821440 P**  
**28.04.2019 US 201962839765 P**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:  
**16.12.2024**

73 Titular/es:

**HUAWEI TECHNOLOGIES CO., LTD. (100.0%)**  
**Huawei Administration Building, Bantian,**  
**Longgang District**  
**Shenzhen, Guangdong 518129, CN**

72 Inventor/es:

**CHEN, HUANBANG;**  
**YANG, HAITAO y**  
**CHEN, JIANLE**

74 Agente/Representante:

**PONS ARIÑO, Ángel**

ES 2 992 659 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

Método y aparato para refinamiento de predicción con flujo óptico para un bloque codificado afín

### 5 Referencia cruzada a solicitudes relacionadas

Esta solicitud de patente reivindica la prioridad a la solicitud de patente provisional de EE. UU. N.º 62/821,440, presentada el 20 de marzo de 2019, y la prioridad a la solicitud de patente provisional de EE. UU. N.º 62/839,765, presentada el 28 de abril de 2019.

10

### Campo técnico

Realizaciones de la presente divulgación generalmente están relacionadas con el campo del procesamiento de imágenes y de manera más particular con un método para refinar la predicción compensada de movimiento afín basada en subbloque con flujo óptico cuando se requieren una o más restricciones.

15

### Antecedentes

El codificado de vídeo (codificación y decodificación de vídeo) se utiliza en un amplio abanico de aplicaciones de vídeo digital, por ejemplo, TV digital de difusión, transmisión de vídeo por la internet y redes móviles, aplicaciones de conversación en tiempo real tales como salas de vídeo, conferencias de vídeo, discos DVD y Blu-ray, sistemas de adquisición y edición de contenido de vídeo, grabadoras de vídeo de aplicaciones de seguridad.

20

La cantidad de datos de vídeo necesarios para representar incluso un vídeo relativamente corto pueden ser sustanciales, que puede tener como resultado dificultades cuando los datos se van a transmitir o comunicar de otro modo por una red de comunicaciones con capacidad limitada de ancho de banda. Así, los datos de vídeo generalmente se comprimen antes de ser comunicados por redes de telecomunicaciones diurnas por módem. El tamaño de un vídeo también podría ser una cuestión cuando el vídeo se almacena en un dispositivo de almacenamiento porque los recursos de memoria pueden ser limitados. Los dispositivos de compresión de vídeo a menudo usan software y/o hardware en la fuente para codificar los datos de vídeo antes de la transmisión o almacenamiento, disminuyendo de ese modo la cantidad de datos necesarios para representar imágenes de vídeo digital. Los datos comprimidos entonces son recibidos en el destino por un dispositivo de descompresión de vídeo que decodifica los datos de vídeo. Con recursos de red limitados y demandas cada vez mayores de calidad de vídeo superior, son deseables técnicas de compresión y descompresión mejoradas que mejoren la relación de compresión y con poco a ningún sacrificio en la calidad de imagen.

25

30

35

Recientemente, una herramienta afín se ha introducido en codificado de vídeo versátil, se pueden utilizar parámetros de modelo de movimiento teóricamente afines para derivar el vector de movimiento de cada muestra en un bloque de codificado. No obstante, debido a la alta complejidad para generar predicción compensada de movimiento afín basada en muestra, se utiliza un método de compensación de movimiento afín basada en subbloque. En este método, el bloque de codificado se divide en subbloques, cada uno de los cuales tiene asignado un vector de movimiento (MV) derivado de parámetros de modelo de movimiento afín. No obstante, pierde la precisión de predicción debido a la predicción basada en subbloque. Por lo tanto, se necesita lograr una buena compensación entre la complejidad de codificado y la precisión de predicción.

40

45

El documento de la técnica anterior WO 2019/010156 A1 divulga un dispositivo puede determinar si permitir o inhabilitar flujo óptico bidireccional (BIO) para una unidad de codificado actual (CU) (p. ej., bloque y/o subbloque).

50

### Compendio

Realizaciones de la presente solicitud proporcionan aparatos y métodos para codificación y decodificación según las reivindicaciones independientes. Realizaciones de la presente solicitud proporcionan aparatos y métodos para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afín, de manera que se puede lograr una buena compensación entre la complejidad y la precisión de predicción afín basada en subbloque.

55

La palabra "invención" como se usa en la presente descripción se refiere al concepto inventivo entendido por el solicitante en el momento de presentar la solicitud de patente. La materia para la que se busca protección se define en las reivindicaciones adjuntas. Todas las siguientes apariciones de la palabra "realización(es)", si hacen referencia a combinaciones de características diferentes a las definidas por las reivindicaciones independientes, se refieren a ejemplos que originalmente se han descrito pero que no representan realizaciones de la invención actualmente reivindicada; estos ejemplos todavía se muestran para fines ilustrativos únicamente.

60

65

Se definen realizaciones por las características de las reivindicaciones independientes, e implementaciones ventajosas adicionales de las realizaciones por las características de las reivindicaciones dependientes.

5 Realizaciones particulares se señalan en las reivindicaciones independientes adjuntas, con otras realizaciones en las reivindicaciones dependientes.

Los objetivos anteriores y otros se logran a través de la materia de asunto de las reivindicaciones independientes. Formas de implementación adicionales son evidentes a partir de las reivindicaciones dependientes, la descripción y las figuras.

10 Según un primer aspecto, la invención se refiere a un método para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afín (es decir, un bloque codificado o decodificado utilizando una herramienta afín). El método se aplica a un subbloque de muestras dentro del bloque codificado afín. El método se realiza a través del aparato de codificación o decodificación. El método incluye:

15 realizar un proceso PROF para que un subbloque actual del bloque codificado afín obtenga valores de muestra de predicción refinados (concretamente, valores de muestra de predicción finales) del subbloque actual (tal como cada subbloque) del bloque codificado afín, en donde una pluralidad de condiciones de restricción para aplicar PROF se cumplen o satisfechas para el bloque codificado afín;

20 en donde la realización de un proceso PROF para un subbloque actual del bloque codificado afín comprende: realizar un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual; y obtener un valor de muestra de predicción refinado de la muestra actual basado en el valor de predicción delta de la muestra actual y un valor de muestra de predicción de la muestra actual del subbloque actual (realizar un procesamiento de flujo óptico para el subbloque actual para obtener valores de predicción delta del subbloque actual; y

30 obtener los valores de muestra de predicción refinada del subbloque actual en función de la valores de predicción delta del subbloque actual y valores de predicción de muestra del subbloque actual). Se puede entender que cuando se generan los valores de muestra de predicción refinados de cada subbloque del bloque codificado afín, los valores de muestra de predicción refinados del bloque codificado afín se generan naturalmente.

35 Por lo tanto, se proporciona un método mejorado que permite que se logre una mejor compensación entre complejidad de codificado y precisión de predicción. El proceso de refinamiento de predicción con flujo óptico (PROF) se realiza de manera condicional para refinar la predicción compensada de movimiento afín basada en subbloque con flujo óptico en granularidad a nivel de píxel/muestra. Estas condiciones garantizan que los cálculos involucrados en el PROF solamente ocurren cuando se puede mejorar la precisión de predicción, reduciendo así el incremento innecesario de la complejidad computacional. Por lo tanto, los efectos

40 beneficiosos logrados por la tecnología divulgada en esta memoria mejoran el rendimiento de compresión general del método de codificado.

45 Se observa que el término "bloque", "bloque de codificado" o "bloque de imagen" utilizados en la presente divulgación pueden incluir unidades de transformada (TU), unidades de predicción (PU), unidades de codificado (CU), etc. En VVC, unidades de transformada y unidades de codificado en su mayoría se alinean excepto en unos pocos escenarios cuando se utiliza la formación de mosaicos de TU o la transformada de subbloque (SBT). Se puede entender que los términos "bloque", "bloque de imagen", "bloque de codificado" y "bloque de ilustración" y los términos "bloque afín", "bloque de imagen afín", "bloque codificado afín" y "bloque de movimiento afín" pueden utilizarse en esta memoria de manera intercambiable. Los términos "muestra" y

50 "píxel" pueden utilizarse de manera intercambiable entre sí en la presente divulgación. Los términos "valor de muestra de predicción" y "valores de píxel de predicción" pueden utilizarse de manera intercambiable entre sí en la presente divulgación. Los términos "ubicación de muestra" y "ubicación de píxel" pueden utilizarse de manera intercambiable entre sí en la presente divulgación en la presente divulgación.

55 El método comprende además: determinar que la pluralidad de condiciones de restricción para aplicar PROF no se cumplen para el bloque codificado afín.

60 En una forma de implementación posible del método según el primer aspecto como tal, la pluralidad de condiciones de restricción para aplicar PROF comprende: primera información de indicación indica que PROF está deshabilitado para una imagen que contiene el bloque codificado afín o primera información de indicación indica que PROF está deshabilitado para sectores asociadas con una imagen que contiene el bloque codificado afín; y segunda información de indicación indica que no hay una división del bloque codificado afín, es decir, la variable `fallbackModeTriggered` se establece a 1. Se puede entender que cuando la variable `fallbackModeTriggered` se establece a 1, no se requiere ninguna división del bloque codificado afín, es decir,

65 cada subbloque del bloque codificado afín tiene el mismo vector de movimiento. Esto indica que el bloque codificado afín solamente tiene movimiento de traslación. Cuando la variable `fallbackModeTriggered` se

establece a 0, se requiere la división del bloque codificado afín, es decir, cada subbloque del bloque codificado afín tiene un vector de movimiento respectivo. Esto indica que el bloque codificado afín no tiene movimiento de traslación.

5 En la presente divulgación, se permite que PROF no sea aplicado en algunos casos o situaciones para el bloque codificado afín. Esos casos o situaciones se determinan según las restricciones para aplicar el PROF. De esta manera, se puede lograr una mejor compensación entre complejidad de codificado y precisión de predicción.

10 En una forma de implementación posible del método según cualquier implementación precedente del primer aspecto o el primer aspecto como tal, la realización de un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual comprende:

15 obtener una segunda matriz de predicción, (en un ejemplo, la segunda matriz de predicción se genera con base en una primera matriz de predicción la cual corresponde a valores de muestra de predicción del subbloque actual). Aquí, los valores de muestra de predicción del subbloque actual pueden obtenerse realizando compensación de movimiento afín basada en subbloque para el subbloque actual, en donde un tamaño de la segunda matriz de predicción es mayor que el tamaño de la primera matriz de predicción (por ejemplo, la primera matriz de predicción tiene el tamaño de  $sbWidth * sbHeight$  y la segunda matriz de predicción tiene el tamaño de  $(sbWidth + 2) * (sbHeight + 2)$  y las variables  $sbWidth$  y  $sbHeight$  representan la anchura y la altura del subbloque actual, respectivamente); es decir, la obtención de una segunda matriz de predicción comprende: generar una primera matriz de predicción con base en información de movimiento del subbloque actual, en donde elementos de la primera matriz de predicción corresponden a valores de muestra de predicción del subbloque actual, y generar la segunda matriz de predicción con base en la primera matriz de predicción; o  
25 generar la segunda matriz de predicción con base en la información de movimiento del subbloque actual;

generar una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical con base en la segunda matriz de predicción, en donde un tamaño de la segunda matriz de predicción es mayor o igual a tamaños de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical; (por ejemplo, la matriz de gradiente de predicción horizontal o la matriz de gradiente de predicción vertical tiene el tamaño de  $sbWidth * sbHeight$  y la segunda matriz de predicción tiene el tamaño de  $(sbWidth + 2) * (sbHeight + 2)$ ); y

30 calcular un valor de predicción delta ( $\Delta(i,j)$ ) de una muestra actual del subbloque actual con base en un valor de gradiente de predicción horizontal de la muestra actual en la matriz de gradiente de predicción horizontal, un valor de gradiente de predicción vertical de la muestra actual en la matriz de gradiente de predicción vertical, y una diferencia (MVD) entre un vector de movimiento de la muestra actual del subbloque actual y un vector de movimiento de una muestra central del subbloque. Se puede entender que el MVD tiene un componente horizontal y un componente vertical. El valor de gradiente de predicción horizontal de la muestra actual en la  
40 matriz de gradiente de predicción horizontal corresponde al componente horizontal del MVD, y el valor de gradiente de predicción vertical de la muestra actual en la matriz de gradiente de predicción vertical corresponde al componente vertical del MVD.

Se observa que el bloque afín puede ser un bloque de codificado o un bloque de decodificación de una imagen de la señal de vídeo. El subbloque actual del bloque codificado afín es, por ejemplo, un bloque 4x4. La ubicación de luminancia ( $x_{Cb}$ ,  $y_{Cb}$ ) denota esa ubicación de la muestra superior-izquierda del bloque codificado afín con relación a la muestra superior-izquierda de la imagen actual. Muestras del subbloque actual se pueden referir al uso de posiciones absolutas de las muestras con respecto (o con relación) a la muestra superior-izquierda de la imagen, por ejemplo, ( $x$ ,  $y$ ), o posiciones relativas de las muestras con respecto a la muestra superior-izquierda del subbloque (en combinación con otras coordenadas), por ejemplo, ( $x_{Sb} + i$ ,  $y_{Sb} + j$ ). Aquí, ( $x_{Sb}$ ,  $y_{Sb}$ ) es la coordenada de la muestra superior izquierda del subbloque con respecto a la muestra superior izquierda de la imagen.

La primera matriz de predicción puede ser una distribución bidimensional que incluye filas y columnas y un elemento de la distribución se puede referir al uso de ( $i$ ,  $j$ ) donde  $i$  es un índice horizontal/fila y  $j$  es un índice vertical/columna. El intervalo de  $i$  y  $j$  puede ser, por ejemplo,  $i = 0 \dots sbWidth - 1$  y  $j = 0 \dots sbHeight - 1$ . Aquí,  $sbWidth$  indica la anchura del subbloque, y  $sbHeight$  indica la altura del subbloque. En algunos ejemplos, el tamaño de la primera matriz de predicción es el mismo que el tamaño del bloque actual. Por ejemplo, el tamaño de la primera matriz de predicción puede ser 4x4, y el bloque actual tiene un tamaño de 4x4.

La segunda matriz de predicción puede ser una distribución bidimensional que incluye filas y columnas y un elemento de la distribución se puede referir al uso de ( $i$ ,  $j$ ) donde  $i$  es un índice horizontal/fila y  $j$  es un índice vertical/columna. El intervalo de  $i$  y  $j$  puede ser, por ejemplo,  $i = -1 \dots sbWidth$  y  $j = -1 \dots sbHeight$ . Aquí,  $sbWidth$  indica la anchura del subbloque, y  $sbHeight$  indica la altura del subbloque. En algunos ejemplos, el tamaño de la segunda matriz de predicción es más grande que el tamaño de la primera matriz de predicción. Es decir, el tamaño de la segunda matriz de predicción puede ser más grande que el tamaño del bloque actual. Por

ejemplo, el tamaño de la segunda matriz de predicción puede ser  $(sbWidth + 2) * (sbHeight + 2)$ , mientras que el bloque actual tiene un tamaño de  $sbWidth * sbHeight$ . Por ejemplo, el tamaño de la segunda matriz de predicción puede ser  $6 \times 6$ , mientras que el bloque actual tiene un tamaño de  $4 \times 4$ .

- 5 Las matrices de gradiente de predicción horizontales y verticales pueden ser cualquier distribución bidimensional que incluye filas y columnas y un elemento de la distribución se puede referir al uso de  $(i, j)$  donde  $x$  es un índice horizontal/fila  $e$  y es un índice vertical/columna. El intervalo de  $i$  y  $j$  puede ser, por ejemplo,  $i = 0 \dots sbWidth - 1$  y  $j = 0 \dots sbHeight - 1$ .  $sbWidth$  indica la anchura del subbloque, y  $sbHeight$  indica la altura del subbloque. En algunos ejemplos, el tamaño de las matrices de gradiente de predicción horizontales y verticales es el mismo que el tamaño del bloque actual. Por ejemplo, el tamaño de las matrices de gradiente de predicción horizontales y verticales puede ser  $4 \times 4$ , y el bloque actual tiene un tamaño de  $4 \times 4$ .

- 10 Un elemento de la matriz de gradiente de predicción horizontal corresponde a un elemento de la matriz de gradiente de predicción vertical si la posición del elemento en la matriz de gradiente de predicción horizontal,  $(x, y)$  es la misma que la posición del elemento en la matriz de gradiente de predicción vertical  $(p, q)$ , es decir,  $(x, y) = (p, q)$ .

- 15 Por lo tanto, un proceso PROF tiene permitido refinar la predicción compensada de movimiento afín basada en subbloque con flujo óptico con la granularidad a nivel de muestra sin incrementar el ancho de banda de acceso de memoria (debido a que la segunda matriz de predicción se basa en la primera matriz de predicción o los valores de muestra de predicción (originales) del subbloque actual) logrando así una granularidad más fina de compensación de movimiento.

- 20 En una forma de implementación posible del método según cualquier implementación precedente del primer aspecto o el primer aspecto como tal, una diferencia de vector de movimiento entre un vector de movimiento de una unidad de muestra actual (por ejemplo, un bloque de muestra  $2 \times 2$ ) que contiene la muestra actual y un vector de movimiento de una muestra central del subbloque se utiliza como la diferencia entre el vector de movimiento de la muestra actual del subbloque actual y el vector de movimiento de la muestra central del subbloque. Aquí, el vector de movimiento de una muestra central del subbloque puede entenderse como el MV del subbloque (es decir, el MV de subbloque) al que pertenece la muestra actual  $(i, j)$ . Al utilizar la unidad de muestra, tal como el bloque de muestra  $2 \times 2$ , para calcular la diferencia de vector de movimiento, se tiene permitido equilibrar las sobrecargas de procesamiento y la precisión de predicción. En una forma de implementación posible del método según cualquier implementación precedente del primer aspecto o el primer aspecto como tal, un elemento de la segunda matriz de predicción se representa por  $I_1(p, q)$ , en donde un intervalo de valores de  $p$  es  $[-1, sbW]$ , y un intervalo de valores de  $q$  es  $[-1, sbH]$ ;

- un elemento de la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de  $i$  es  $[0, sbW - 1]$ , y un intervalo de valores de  $j$  es  $[0, sbH - 1]$ ; y

- 40 un elemento de la matriz de gradiente de predicción vertical se representa por  $Y(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de  $i$  es  $[0, sbW - 1]$ , y un intervalo de valores de  $j$  es  $[0, sbH - 1]$ ; y en donde

- 45  $sbW$  representa una anchura del subbloque actual en el bloque codificado afín,  $sbH$  representa una altura del subbloque actual en el bloque codificado afín.

- En otra manera de representación, un elemento de la segunda matriz de predicción se representa por  $I_1(p, q)$ , en donde un intervalo de valores de  $p$  es  $[0, subW + 1]$ , y un intervalo de valores de  $q$  es  $[0, subH + 1]$ ;

- 50 un elemento de la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de  $i$  es  $[1, sbW]$ , y un intervalo de valores de  $j$  es  $[1, sbH]$ ; y

- 55 un elemento de la matriz de gradiente de predicción vertical se representa por  $Y(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de  $i$  es  $[1, sbW]$ , y un intervalo de valores de  $j$  es  $[1, sbH]$ ; en donde

- 60  $sbW$  representa una anchura del subbloque actual en el bloque codificado afín,  $sbH$  representa una altura del subbloque actual en el bloque codificado afín.

- Se puede entender que la muestra superior-izquierda (o el origen de la coordenada) se ubica en  $(1, 1)$  debido a que  $p$  tiene un valor de  $[0, subW + 1]$  y  $q$  tiene un valor de  $[0, subH + 1]$ ; mientras que, la muestra superior-izquierda (o el origen de la coordenada) se ubica en  $(0, 0)$  debido a que  $p$  tiene un valor de  $[-1, subW]$  y  $q$  tiene un valor de  $[-1, subH]$ .

En una forma de implementación posible del método según cualquier implementación precedente del primer aspecto o el primer aspecto como tal, antes de la realización de un proceso PROF para un subbloque actual del bloque codificado afin, el método comprende además: realizar compensación de movimiento afin basada en subbloque para que el subbloque actual del bloque codificado afin obtenga valores de muestra de predicción (originales o a refinar) del subbloque actual.

Según un segundo aspecto de la invención, se proporciona un método para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afin, el método comprende:

realizar un proceso PROF para un subbloque actual del bloque codificado afin para obtener valores de muestra de predicción refinados (concretamente, valores de muestra de predicción finales) del subbloque actual del bloque codificado afin, en donde una pluralidad de condiciones de decisión de flujo óptico se cumplen para el bloque codificado afin; aquí, se cumple la pluralidad de condiciones de decisión de flujo óptico, lo que significa que no se cumplen todas las restricciones para aplicar PROF;

en donde la realización de un proceso PROF para un subbloque actual del bloque codificado afin comprende: realizar un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual; y obtener un valor de muestra de predicción refinado de la muestra actual con base en el valor de predicción delta de la muestra actual y un valor de muestra de predicción (original o a refinar) de la muestra actual del subbloque actual.

Por lo tanto, se proporciona un método mejorado que permite que se logre una mejor compensación entre complejidad de codificado y precisión de predicción. El proceso de refinamiento de predicción con flujo óptico (PROF) se realiza de manera condicional para refinar la predicción compensada de movimiento afin basada en subbloque con flujo óptico en granularidad a nivel de píxel/muestra. Estas condiciones garantizan que los cálculos involucrados en el PROF solamente ocurren cuando se puede mejorar la precisión de predicción, reduciendo así el incremento innecesario de la complejidad computacional. Por lo tanto, los efectos beneficiosos logrados por la tecnología divulgada en esta memoria mejoran el rendimiento de compresión general del método de codificado.

En una forma de implementación posible del método según el segundo aspecto como tal, antes de la realización de un proceso PROF para un subbloque actual del bloque codificado afin, el método comprende además: determinar que se cumple la pluralidad de condiciones de decisión de flujo óptico para el bloque codificado afin.

En una forma de implementación posible del método según el segundo aspecto como tal, la pluralidad de condiciones de decisión de flujo óptico comprende: primera información de indicación indica que PROF está habilitado para una imagen que contiene el bloque codificado afin o primera información de indicación indica que PROF está habilitado para sectores asociadas con una imagen que contiene el bloque codificado afin; y segunda información de indicación indica la división del bloque codificado afin, tal como la variable `fallbackModeTriggered` que se establece igual a 0. Se puede entender que cuando la variable `fallbackModeTriggered` se establece igual a 0, se requiere la división del bloque codificado afin, es decir, cada subbloque del bloque codificado afin tiene un vector de Movimiento respectivo, lo que indica que el bloque codificado afin tiene movimiento sin traslación.

Se permite que PROF pueda aplicarse cuando no se cumplen todas las restricciones para aplicar PROF según el diseño de las restricciones para aplicar PROF. Por lo tanto, se permite una compensación entre complejidad de codificado y precisión de predicción.

En una forma de implementación posible del método según cualquier implementación precedente del segundo aspecto o el segundo aspecto como tal, la realización de un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual comprende:

obtener una segunda matriz de predicción, en donde los elementos de la segunda matriz de predicción se basan en valores de muestra de predicción del subbloque actual; en algunos ejemplos, la obtención de una segunda matriz de predicción, comprende: generar una primera matriz de predicción con base en información de movimiento del subbloque actual, en donde elementos de la primera matriz de predicción corresponden a valores de muestra de predicción del subbloque actual, y generar la segunda matriz de predicción con base en la primera matriz de predicción; o generar la segunda matriz de predicción con base en la información de movimiento del subbloque actual;

generar una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical con base en la segunda matriz de predicción, en donde un tamaño de la segunda matriz de predicción es mayor o igual a tamaños de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical; y

5 calcular un valor de predicción delta ( $\Delta(i,j)$ ) de una muestra actual del subbloque actual con base en un valor de gradiente de predicción horizontal de la muestra actual en la matriz de gradiente de predicción horizontal, un valor de gradiente de predicción vertical de la muestra actual en la matriz de gradiente de predicción vertical, y una diferencia entre un vector de movimiento de la muestra actual del subbloque actual y un vector de movimiento de una muestra central del subbloque.

10 En una forma de implementación posible del método según cualquier implementación precedente del segundo aspecto o el segundo aspecto como tal, comprende además: realizar compensación de movimiento afin basada en subbloque para el subbloque actual del bloque codificado afin, para obtener los valores de muestra de predicción (originales) del subbloque actual del bloque codificado afin.

15 En una forma de implementación posible del método según cualquier implementación precedente del segundo aspecto o el segundo aspecto como tal, una diferencia de vector de movimiento entre un vector de movimiento de una unidad de muestra actual (por ejemplo, un bloque de muestra 2x2) al que pertenece la muestra actual y un vector de movimiento de una muestra central del subbloque se utiliza como la diferencia entre el vector de movimiento de la muestra actual del subbloque actual y el vector de movimiento de la muestra central del subbloque.

20 En una forma de implementación posible del método según cualquier implementación precedente del primer aspecto o el primer aspecto como tal,

un elemento de la segunda matriz de predicción se representa por  $I_1(p, q)$ , en donde un intervalo de valores de p es  $[-1, sbW]$ , y un intervalo de valores de q es  $[-1, sbH]$ ;

25 un elemento de la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afin, en donde un intervalo de valores de i es  $[0, sbW-1]$ , y un intervalo de valores de j es  $[0, sbH-1]$ ; y

30 un elemento de la matriz de gradiente de predicción vertical se representa por  $Y(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afin, en donde un intervalo de valores de i es  $[0, sbW-1]$ , y un intervalo de valores de j es  $[0, sbH-1]$ ; y en donde

sbW representa una anchura del subbloque actual en el bloque codificado afin, sbH representa una altura del subbloque actual en el bloque codificado afin.

35 Según un tercer aspecto, la invención se refiere a un aparato para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afin (es decir, un bloque codificado o decodificado utilizando una herramienta afin). El aparato corresponde a un aparato de codificación o decodificación. El aparato puede incluir:

40 una unidad de determinación configurada para determinar que la pluralidad de condiciones de restricción para aplicar PROF no se cumplen para el bloque codificado afin.

45 una unidad de procesamiento de predicción configurada para realizar un proceso PROF para un subbloque actual del bloque codificado afin para obtener valores de muestra de predicción refinados (concretamente, valores de muestra de predicción finales) del subbloque actual (tal como cada subbloque) del bloque codificado afin, en donde una pluralidad de condiciones de restricción para aplicar PROF no se cumplen o satisfacen para el bloque codificado afin;

50 en donde la unidad de procesamiento de predicción se configura para realizar un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual; y obtener un valor de muestra de predicción refinado de la muestra actual con base en el valor de predicción delta de la muestra actual y un valor de muestra de predicción de la muestra actual del subbloque actual (realizar un procesamiento de flujo óptico para el subbloque actual para obtener valores de predicción delta del subbloque actual; y obtener los valores de muestra de predicción refinados del subbloque actual con base en los valores de predicción delta del subbloque actual y valores de muestra de predicción del subbloque actual).

55 Se puede entender que cuando se generan los valores de muestra de predicción refinados de cada subbloque del bloque codificado afin, los valores de muestra de predicción refinados del bloque codificado afin se generan naturalmente.

60 En una forma de implementación posible del aparato según el tercer aspecto como tal, la pluralidad de condiciones de restricción para aplicar PROF comprende: primera información de indicación indica que PROF está deshabilitado para una imagen que contiene el bloque codificado afin o primera información de indicación indica que PROF está deshabilitado para sectores asociadas con una imagen que contiene el bloque codificado afin; y segunda información de indicación indica que no hay división del bloque codificado afin, es decir, la variable `fallbackModeTriggered` se establece a 1. Se puede entender que cuando la variable `fallbackModeTriggered` se establece a 1, no se requiere una división del bloque codificado afin, es decir, cada

65

subbloque del bloque codificado afin tiene el mismo vector de movimiento. Esto indica que el bloque codificado afin solamente tiene movimiento de traslación. Cuando la variable fallbackModeTriggered se establece a 0, se requiere la división del bloque codificado afin, es decir, cada subbloque del bloque codificado afin tiene un vector de movimiento respectivo. Esto indica que el bloque codificado afin no tiene movimiento de traslación.

5

En una forma de implementación posible del aparato según cualquier implementación precedente del tercer aspecto o el tercer aspecto como tal, la unidad de procesamiento de predicción se configura para obtener una segunda matriz de predicción, (en un ejemplo, la segunda matriz de predicción se genera con base en una primera matriz de predicción que corresponde a valores de muestra de predicción del subbloque actual. Aquí, los valores de muestra de predicción del subbloque actual pueden obtenerse realizando compensación de movimiento afin basada en subbloque para el subbloque actual, en donde un tamaño de la segunda matriz de predicción es mayor que el tamaño de la primera matriz de predicción (por ejemplo, la primera matriz de predicción tiene el tamaño de  $sbWidth * sbHeight$  y la segunda matriz de predicción tiene el tamaño de  $(sbWidth + 2) * (sbHeight + 2)$  y las variables  $sbWidth$  y  $sbHeight$  representan la anchura y la altura del subbloque actual, respectivamente); es decir, la obtención de una segunda matriz de predicción comprende: generar una primera matriz de predicción con base en información de movimiento del subbloque actual, en donde elementos de la primera matriz de predicción corresponden a valores de muestra de predicción del subbloque actual, y generar la segunda matriz de predicción con base en la primera matriz de predicción; o generar la segunda matriz de predicción con base en la información de movimiento del subbloque actual;

10

15

20

generar una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical con base en la segunda matriz de predicción, en donde un tamaño de la segunda matriz de predicción es mayor o igual a tamaños de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical; (por ejemplo, la matriz de gradiente de predicción horizontal o la matriz de gradiente de predicción vertical tiene el tamaño de  $sbWidth * sbHeight$  y la segunda matriz de predicción tiene el tamaño de  $(sbWidth + 2) * (sbHeight + 2)$ ); y

25

calcular un valor de predicción delta ( $\Delta(i,j)$ ) de una muestra actual del subbloque actual con base en un valor de gradiente de predicción horizontal de la muestra actual en la matriz de gradiente de predicción horizontal, un valor de gradiente de predicción vertical de la muestra actual en la matriz de gradiente de predicción vertical, y una diferencia entre un vector de movimiento de la muestra actual del subbloque actual y un vector de movimiento de una muestra central del subbloque.

30

En una forma de implementación posible del aparato según cualquier implementación precedente del tercer aspecto o el tercer aspecto como tal, una diferencia de vector de movimiento entre un vector de movimiento de una unidad de muestra actual (por ejemplo, un bloque de muestra  $2 \times 2$ ) que contiene la muestra actual y un vector de movimiento de una muestra central del subbloque se utiliza como la diferencia entre el vector de movimiento de la muestra actual del subbloque actual y el vector de movimiento de la muestra central del subbloque. Aquí, el vector de movimiento de una muestra central del subbloque puede entenderse como el MV del subbloque (es decir, el MV de subbloque) al que pertenece la muestra actual  $I(i,j)$ . Al utilizar la unidad de muestra, tal como el bloque de muestra  $2 \times 2$ , para calcular la diferencia de vector de movimiento, se tiene permitido equilibrar las sobrecargas de procesamiento y la precisión de predicción. En una forma de implementación posible del aparato según cualquier implementación precedente del primer aspecto o el tercer aspecto como tal, un elemento de la segunda matriz de predicción se representa por  $I(p, q)$ , en donde un intervalo de valores de  $p$  es  $[-1, sbW]$ , y un intervalo de valores de  $q$  es  $[-1, sbH]$ ;

35

40

45

un elemento de la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afin, en donde un intervalo de valores de  $i$  es  $[0, sbW-1]$ , y un intervalo de valores de  $j$  es  $[0, sbH-1]$ ; y

50

un elemento de la matriz de gradiente de predicción vertical se representa por  $Y(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afin, en donde un intervalo de valores de  $i$  es  $[0, sbW-1]$ , y un intervalo de valores de  $j$  es  $[0, sbH-1]$ ; y

55

$sbW$  representa una anchura del subbloque actual en el bloque codificado afin,  $sbH$  representa una altura del subbloque actual en el bloque codificado afin.

En una forma de implementación posible del aparato según cualquier implementación precedente del tercer aspecto o el tercer aspecto como tal, la unidad de procesamiento de predicción 1503 se configura para realizar compensación de movimiento afin basada en subbloque para el subbloque actual del bloque codificado afin para obtener valores de muestra de predicción (originales o a refinar) del subbloque actual.

60

Según la invención, se proporciona un aparato para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afin, el aparato comprende:

65

una unidad de determinación configurada para determinar que se cumple una pluralidad de condiciones de



decisión de flujo óptico para el bloque codificado afín; aquí, se cumple la pluralidad de condiciones de decisión de flujo óptico, lo que significa que no se cumplen todas las restricciones para aplicar PROF;

- 5 una unidad de procesamiento de predicción configurada para realizar un proceso PROF para un subbloque actual del bloque codificado afín para obtener valores de muestra de predicción refinados (concretamente, valores de muestra de predicción finales) del subbloque actual del bloque codificado afín, en donde se cumple una pluralidad de condiciones de decisión de flujo óptico para el bloque codificado afín; en donde la unidad de procesamiento de predicción se configura para realizar un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual; y obtener un valor de muestra de predicción refinado de la muestra actual con base en el valor de predicción delta de la muestra actual y un valor de muestra de predicción (original o a refinar) de la muestra actual del subbloque actual.

- 15 La pluralidad de condiciones de decisión de flujo óptico comprende: primera información de indicación indica que PROF está habilitado para una imagen que contiene el bloque codificado afín o primera información de indicación indica que PROF está habilitado para sectores asociadas con una imagen que contiene el bloque codificado afín; y segunda información de indicación indica la división del bloque codificado afín, tal como la variable `fallbackModeTriggered` que se establece igual a 0. Se puede entender que cuando la variable `fallbackModeTriggered` se establece igual a 0, se requiere la división del bloque codificado afín, es decir, cada subbloque del bloque codificado afín tiene un vector de Movimiento respectivo, lo que indica que el bloque codificado afín tiene movimiento sin traslación.

La unidad de procesamiento de predicción se configura para

- 25 obtener una segunda matriz de predicción, en donde los elementos de la segunda matriz de predicción se basan en valores de muestra de predicción del subbloque actual; en algunos ejemplos, la obtención de una segunda matriz de predicción, comprende: generar una primera matriz de predicción con base en información de movimiento del subbloque actual, en donde elementos de la primera matriz de predicción corresponden a valores de muestra de predicción del subbloque actual, y generar la segunda matriz de predicción con base en la primera matriz de predicción; o generar la segunda matriz de predicción con base en la información de movimiento del subbloque actual;

- 35 generar una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical con base en la segunda matriz de predicción, en donde un tamaño de la segunda matriz de predicción es mayor o igual a tamaños de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical; y

- 40 calcular un valor de predicción delta ( $\Delta(i,j)$ ) de una muestra actual del subbloque actual con base en un valor de gradiente de predicción horizontal de la muestra actual en la matriz de gradiente de predicción horizontal, un valor de gradiente de predicción vertical de la muestra actual en la matriz de gradiente de predicción vertical, y una diferencia entre un vector de movimiento de la muestra actual del subbloque actual y un vector de movimiento de una muestra central del subbloque.

- 45 En una forma de implementación posible del aparato según cualquier implementación precedente del cuarto aspecto o el cuarto aspecto como tal, la unidad de procesamiento de predicción se configura para realizar compensación de movimiento afín basada en subbloque para el subbloque actual del bloque codificado afín, para obtener los valores de muestra de predicción (originales) del subbloque actual del bloque codificado afín.

- 50 En una forma de implementación posible del aparato según cualquier implementación precedente del cuarto aspecto o el cuarto aspecto como tal, una diferencia de vector de movimiento entre un vector de movimiento de una unidad de muestra actual (por ejemplo, un bloque de muestra 2x2) al que pertenece la muestra actual y un vector de movimiento de una muestra central del subbloque se utiliza como la diferencia entre el vector de movimiento de la muestra actual del subbloque actual y el vector de movimiento de la muestra central del subbloque.

- 55 En una forma de implementación posible del aparato según cualquier implementación precedente del cuarto aspecto o el cuarto aspecto como tal, un elemento de la segunda matriz de predicción se representa por  $I_1(p, q)$ , en donde un intervalo de valores de  $p$  es  $[-1, sbW]$ , y un intervalo de valores de  $q$  es  $[-1, sbH]$ ;

- 60 un elemento de la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de  $i$  es  $[0, sbW-1]$ , y un intervalo de valores de  $j$  es  $[0, sbH-1]$ ; y

- 65 un elemento de la matriz de gradiente de predicción vertical se representa por  $X(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de  $i$  es  $[0, sbW-1]$ , y un intervalo de valores de  $j$  es  $[0, sbH-1]$ ; y

sbW representa una anchura del subbloque actual en el bloque codificado afín, sbH representa una altura del subbloque actual en el bloque codificado afín.

5 El método según el primer aspecto de la invención puede realizarse por el aparato según el tercer aspecto de la invención. Características y formas de implementación adicionales del aparato según el tercer aspecto de la invención corresponden a las características y formas de implementación del método según el primer aspecto de la invención.

10 El método según el segundo aspecto de la invención puede realizarse por el aparato según el cuarto aspecto de la invención. Características y formas de implementación adicionales del aparato según el cuarto aspecto de la invención corresponden a las características y formas de implementación del método según el segundo aspecto de la invención.

15 Según un quinto aspecto, la invención se refiere a un codificador (20) que comprende circuitos de procesamiento para llevar a cabo el método según los aspectos primero o segundo como tal o la forma de implementación de los mismos.

20 Según un sexto aspecto, la invención se refiere a un decodificador (30) que comprende circuitos de procesamiento para llevar a cabo el método según los aspectos primero o segundo como tal o la forma de implementación de los mismos.

Según un séptimo aspecto, la invención se refiere a un decodificador. El decodificador comprende:

25 uno o más procesadores; y

un soporte de almacenamiento legible por ordenador no transitorio acoplado a los procesadores y que almacena programación para ejecución por los procesadores, en donde la programación, cuando es ejecutada por los procesadores, configura el decodificador para llevar a cabo el método según el primer aspecto como tal o la forma de implementación del mismo.

30 Según un octavo aspecto, la invención se refiere a un codificador. El codificador comprende:

uno o más procesadores; y

35 un soporte de almacenamiento legible por ordenador no transitorio acoplado a los procesadores y que almacena programación para ejecución por los procesadores, en donde la programación, cuando es ejecutada por los procesadores, configura el codificador para llevar a cabo el método según el primer aspecto como tal o la forma de implementación del mismo

40 Según un noveno aspecto, la invención se refiere a un aparato para codificar un flujo de vídeo que incluye un procesador y una memoria. La memoria almacena instrucciones que ocasionan que el procesador ejecute el método según el segundo aspecto.

45 Según un décimo aspecto, la invención se refiere a un aparato para decodificar un flujo de vídeo que incluye un procesador y una memoria. La memoria almacena instrucciones que ocasionan que el procesador ejecute el método según el primer aspecto.

50 Según un onceavo aspecto, la invención se refiere a un programa informático que comprende un código de programa para realizar el método según el primer o segundo aspecto o cualquier realización posible del primer o segundo aspecto cuando se realiza en un ordenador.

55 Según un décimo segundo aspecto, se propone un soporte de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo que, cuando son ejecutadas, ocasionan que uno o más procesadores configurados codifiquen datos de vídeo. Las instrucciones ocasionan que uno o más procesadores ejecuten un método según el primer o segundo aspecto o cualquier realización posible del primer o segundo aspecto.

60 Según un aspecto adicional, se proporciona un método de codificación de imágenes de vídeo y comprende: determinar información de indicación, en donde la información de indicación se utiliza para indicar si un bloque de imágenes a codificar se va a codificar según un método de interpredicción objetivo, en donde el método de interpredicción objetivo comprende el método de interpredicción según el primer o segundo aspecto o cualquier realización posible del primer o segundo aspecto; y codificar la información de indicación en un flujo de bits.

65 Según un aspecto adicional, se proporciona un método de decodificación de imágenes de vídeo y comprende: analizar un flujo de bits, para obtener información de indicación, en donde la información de indicación se utiliza para indicar si un bloque de imágenes a decodificar va a ser procesado según un método de interpredicción objetivo, en donde el método de interpredicción objetivo comprende el método de interpredicción según el

primer o segundo aspecto o cualquier realización posible del primer o segundo aspecto; y procesar el bloque de imágenes a decodificar según el método de interpretación objetivo cuando la información de indicación indica que se realiza procesamiento según el método de interpretación objetivo.

- 5 Detalles de una o más realizaciones se establecen en los dibujos acompañantes y en la siguiente descripción. Otras características, objetivos y ventajas serán evidentes a partir de la descripción, los dibujos y las reivindicaciones.

Breve descripción de los dibujos

- 10 En lo sucesivo, se describen realizaciones de la invención con más detalle con referencia a las figuras y dibujos adjuntos, en los que:

- 15 la FIG. 1A es un diagrama de bloques que muestra un ejemplo de un sistema de codificado de vídeo configurado para implementar realizaciones de la invención;

la FIG. 1B es un diagrama de bloques que muestra otro ejemplo de un sistema de codificado de vídeo configurado para implementar realizaciones de la invención;

- 20 la FIG. 2 es un diagrama de bloques que muestra un ejemplo de un codificador de vídeo configurado para implementar realizaciones de la invención;

- 25 la FIG. 3 es un diagrama de bloques que muestra una estructura ejemplar de un decodificador de vídeo configurado para implementar realizaciones de la invención;

- la FIG. 4 es un diagrama de bloques que ilustra un ejemplo de un aparato de codificación o un aparato de decodificación;

- 30 la FIG. 5 es un diagrama de bloques que ilustra otro ejemplo de un aparato de codificación o un aparato de decodificación;

la FIG. 6 es un diagrama que muestra información de movimiento candidata espacial y temporal del bloque actual;

- 35 la FIG. 7 es un diagrama que muestra un bloque codificado afín actual y un bloque codificado afín vecino en el que se ubica A1;

- 40 la FIG. 8A es un diagrama que muestra un ejemplo para describir el método de predicción de vector de movimiento de punto de control construido;

- la FIG. 8B es un diagrama que muestra un ejemplo para describir el método de predicción de vector de movimiento de punto de control construido;

- 45 la FIG. 9A es un diagrama de flujo que muestra un proceso de un método de decodificación según una realización de esta solicitud;

la FIG. 9B es un diagrama que muestra un método de predicción de vectores de movimiento de punto de control construidos;

- 50 la FIG. 9C es un diagrama que muestra muestras o píxeles de un bloque codificado afín actual y que muestra vectores de movimiento del punto de control superior-izquierdo y el punto de control superior-derecho;

- 55 la FIG. 9D es un diagrama que ilustra una ventana de señal de predicción 6x6 para calcular o generar una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical y un subbloque 4x4;

la FIG. 9E es un diagrama que ilustra una ventana de señal de predicción 18x18 para calcular o generar una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical y un bloque 16x16;

- 60 la FIG. 10 es un diagrama que muestra la diferencia  $\Delta v(i,j)$  (flecha roja) entre el MV de muestra calculado para la ubicación de muestra  $(i,j)$ , denotada por  $v(i,j)$ , y el MV de subbloque ( $V_{SB}$ ) del subbloque al que pertenece la muestra  $(i,j)$ ;

- 65 la FIG. 11A es un diagrama que muestra un método para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afín según una realización de la presente divulgación;

la FIG. 11B es un diagrama que muestra otro método para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afín según otra realización de la presente divulgación;

la FIG. 12 es un diagrama que muestra un proceso PROF según una realización de la presente divulgación.

la FIG. 13 es un diagrama que muestra una región circundante y una región interior de un bloque de predicción  $(M+2)*(N+2)$  según una realización de la presente divulgación; y

la FIG. 14 es un diagrama que muestra un área circundante y un área interna de un bloque de predicción  $(M+2)*(N+2)$  según otra realización de la presente divulgación.

la FIG. 15 es un diagrama de bloques que muestra una estructura ejemplar de un aparato para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afín de una señal de vídeo según algunas aspectos de la presente divulgación;

la FIG. 16 es un diagrama de bloques que muestra una estructura ejemplar de un sistema de suministro de contenido que realiza un servicio de entrega de contenido; y

la FIG. 17 es un diagrama de bloques que muestra una estructura de un ejemplo de un dispositivo terminal.

En lo sucesivo, signos de referencia idénticos se refieren a características idénticas o al menos funcionalmente equivalentes en caso de especificarse de manera explícita de otra manera.

#### Descripción detallada de las realizaciones

En la siguiente descripción, se hace referencia a las figuras acompañantes, que forman parte de la divulgación, y que muestran, a manera de ilustración, aspectos específicos de realizaciones de la invención o aspectos específicos en los que se pueden utilizar las realizaciones de la presente invención. Se entiende que realizaciones de la invención pueden utilizarse en otros aspectos y comprenden cambios estructurales o lógicos no representados en las figuras. La siguiente descripción detallada, entonces, no se tomará en un sentido limitativo, y el alcance de la presente invención es definido por las reivindicaciones anexas.

Por ejemplo, se entiende que una divulgación en conexión con un método descrito también puede ser verdadera para un dispositivo o sistema correspondiente configurado para realizar el método y viceversa. Por ejemplo, si se describe una o una pluralidad de etapas de método específicas, un dispositivo correspondiente puede incluir una o una pluralidad de unidades, por ejemplo unidades funcionales, para realizar la etapa de método o la pluralidad de etapas de método (por ejemplo una unidad que realizar una o una pluralidad de etapas, o una pluralidad de unidades, cada una realizando una o más de la pluralidad de etapas), incluso si dicha una o más unidades no se describen o ilustran explícitamente en las figuras. Por otra parte, por ejemplo, si un aparato específico se describe en función de una o una pluralidad de unidades, por ejemplo unidades funcionales, un método correspondiente puede incluir una etapa para realizar la funcionalidad de una o una pluralidad de unidades (por ejemplo una etapa que realiza la funcionalidad de una o una pluralidad de unidades, o una pluralidad de etapas, cada una que realiza la funcionalidad de una o más de la pluralidad de unidades), incluso si dicha etapa o pluralidad de etapas no se describen o ilustran explícitamente en las figuras. Además, se entiende que las características de las diversas realizaciones y/o aspectos ejemplares descritos en esta memoria se pueden combinar entre sí, a menos que específicamente se observe de otra manera.

El codificado de vídeo típicamente se refiere al procesamiento de una secuencia de imágenes, que forman el vídeo o la secuencia de vídeo. En lugar del término "ilustración" se puede usar el término "fotograma" o "imagen" como sinónimos en el campo de codificado de vídeo. El codificado de vídeo (o codificado en general) comprende dos partes codificación de vídeo y decodificación de vídeo. La codificación de vídeo se realiza en el lado fuente, típicamente comprendiendo el procesamiento (por ejemplo por compresión) de las imágenes de vídeo originales para reducir la cantidad de datos requeridos para representar las imágenes de vídeo (para almacenamiento y/o transmisión más eficientes). La decodificación de vídeo se realiza en el lado de destino y típicamente comprende el procesamiento inverso en comparación con el codificador para reconstruir las imágenes de vídeo. Realizaciones que hacen referencia a "codificado" de imágenes de vídeo (o imágenes en general) deberán entenderse como relacionadas con la "codificación" o "decodificación" de imágenes de vídeo o secuencias de vídeo respectivas. La combinación de la parte de codificación y la parte de decodificación también se denomina CODEC (Codificación y Decodificación).

En el caso de codificado de vídeo sin pérdida, las imágenes de vídeo originales pueden ser reconstruidas, es decir las imágenes de vídeo reconstruidas tienen la misma calidad que las imágenes de vídeo originales (asumiendo que no hay pérdida de transmisión u otra pérdida de datos durante el almacenamiento o transmisión). En caso decodificado de vídeo con pérdida, se realiza compresión adicional, por ejemplo mediante cuantización, para reducir la cantidad de datos que representan las imágenes de vídeo, que no pueden ser completamente reconstruidas en el decodificador, es decir la calidad de las imágenes de vídeo

reconstruidas es inferior o peor en comparación con la calidad de las imágenes de vídeo originales.

Varias normas de codificado de vídeo pertenecen al grupo de "códecs de vídeo híbrido con pérdida" (es decir combinan predicción espacial y temporal en el dominio de muestra y codificación de transformada 2D para aplicar cuantización en el dominio de transformada). Cada imagen de una secuencia de vídeo típicamente se particiona en un conjunto de bloques que no se superponen y la codificación típicamente se realiza en un nivel de bloque. En otras palabras, en el codificador el vídeo típicamente se procesa, es decir codifica, en un nivel de bloque (bloque de vídeo), por ejemplo utilizando predicción espacial (intraimagen) y/o predicción temporal (interimagen) para generar un bloque de predicción, sustrayendo el bloque de predicción del bloque actual (bloque actualmente procesado/a procesar) para obtener un bloque residual, transformando el bloque residual y cuantificando el bloque residual en el dominio de transformada para reducir la cantidad de datos que se van a transmitir (compresión), mientras que en el decodificador el procesamiento inverso, en comparación con el codificador, se aplica al bloque codificado o se comprime para reconstruir el bloque actual para representación. Además, el codificador duplica el bucle de procesamiento de decodificador de manera que ambos generarán predicciones idénticas (por ejemplo intrapredicciones e interpredicciones) y/o reconstrucciones para procesamiento, es decir codificado, de los bloques subsiguientes.

En las siguientes realizaciones de un sistema de codificado de vídeo 10, un codificador de vídeo 20 y un decodificador de vídeo 30 se describen con base en las Figs. 1 a 3.

La Fig. 1A es un diagrama de bloques esquemático que ilustra un ejemplo de sistema de codificado 10, por ejemplo un sistema de codificado de vídeo 10 (o para abreviar sistema de codificado 10) que puede utilizar técnicas de esta presente solicitud. El codificador de vídeo 20 (o para abreviar codificador 20) y el decodificador de vídeo 30 (o para abreviar decodificador 30) del sistema de codificado de vídeo 10 representan ejemplos de dispositivos que pueden configurarse para realizar técnicas según diversos ejemplos descritos en la presente solicitud.

Tal como se muestra en la FIG. 1A, el sistema de codificado 10 comprende un dispositivo fuente 12 configurado para proporcionar datos de imagen codificados 21 por ejemplo, a un dispositivo de destino 14 para decodificar los datos de imagen codificados 21.

El dispositivo fuente 12 comprende un codificador 20, y adicionalmente, es decir opcionalmente, puede comprender una fuente de imagen 16, un preprocesador (o unidad de preprocesamiento) 18, por ejemplo un preprocesador de imagen 18, y una interfaz de comunicación o unidad de comunicación 22.

La fuente de imagen 16 puede comprender o puede ser cualquier tipo de dispositivo de captura de imagen, por ejemplo una cámara para capturar una imagen del mundo real y/o cualquier tipo de dispositivo de generación de imagen, por ejemplo un procesador de gráficos por ordenador para generar una imagen animada por ordenador, o cualquier tipo de otro dispositivo para obtener y/o proporcionar una imagen del mundo real, una imagen generada por ordenador (por ejemplo un contenido de pantalla, una imagen de realidad virtual (VR)) y/o cualquier combinación de los mismos (por ejemplo una imagen de realidad aumentada (AR)). La fuente de imagen puede ser cualquier tipo de memoria o almacenamiento que almacene cualquiera de las imágenes antes mencionadas.

A diferencia del preprocesador 18 y el procesamiento realizado por la unidad de preprocesamiento 18, la imagen o datos de imagen 17 también pueden denominarse imagen sin procesar o datos de imagen sin procesar 17.

El preprocesador 18 se configura para recibir los datos de imagen (sin procesar) 17 y para realizar preprocesamiento en los datos de imagen 17 para obtener una imagen preprocesada 19 o datos de imagen preprocesada 19. El preprocesamiento realizado por el preprocesador 18, por ejemplo, puede comprender recorte, conversión de formato de color (por ejemplo de RGB a YCbCr), corrección de color, o eliminación de ruido. Se puede entender que la unidad de preprocesamiento 18 puede ser un componente opcional.

El codificador de vídeo 20 se configura para recibir los datos de imagen preprocesada 19 y puede proporcionar datos de imagen codificada 21 (detalles adicionales se describirán a continuación, por ejemplo, con base en la Fig. 2).

La interfaz de comunicación 22 del dispositivo fuente 12 se puede configurar para recibir los datos de imagen codificada 21 y transmitir los datos de imagen codificada 21 (o cualquier versión procesada adicional de los mismos) sobre el canal de comunicación 13 a otro dispositivo, por ejemplo el dispositivo de destino 14 o cualquier otro dispositivo, para almacenamiento o reconstrucción directa.

El dispositivo de destino 14 comprende un decodificador 30 (por ejemplo un decodificador de vídeo 30), y adicionalmente, es decir opcionalmente, puede comprender una interfaz de comunicación o unidad de comunicación 28, un posprocesador 32 (o unidad de posprocesamiento 32) y un dispositivo de exhibición 34.

La interfaz de comunicación 28 del dispositivo de destino 14 se configura para recibir los datos de imagen codificada 21 (o cualquier versión procesada adicional de los mismos), por ejemplo directamente desde el dispositivo fuente 12 o desde cualquier otra fuente, por ejemplo un dispositivo de almacenamiento, por ejemplo un dispositivo de almacenamiento de datos de imagen codificada, y proporcionar los datos de imagen codificada 21 al decodificador 30.

La interfaz de comunicación 22 y la interfaz de comunicación 28 pueden configurarse para transmitir o recibir los datos de imagen codificada 21 o los datos codificados 13 a través de un enlace de comunicación directa entre el dispositivo fuente 12 y el dispositivo de destino 14, por ejemplo una conexión cableada o inalámbrica directa, o a través de cualquier tipo de red, por ejemplo una red cableada o inalámbrica o cualquier combinación de las mismas, o cualquier tipo de red privada y pública o cualquier tipo de combinación de las mismas.

La interfaz de comunicación 22 por ejemplo, puede configurarse para empaquetar los datos de imagen codificada 21 en un formato apropiado, por ejemplo paquetes, y/o procesar los datos de imagen codificada utilizando cualquier tipo de codificación de transmisión o procesamiento para transmisión sobre un enlace de comunicación o red de comunicación.

La interfaz de comunicación 28, que forma la contraparte de la interfaz de comunicación 22, por ejemplo, puede configurarse para recibir los datos transmitidos y procesar los datos de transmisión utilizando cualquier tipo de decodificación o procesamiento de transmisión correspondiente y/o desempaquetado para obtener datos de imagen codificada 21.

Ambas, la interfaz de comunicación 22 y la interfaz de comunicación 28 pueden configurarse como interfaces de comunicación unidireccionales conforme a lo indicado por la flecha para el canal de comunicación 13 en la Fig. 1A apuntando desde el dispositivo fuente 12 al dispositivo de destino 14, o interfaces de comunicación bidireccionales, y se pueden configurar, por ejemplo para enviar y recibir mensajes, por ejemplo para establecer una conexión, para reconocer e intercambiar cualquier otra información relacionada con el enlace de comunicación y/o transmisión de datos, por ejemplo transmisión de datos de imagen codificada.

El decodificador 30 se configura para recibir los datos de imagen codificados 21 y proporcionar datos de imagen decodificados 31 o una imagen decodificada 31 (detalles adicionales se describirán más adelante, p. ej. sobre la base de la Fig. 3 o la Fig. 5).

El posprocesador 32 del dispositivo de destino 14 se configura para posprocesar los datos de imagen decodificada 31 (también denominados datos de imagen reconstruida), por ejemplo la imagen decodificada 31, para obtener datos de imagen posprocesada 33, por ejemplo una imagen posprocesada 33. El posprocesamiento realizado por la unidad de posprocesamiento 32 puede comprender, p. ej. conversión de formato de color (p. ej. de YCbCr a RGB), corrección de color, recorte, o remuestreo, o cualquier otro procesamiento, p. ej. preparar los datos de imagen de decodificados 31 para exponer, p. ej. por el dispositivo de exposición 34.

El dispositivo de exhibición 34 del dispositivo de destino 14 se configura para recibir los datos de imagen posprocesada 33 para exhibir la imagen, por ejemplo a un usuario o un telespectador. El dispositivo de exhibición 34 puede ser o comprender cualquier tipo de pantalla para representar la imagen reconstruida, por ejemplo una pantalla o monitor integrado o externo. Las pantallas, por ejemplo pueden comprender pantallas de cristal líquido (LCD), pantallas de diodos de emisión de luz orgánica (OLED), pantallas de plasma, proyectores, pantallas micro LED, cristal líquido en silicio (LCoS), procesador de luz digital (DLP) o cualquier tipo de otra pantalla.

Aunque la Fig. 1A representa el dispositivo fuente 12 y el dispositivo de destino 14 como dispositivos separados, realizaciones de los dispositivos también pueden comprender ambos o ambas funcionalidades, el dispositivo fuente 12 o la funcionalidad correspondiente y el dispositivo de destino 14 o la funcionalidad correspondiente. En dichas realizaciones el dispositivo fuente 12 o la funcionalidad correspondiente y el dispositivo de destino 14 o la funcionalidad correspondiente pueden implementarse utilizando el mismo hardware y/o software o mediante hardware y/o software separados o cualquier combinación de los mismos.

Como será evidente para un experto en la técnica sobre la base de la descripción, la existencia y la partición (exacta) de funcionalidades de diferentes unidades o funcionalidades del dispositivo fuente 12 y/o el dispositivo de destino 14 como se muestran en la Fig. 1A pueden variar dependiendo de un dispositivo real y la aplicación.

El codificador 20 (por ejemplo un codificador de vídeo 20) o el decodificador 30 (por ejemplo un decodificador de vídeo 30) o ambos del codificador 20 y el decodificador 30 pueden implementarse a través de circuitos de procesamiento como se utilizan en la Fig. 1B, tal como uno o más microprocesadores, procesadores de señal digital (DSP), circuitos integrados de aplicación específica (ASIC), matrices de puertas programables en campo (FPGA), lógica discreta, hardware, codificado de vídeo dedicado o cualquier combinación de los mismos. El

codificador 20 puede implementarse a través de un circuito de procesamiento 46 para incorporar los diversos módulos como se analiza con respecto al codificador 20 de la FIG. 2 y/o cualquier otro sistema o subsistema de codificador descrito en esta memoria. El decodificador 30 puede implementarse a través del circuito de procesamiento 46 para incorporar los diversos módulos como se analiza con respecto al decodificador 30 de la FIG. 3 y/o cualquier otro sistema o subsistema de decodificador descrito en esta memoria. El circuito de procesamiento puede configurarse para realizar las diversas operaciones como se analiza más adelante. Como se muestra en la fig. 5, si las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un soporte de almacenamiento legible por ordenador no-transitorio conveniente y puede ejecutar las instrucciones en hardware utilizando uno o más procesadores para llevar a cabo las técnicas de esta divulgación. Cualquiera del codificador de vídeo 20 y el decodificador de vídeo 30 puede integrarse como parte de un codificador/decodificador combinado (CODEC) en un solo dispositivo, por ejemplo, como se muestra en la Fig. 1B.

El dispositivo fuente 12 y el dispositivo de destino 14 pueden comprender cualquiera de una amplia variedad de dispositivos, incluyendo cualquier tipo de dispositivos manuales o estacionarios, por ejemplo ordenadores notebook o portátil, teléfonos móviles, teléfonos inteligentes, Tabletas u ordenadores de Tableta, cámaras, ordenadores de escritorio, descodificadores, televisiones, dispositivos de exhibición, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión por secuencias de vídeo (tal como servidores de servicios de contenido o servidores de entrega de contenido), dispositivo receptor de difusión, dispositivo transmisor de difusión, o similar y puede no utilizar algún tipo de sistema operativo. En algunos casos, el dispositivo fuente 12 y el dispositivo de destino 14 pueden equiparse para comunicación inalámbrica. Así, el dispositivo fuente 12 y el dispositivo de destino 14 pueden ser dispositivos de comunicación inalámbrica.

En algunos casos, el sistema de codificado de vídeo 10 ilustrado en la Fig. 1A es meramente un ejemplo y las técnicas de la presente solicitud pueden aplicarse a ajustes de codificado de vídeo (p. ej., codificación de vídeo o decodificación de vídeo) que no necesariamente incluyen cualquier comunicación de datos entre los dispositivos de codificación y decodificación. En otros ejemplos, pueden recuperarse datos de una memoria local, transmitirse por una red, o algo semejante. Un dispositivo de codificación de vídeo puede codificar y almacenar datos en una memoria y/o un dispositivo de decodificación de vídeo puede recuperar y decodificar datos de la memoria. En algunos ejemplos, codificación y decodificación se realizan por dispositivos que no se comunican entre sí, sino que simplemente codifican datos en una memoria y/o recuperan y decodifican datos de la memoria.

Por conveniencia de la descripción, en esta memoria se describen realizaciones de la invención, por ejemplo, por referencia a codificado de vídeo de Alta Eficiencia (HEVC) o al software de referencia del codificado de vídeo Versátil (VVC), la norma de codificado de vídeo de siguiente generación desarrollada por el Equipo de Colaboración Conjunto en codificado de vídeo (JCT-VC) del Grupo de Expertos de codificado de vídeo ITU-T (VCEG) y el Grupo de Expertos de Imágenes en Movimiento ISO/IEC (MPEG). Un experto en la técnica entenderá que realizaciones de la invención no se limitan a HEVC o VVC.

#### Codificador y método de codificación

La Fig. 2 muestra un diagrama de bloques esquemático de un codificador de vídeo ejemplar 20 que se configura para implementar las técnicas de la presente solicitud. En el ejemplo de la Fig. 2, el codificador de vídeo 20 comprende una entrada 201 (o interfaz de entrada 201), una unidad de cálculo residual 204, una unidad de procesamiento de transformada 206, una unidad de cuantización 208, una unidad de cuantización inversa 210, y una unidad de procesamiento de transformada inversa 212, una unidad de reconstrucción 214, una unidad de filtro de bucle 220, un búfer de imágenes decodificadas (DPB) 230, una unidad de selección de modo 260, una unidad de codificación entrópica 270, y una salida 272 (o interfaz de salida 272). La unidad de selección de modo 260 puede incluir una unidad de interpretación 244, una unidad de intrapredicción 254 y una unidad de división 262. La unidad de interpretación 244 puede incluir una unidad de estimación de movimiento y una unidad de compensación de movimiento (que no se muestra). El codificador de vídeo 20 como se muestra en la Fig. 2 también se pueden denominar codificador de vídeo híbrido o un codificador de vídeo según un códec de vídeo híbrido.

La unidad de cálculo residual 204, la unidad de procesamiento de transformada 206, la unidad de cuantización 208, la unidad de selección de modo 260 pueden ser referidas como que forman una trayectoria de señal de avance del codificador 20, mientras que la unidad de cuantización inversa 210, la unidad de procesamiento de transformada inversa 212, la unidad de reconstrucción 214, el búfer 216, el filtro de bucle 220, el búfer de imágenes decodificadas (DPB) 230, la unidad de interpretación 244 y la unidad de intrapredicción 254 pueden ser referidas como que forman una trayectoria de señal de retorno del codificador de vídeo 20, en donde la trayectoria de señal de retorno del codificador de vídeo 20 corresponde a la trayectoria de señal del decodificador (ver decodificador de vídeo 30 en la Fig. 3). La unidad de cuantización inversa 210, la unidad de procesamiento de transformada inversa 212, la unidad de reconstrucción 214, el filtro de bucle 220, el búfer de imágenes decodificadas (DPB) 230, la unidad de interpretación 244 y la unidad de intrapredicción 254 también se refieren a la formación del "decodificador integrado" del codificador de vídeo 20.

Imágenes y división de imagen (Imágenes y Bloques)

El codificador 20 puede configurarse para recibir, por ejemplo a través de la entrada 201, una imagen 17 (o datos de imagen 17), por ejemplo imagen de una secuencia de imágenes que forman un vídeo o secuencia de vídeo. La imagen o datos de imagen recibidos también pueden ser una imagen preprocesada 19 (o datos de imagen preprocesados 19). Para propósitos de simplicidad la siguiente descripción se refiere a la imagen 17. La imagen 17 también pueden denominarse imagen actual o imagen a codificar (en particular en el codificado de vídeo para distinguir la imagen actual de otras imágenes, por ejemplo imágenes previamente codificadas y/o decodificadas de la misma secuencia de vídeo, es decir la secuencia de vídeo que también comprende la imagen actual).

Una imagen (digital) es o puede ser vista como una distribución bidimensional o matriz de muestras con valores de intensidad. Una muestra en la distribución también pueden denominarse píxel (forma corta de elemento de imagen) o un pel. El número de muestras en la dirección (o eje) horizontal y vertical de la distribución o imagen definen el tamaño y/o resolución de la imagen. Para representación de color, típicamente se emplean tres componentes de color, es decir la imagen puede ser representada o puede incluir tres distribuciones de muestra. En formato o espacio de color RGB una imagen comprende una distribución de muestra rojo, verde y azul correspondiente. Sin embargo, en codificado de vídeo cada píxel típicamente se representa en un formato de luminancia y crominancia o espacio de color, p. ej. YCbCr, que comprende un componente de luminancia indicado por Y (a veces en cambio también se usa L) y dos componentes de crominancia indicados por Cb y Cr. El componente de luminancia (o luma por abreviar) Y representa la intensidad de nivel de gris o brillo (p. ej. como en una imagen en escala de grises), mientras los dos componentes de crominancia (o croma por abreviar) Cb y Cr representan los componentes de información de cromaticidad o color. Por consiguiente, la imagen en formato YCbCr comprende una distribución de muestras de luminancia de valores de muestra de luminancia (Y) y dos distribuciones de muestras de crominancia de valores de crominancia (Cb y Cr). Imágenes en formato RGB pueden convertirse o transformarse en formato YCbCr y viceversa, el proceso también se conoce como transformación o conversión de color. Si una imagen es monocromática, la imagen puede comprender solamente una distribución de muestras de luminancia. Por consiguiente, una imagen puede ser, por ejemplo, una distribución de muestras de luminancia en formato monocromático o una distribución de muestras de luminancia y dos distribuciones correspondientes de muestras de crominancia en formato de color 4:2:0, 4:2:2, y 4:4:4.

Realizaciones del codificador de vídeo 20 pueden comprender una unidad de división de imagen (no representada en la Fig. 2) configurada para dividir la imagen 17 en una pluralidad de bloques de imagen 203 (típicamente no se superponen). Estos bloques también pueden denominarse bloques raíz, macrobloques (H.264/AVC) o bloques de árbol de codificado (CTB) o unidades de árbol de codificado (CTU) (H.265/HEVC y VVC). La unidad de división de imagen puede configurarse para utilizar el mismo tamaño de bloque para todas las imágenes de una secuencia de vídeo y la rejilla correspondiente que define el tamaño de bloque, o para cambiar el tamaño de bloque entre imágenes o subconjuntos o grupos de imágenes, y dividir cada imagen en los bloques correspondientes.

En realizaciones adicionales, el codificador de vídeo puede configurarse para recibir directamente un bloque 203 de la imagen 17, por ejemplo uno, varios o todos los bloques que forman la imagen 17. El bloque de imagen 203 también puede denominarse bloque de imagen actual o bloque de imagen a codificar.

Al igual que la imagen 17, el bloque de imagen 203 una vez más es o puede ser visto como una distribución bidimensional o matriz de muestras con valores de intensidad (valores de muestra), aunque de dimensión más pequeña que la imagen 17. En otras palabras, el bloque 203 puede comprender, por ejemplo, una distribución de muestras (por ejemplo una distribución de luminancia en caso de una imagen monocromática 17, o una distribución de luminancia o crominancia en el caso de una imagen de color) o tres distribuciones de muestras (por ejemplo una distribución de luminancia y dos distribuciones de crominancia en el caso de una imagen de color 17) o cualquier otro número y/o tipo de distribuciones dependiendo del formato de color aplicado. El número de muestras en dirección (o eje) horizontal y vertical del bloque 203 define el tamaño de bloque 203. Por consiguiente, un bloque, por ejemplo, puede ser una distribución de muestras MxN (M-columna por N-fila), o una distribución de coeficientes de transformada MxN.

Realizaciones del codificador de vídeo 20, como se muestra en la Fig. 2, pueden configurarse para codificar la imagen 17 bloque por bloque, por ejemplo, la codificación y la predicción se realiza por el bloque 203.

Realizaciones del codificador de vídeo 20 como se muestra en la Fig. 2 pueden configurarse adicionalmente para dividir y/o codificar la imagen utilizando sectores (también referidas como sectores de vídeo), en donde una imagen puede dividirse o codificarse utilizando uno o más sectores (típicamente sin superponerse), y cada sector puede comprender uno o más bloques (por ejemplo, CTU).

Realizaciones del codificador de vídeo 20 tal como se muestra en la Fig. 2 pueden configurarse adicionalmente



- para dividir y/o codificar la imagen utilizando grupos de mosaicos (también denominados grupos de mosaicos de vídeo) y/o mosaicos (también denominados mosaicos de vídeo), en donde una imagen puede dividirse en o codificarse utilizando uno o más grupos de mosaicos (típicamente sin superponerse), y cada grupo de mosaicos puede comprender, por ejemplo, uno o más bloques (por ejemplo CTU) o uno o más mosaicos, en donde cada mosaico, por ejemplo puede ser de forma rectangular y puede comprender uno o más bloques (por ejemplo CTU), por ejemplo bloques completos o fraccionales.

#### Cálculo residual

- La unidad de cálculo residual 204 puede configurarse para calcular un bloque residual 205 (también denominado residual 205) con base en el bloque de imagen 203 y un bloque de predicción 265 (detalles adicionales referentes al bloque de predicción 265 se proporcionan más adelante), por ejemplo sustrayendo valores de muestra del bloque de predicción 265 de los valores de muestra del bloque de imagen 203, muestra por muestra (píxel por píxel) para obtener el bloque residual 205 en el dominio de muestra.

#### Transformada

- La unidad de procesamiento de transformada 206 se puede configurar para aplicar una transformada, p. ej., una transformada de coseno discreta (DCT) o una transformada de seno discreta (DST), en los valores de muestra del bloque residual 205 para obtener coeficientes de transformada 207 en un dominio de transformadas. Los coeficientes de transformada 207 también se pueden denominar coeficientes residuales de transformada y representan el bloque residual 205 en el dominio de transformada.

- La unidad de procesamiento de transformada 206 se puede configurar para aplicar aproximaciones enteras de DCT/DST, tales como transformadas especificadas para H.265/HEVC. Comparado con una transformada DCT ortogonal, tales aproximaciones a entero típicamente se escalan por cierto factor. A fin de preservar la norma del bloque residual que se procesa por transformadas de avance e inversa, se aplican factores de escalado adicionales como parte del proceso de transformada. Los factores de escalado típicamente se eligen con base en ciertas restricciones tales como factores de escalado que son una potencia de dos para operaciones de cambio, profundidad de bit de los coeficientes de transformada, compensación entre precisión y costes de implementación, etc. Factores de escalado específicos, por ejemplo, son especificados para la transformada inversa, por ejemplo por la unidad de procesamiento de transformada inversa 212 (y la transformada inversa correspondiente, por ejemplo por la unidad de procesamiento de transformada inversa 312 en el decodificador de vídeo 30) y factores de escalado correspondientes para la transformada de avance, por ejemplo por la unidad de procesamiento de transformada 206, en un codificador 20 por consiguiente pueden especificarse.

- Realizaciones del codificador de vídeo 20 (respectivamente la unidad de procesamiento de transformada 206) pueden configurarse para emitir parámetros de transformada, por ejemplo un tipo de transformada o transformadas, por ejemplo directamente o codificadas o comprimidas a través de la unidad de codificación por entropía 270, de manera que, por ejemplo, el decodificador de vídeo 30 puede recibir y utilizar los parámetros de transformada para decodificación.

#### Cuantización

- La unidad de cuantización 208 puede configurarse para cuantizar los coeficientes de transformada 207 para obtener coeficientes cuantizados 209, por ejemplo, aplicando cuantización escalar o cuantización vectorial. Los coeficientes cuantizados 209 también pueden denominarse coeficientes de transformada cuantizados 209 o coeficientes residuales cuantizados 209.
- El proceso de cuantización puede reducir la profundidad de bits asociada con algunos o todos los coeficientes de transformada 207. Por ejemplo, un coeficiente de transformada de n bits puede redondearse hacia abajo a un coeficiente de transformada de m bits durante la cuantización, donde n es mayor que m. El grado de cuantización puede modificarse ajustando un parámetro de cuantización (QP). Por ejemplo, para la cuantización escalar, se puede aplicar una escala diferente para lograr una cuantización más fina o más basta.
- Los tamaños de etapa de cuantización más pequeños corresponden a una cuantización más fina, mientras que los tamaños de etapa de cuantización más grandes corresponden a una cuantización más basta. El tamaño de la etapa de cuantización aplicable puede indicarse mediante un parámetro de cuantización (QP). El parámetro de cuantización puede ser, por ejemplo, un índice de un conjunto predefinido de tamaños de etapa de cuantización aplicables. Por ejemplo, los parámetros de cuantización pequeños pueden corresponder a una cuantización fina (tamaños de etapas de cuantización pequeños) y los parámetros de cuantización grandes pueden corresponder a una cuantización basta (tamaños de etapas de cuantización grandes) o viceversa. La cuantización puede incluir división por un tamaño de etapa de cuantización y una descuantización correspondiente y/o inversa, por ejemplo por la unidad de cuantización inversa 210, puede incluir multiplicación por el tamaño de etapa de cuantización. Realizaciones según algunas normas, por ejemplo HEVC, pueden configurarse para utilizar un parámetro de cuantización con el fin de determinar el tamaño de etapa de cuantización. Generalmente, el tamaño de etapa de cuantización puede calcularse con base en un parámetro

- de cuantización utilizando una aproximación de punto fijo de una ecuación incluyendo división. Factores de escalado adicionales pueden introducirse para cuantización y descuantización con el fin de restaurar la norma del bloque residual que pudiera modificarse debido al escalado utilizado en la aproximación de punto fijo de la ecuación para tamaño de etapa de cuantización y parámetro de cuantización. En un ejemplo de implementación, el escalado de la transformada inversa y la descuantización podrían combinarse. Alternativamente, tablas de cuantización personalizadas pueden utilizarse y señalizadas desde un codificador a un decodificador, por ejemplo en un flujo de bits. La cuantización es una operación con pérdida, en donde la pérdida aumenta con el incremento de los tamaños de etapa de cuantización.
- 5
- 10 Realizaciones del codificador de vídeo 20 (respectivamente, unidad de cuantización 208) pueden configurarse para emitir parámetros de cuantización (QP), por ejemplo directamente o codificarse a través de la unidad de codificación por entropía 270, de manera que, por ejemplo, el decodificador de vídeo 30 puede recibir y aplicar los parámetros de cuantización para decodificación.
- 15 Cuantización inversa
- La unidad de cuantización inversa 210 se configura para aplicar la cuantización inversa de la unidad de cuantización 208 sobre los coeficientes cuantizados para obtener coeficientes descuantizados 211, por ejemplo, aplicando la inversa del esquema de cuantización aplicado por la unidad de cuantización 208 en función o usando el mismo tamaño de etapa de cuantización que la unidad de cuantización 208. Los coeficientes descuantizados 211 también pueden denominarse coeficientes residuales descuantizados 211 y corresponden - aunque típicamente no son idénticos a los coeficientes de transformada debido a la pérdida por cuantización - a los coeficientes de transformada 207.
- 20
- 25 Transformada inversa
- La unidad de procesamiento de transformada inversa 212 se configura para aplicar la transformada inversa de la transformada aplicada por la unidad de procesamiento de transformada 206, por ejemplo una transformada inversa de coseno discreta (DCT) o transformada inversa de seno discreta (DST), u otras transformadas inversas para obtener un bloque residual reconstruido 213 (o coeficientes descuantizados correspondientes 213) en el dominio de muestra. El bloque residual reconstruido 213 también pueden denominarse bloque de transformada 213.
- 30
- Reconstrucción
- La unidad de reconstrucción 214 (por ejemplo sumadora 214) se configura para añadir el bloque de transformada 213 (es decir bloque residual reconstruido 213) al bloque de predicción 265 para obtener un bloque reconstruido 215 en el dominio de muestra, por ejemplo añadiendo, muestra por muestra, los valores de muestra del bloque residual reconstruido 213 y los valores de muestra del bloque de predicción 265.
- 35
- 40 Filtrado
- La unidad de filtro de bucle 220 (o para abreviar "filtro de bucle" 220), se configura para filtrar el bloque reconstruido 215 para obtener un bloque filtrado 221, o, en general, para filtrar muestras reconstruidas para obtener muestras filtradas. La unidad de filtro de bucle, por ejemplo, se configura para suavizar las transiciones de píxel, o de otra manera mejorar la calidad del vídeo. La unidad de filtro de bucle 220 puede comprender uno o más filtros de bucle tal como un filtro de des-bloqueo, un filtro de desfase adaptable a la muestra (SAO) o uno o más filtros diferentes, por ejemplo, un filtro bilateral, un filtro de bucle adaptable (ALF), un filtro de agudización, un filtro de suavizado o filtros colaborativos, o cualquier combinación de los mismos. Aunque la unidad de filtro de bucle 220 se muestra en la FIG. 2 como que es un filtro en bucle, en otras configuraciones, la unidad de filtro de bucle 220 puede implementarse como un filtro posbucle. El bloque filtrado 221 también puede denominarse bloque reconstruido filtrado 221.
- 45
- 50 Realizaciones del codificador de vídeo 20 (respectivamente unidad de filtro de bucle 220) pueden configurarse para emitir parámetros de filtro de bucle (tal como información de compensación de muestra adaptable), por ejemplo directamente o codificados a través de la unidad de codificación por entropía 270, de manera que, por ejemplo, un decodificador 30 puede recibir y aplicar los mismos parámetros de filtro de bucle o filtros de bucle respectivos para decodificación.
- 55
- 60 Búfer de imágenes decodificadas
- El búfer de imágenes decodificadas (DPB) 230 puede ser una memoria que almacena imágenes de referencia o en general datos de imágenes de referencia para codificar datos de vídeo a través del codificador de vídeo 20. La DPB 230 puede formarse por cualquiera de una variedad de dispositivos de memoria, tal como memoria de acceso aleatorio dinámica (DRAM), incluyendo DRAM sincronizada (SDRAM), RAM magneto-resistiva (MRAM), RAM resistiva (RRAM), u otros tipos de dispositivos de memoria. El búfer de imágenes decodificadas
- 65

(DPB) 230 puede configurarse para almacenar uno o más bloques filtrados 221. El búfer de imágenes decodificadas 230 además puede configurarse para almacenar otros bloques previamente filtrados, por ejemplo, bloques previamente reconstruidos y filtrados 221, de la misma imagen actual o de imágenes diferentes, por ejemplo imágenes previamente reconstruidas, y puede proporcionar imágenes completas previamente reconstruidas, es decir decodificadas, (y bloques de referencia y muestras correspondientes) y/o una imagen actual parcialmente reconstruida (y bloques de referencia y muestras correspondientes), por ejemplo para interpredicción. El búfer de imágenes decodificadas (DPB) 230 también puede configurarse para almacenar uno o más bloques reconstruidos no filtrados 215, o en general muestras reconstruidas no filtradas, por ejemplo si el bloque reconstruido 215 no es filtrado por la unidad de filtro de bucle 220, o cualquier otra versión adicional procesada de los bloques reconstruidos o muestras.

#### Selección de Modo (División y predicción)

La unidad de selección de modo 260 comprende una unidad de división 262, una unidad de interpredicción 244 y una unidad de intrapredicción 254, y se configura para recibir u obtener datos de imagen originales, por ejemplo un bloque original 203 (bloque actual 203 de la imagen actual 17), y datos de imagen reconstruidos por ejemplo muestras reconstruidas filtradas y/o no filtradas o bloques de la misma imagen (actual) y/o de una o una pluralidad de imágenes previamente decodificadas, por ejemplo del búfer de imágenes decodificadas 230 u otros búferes (por ejemplo búfer de línea, que no se muestra). Los datos de imagen reconstruidos se utilizan como datos de imagen de referencia para predicción, por ejemplo interpredicción o intrapredicción, para obtener un bloque de predicción 265 o predictor 265.

La unidad de selección de modo 260 puede configurarse para determinar o seleccionar una división para un modo de predicción de bloque actual (incluyendo la no división) y un modo de predicción (por ejemplo un modo de intra- o interpredicción) y generar un bloque de predicción correspondiente 265, que se utiliza para el cálculo del bloque residual 205 y para la reconstrucción del bloque reconstruido 215.

Realizaciones de la unidad de selección de modo 260 pueden configurarse para seleccionar el modo de división y predicción (por ejemplo entre aquellos soportados por la unidad de selección de modo 260, o disponibles para esta), que proporcione el mejor emparejamiento o, en otras palabras, el residual mínimo (residual mínimo significa mejor compresión para transmisión o almacenamiento), o una sobrecarga de señalización mínima (sobrecarga de señalización mínima significa mejor compresión para transmisión o almacenamiento), o lo que considere o equilibre ambos. La unidad de selección de modo 260 puede configurarse para determinar el modo de división y predicción con base en la optimización de tasa-distorsión (RDO), es decir seleccionar el modo de predicción que proporcione una tasa-distorsión mínima. Términos tales como "mejor", "mínimo", "óptimo" etc. en este contexto no necesariamente se refieren a un "mejor", "mínimo", "óptimo", etc. global sino que también se pueden referir al cumplimiento de un criterio de terminación o selección tal como un valor que excede o que queda por debajo de un umbral u otras restricciones que potencialmente conducen a una "selección subóptima" pero reduciendo la complejidad y el tiempo de procesamiento.

En otras palabras, la unidad de división 262 puede configurarse para dividir el bloque 203 en unidades de bloque más pequeñas o subbloques (que forman bloques nuevamente), por ejemplo por ejemplo utilizando iterativamente división de árbol cuádruple (QT), división binaria (BT) o división de árbol triple (TT) o cualquier combinación de las mismas, y realizar, por ejemplo, la predicción para cada una de las divisiones de bloque o subbloques, en donde la selección de modo comprende la selección de la estructura de árbol del bloque dividido 203 y los modos de predicción son aplicados a cada una de las divisiones de bloque actual o subbloques.

En lo sucesivo, el procesamiento de división (por ejemplo por la unidad de división 260) y predicción (por la unidad de interpredicción 244 y la unidad de intrapredicción 254) realizado por un codificador de vídeo ejemplar 20 se explicará con más detalle.

#### División

La unidad de división 262 puede dividir (o partir ) un bloque actual 203 en divisiones más pequeñas, por ejemplo, bloques más pequeños de tamaño cuadrado o rectangular. Estos bloques más pequeños (que también pueden denominarse subbloques) se pueden dividir adicionalmente en divisiones incluso más pequeñas. Esta también se refiere a división de árbol o división de árbol jerárquica, en donde un bloque raíz, por ejemplo, al nivel de árbol raíz 0 (nivel de jerarquía 0, profundidad 0), puede dividirse de manera recursiva, por ejemplo puede dividirse en dos o más bloques de un siguiente nivel de árbol inferior, por ejemplo, nodos en el nivel de árbol 1 (nivel de jerarquía 1, profundidad 1), en donde estos bloques nuevamente pueden dividirse en dos o más bloques de un siguiente nivel inferior, por ejemplo nivel de árbol 2 (nivel de jerarquía 2, profundidad 2), etc. hasta que la división finaliza, por ejemplo, debido a que se cumple un criterio de terminación, por ejemplo se alcanza una profundidad de árbol máxima o tamaño de bloque mínimo. Bloques que no se dividen adicionalmente también se denominan bloques-hoja o nodos hoja del árbol. Un árbol que utiliza la división en dos divisiones se denomina árbol binario (BT), un árbol que utiliza la división en tres

divisiones se refiere como árbol ternario (TT), y un árbol que utiliza la división en cuatro divisiones se refiere como árbol cuádruple (QT).

5 Tal como se ha mencionado antes, el término "bloque" tal como se utiliza en esta memoria puede ser una parte, en particular una parte cuadrada o rectangular, de una imagen. Con referencia, por ejemplo, a HEVC y VVC, el bloque puede ser o corresponder a una unidad de árbol de codificación (CTU), una unidad de codificación (CU), unidad de predicción (PU), y unidad de transformada (TU) y/o a los bloques correspondientes, por ejemplo, un bloque de árbol de codificación (CTB), un bloque de codificado (CB), un bloque de transformada (TB) o bloque de predicción (PB).

10 Por ejemplo una unidad de árbol de codificación (CTU) puede ser o comprender un CTB de muestras de luminancia, dos CTB correspondientes de muestras de crominancia de una imagen que tiene tres distribuciones de muestra, o un CTB de muestras de una imagen monocromática o una imagen que se codifica utilizando tres planos de color separados y estructuras de sintaxis utilizadas para codificar las muestras. De manera correspondiente, un bloque de árbol de codificación (CTB) puede ser un bloque de muestras NxN para cierto valor de N de manera que la división de un componente en CTB es una división. Una unidad de codificación (CU) puede ser o comprender un bloque de codificado de muestras de luminancia, dos bloques de codificado correspondientes de muestras de crominancia de una imagen que tiene tres distribuciones de muestra, o un bloque de codificado de muestras de una imagen monocromática o una imagen que se codifica utilizando tres planos de color separados y estructuras de sintaxis utilizadas para codificar las muestras. De manera correspondiente, un bloque de codificado (CB) puede ser un bloque de muestras MxN para algunos valores de M y N de manera que la división de un CTB en bloques de codificado es una división.

25 En realizaciones, por ejemplo, según HEVC, una unidad de árbol de codificación (CTU) puede dividirse en CU utilizando una estructura de árbol cuádruple denotada como árbol de codificación. La decisión respecto a si se codifica un área de imagen utilizando predicción de interimagen (temporal) o intraimagen (espacial) se realiza al nivel de CU. Cada CU puede dividirse adicionalmente en una, dos o cuatro PU según el tipo de división de PU. Dentro de una PU, se aplica el mismo proceso de predicción y la información relevante es transmitida al decodificador sobre una base de PU. Después de obtener el bloque residual aplicando el proceso de predicción con base en el tipo de división de PU, una CU puede dividirse en unidades de transformada (TU) según otra estructura de árbol cuádruple similar al árbol de codificado para la CU.

35 En realizaciones, por ejemplo, según la última norma de codificado de vídeo actualmente en desarrollo, que se refiere como codificado de vídeo Versátil (VVC), la división de árbol cuádruple y árbol binario combinada (QTBT) se utiliza, por ejemplo, para dividir un bloque de codificado. En la estructura de bloque QTBT, una CU puede tener una forma cuadrada o rectangular. Por ejemplo, una unidad de árbol de codificado (CTU) primero se divide por una estructura de árbol cuádruple. Los nodos hoja de árbol cuádruple se dividen adicionalmente por una estructura de árbol binario o árbol ternario (o triple). Los nodos hoja de árbol de división se denominan unidades de codificado (CU), y esa segmentación se utiliza para procesamiento de predicción y transformada sin división adicional. Esto significa que la CU, PU y TU tienen el mismo tamaño de bloque en la estructura de bloque de codificado QTBT. En paralelo, la división múltiple, por ejemplo, la división de árbol triple puede utilizarse junto con la estructura de bloque QTBT.

45 En un ejemplo, la unidad de selección de modo 260 del codificador de vídeo 20 se puede configurar para realizar cualquier combinación de las técnicas de división descritas en esta memoria.

50 Tal como se ha descrito antes, el codificador de vídeo 20 se configura para determinar o seleccionar el mejor modo de predicción o un modo de predicción óptimo entre un conjunto de modos de predicción (por ejemplo, predeterminado). El conjunto de modos de predicción puede comprender, por ejemplo, modos de intrapredicción y/o modos de interpredicción.

#### Intrapredicción

55 El conjunto de modos de intrapredicción puede comprender 35 modos de intrapredicción diferentes, por ejemplo modos no direccionales tales como el modo DC (o medio) y el modo plano o modos direccionales, por ejemplo conforme a lo definido en HEVC, o puede comprender 67 modos de intrapredicción diferentes, por ejemplo modos no direccionales tales como el modo DC (o medio) y modo plano, o modos direccionales, por ejemplo conforme a lo definido para VVC.

60 La unidad de intrapredicción 254 se configura para utilizar muestras reconstruidas de bloques vecinos de la misma imagen actual para generar un bloque de intrapredicción 265 según un modo de intrapredicción del conjunto de modos de intrapredicción.

65 La unidad de intrapredicción 254 (o en general la unidad de selección de modo 260) se configura adicionalmente para emitir parámetros de intrapredicción (o en general información indicativa del modo de intrapredicción seleccionado para el bloque actual) a la unidad de codificación por entropía 270 en forma de

elementos de sintaxis 266 para inclusión en los datos de imagen codificados 21, de manera que, por ejemplo, el decodificador de vídeo 30 puede recibir y utilizar los parámetros de predicción para decodificación.

#### Interpredicción

5

El conjunto de (o posibles) modos de interpredicción depende de las imágenes de referencia disponibles (es decir imágenes previas al menos parcialmente decodificadas, por ejemplo almacenadas en DPB 230) y otros parámetros de interpredicción, por ejemplo ya sea toda la imagen de referencia o únicamente una parte, por ejemplo un área de ventana de búsqueda alrededor del área del bloque actual, de la imagen de referencia, se utiliza para buscar un mejor bloque de referencia de emparejamiento, y/o por ejemplo si se aplica interpolación de píxel, por ejemplo interpolación de medio/semi-pel, y/o interpolación de cuarto-pel, o no.

10

Adicional a los modos de predicción anteriores, se puede aplicar el modo de salto, y/o modo directo.

15

La unidad de interpredicción 244 puede incluir una unidad de estimación de movimiento (ME) y una unidad de compensación de movimiento (MC) (ninguna se muestra en la Fig. 2). La unidad de estimación de movimiento puede configurarse para recibir u obtener el bloque de imagen 203 (bloque de imagen actual 203 de la imagen actual 17) y una imagen decodificada 231, o al menos uno o una pluralidad de bloques previamente reconstruidos, por ejemplo bloques reconstruidos de uno o una pluralidad de otras/diferentes imágenes previamente decodificadas 231, para estimación de movimiento. Por ejemplo, una secuencia de vídeo puede comprender la imagen actual y las imágenes previamente decodificadas 231, o en otras palabras, la imagen actual y las imágenes previamente decodificadas 231 pueden ser parte o formar una secuencia de imágenes que forman una secuencia de vídeo.

20

25

El codificador 20, por ejemplo, puede configurarse para seleccionar un bloque de referencia de una pluralidad de bloques de referencia de las mismas o diferentes imágenes de la pluralidad de otras imágenes y proporcionar una imagen de referencia (o índice de imagen de referencia) y/o un desfase (desfase espacial) entre la posición (coordenadas x, y) del bloque de referencia y la posición del bloque actual como parámetros de interpredicción a la unidad de estimación de movimiento. Este desfase también se denomina vector de movimiento (MV).

30

La unidad de compensación de movimiento se configura para obtener, por ejemplo recibir, un parámetro de interpredicción y realizar interpredicción con base o utilizando el parámetro de interpredicción para obtener un bloque de interpredicción 265. La compensación de movimiento, realizada por la unidad de compensación de movimiento puede involucrar la extracción o generación del bloque de predicción con base en el vector de movimiento/bloque determinado por la estimación de movimiento, posiblemente realizando interpolaciones a precisión de submuestra. El filtrado de interpolación puede generar muestras adicionales de muestras conocidas, incrementando así potencialmente el número de bloques de predicción candidatos que pueden utilizarse para codificar un bloque de imagen. Al momento de recibir el vector de movimiento para la PU del bloque de imagen actual, la unidad de compensación de movimiento puede ubicar el bloque de predicción al que apunta el vector de movimiento en una de las listas de imágenes de referencia.

35

40

La unidad de compensación de movimiento también puede generar elementos de sintaxis asociados con los bloques y las sectores de vídeo para uso por el decodificador de vídeo 30 en la decodificación de los bloques de imagen del sector de vídeo. Además o como una alternativa a sectores y elementos de sintaxis respectivos, pueden generarse o utilizarse grupos y/o mosaicos y elementos de sintaxis respectivos.

45

#### Codificado de entrópica

50

La unidad de codificación entrópica 270 se configura para aplicar, por ejemplo, un algoritmo o esquema de codificado de entropía (por ejemplo, un esquema de codificado de longitud variable (VLC), un esquema de VLC adaptativo al contexto (CAVLC), un esquema de codificado aritmético, una binarización, un codificado aritmético binario adaptativo al contexto (CABAC), codificado aritmético binario adaptativo al contexto (SBAC) basado en sintaxis, codificado de entropía de división de intervalo de probabilidad (PIPE) u otra metodología o técnica de codificado de entropía) o derivación (sin compresión) en los coeficientes cuantizados 209, parámetros de interpredicción, parámetros de intrapredicción, parámetros de filtro de bucle y/u otros elementos de sintaxis para obtener datos de imagen codificados 21 que pueden emitirse a través de la salida 272, por ejemplo, en forma de un flujo de bits codificado 21, de modo que, por ejemplo, el decodificador de vídeo 30 puede recibir y utilizar los parámetros para decodificar. El flujo de bits codificado 21 puede transmitirse al decodificador de vídeo 30, o almacenarse en una memoria para transmisión o recuperación posterior por el decodificador de vídeo 30.

55

60

Pueden utilizarse otras variaciones estructurales del codificador de vídeo 20 para codificar el flujo de vídeo. Por ejemplo un codificador basado en no-transformada 20 puede cuantizar la señal residual directamente sin la unidad de procesamiento de transformada 206 para ciertos bloques o fotogramas. En otra implementación, un codificador 20 puede tener la unidad de cuantización 208 y la unidad de cuantización inversa 210

65

combinadas en una sola unidad.

#### Decodificador y método de decodificación

5 La Fig. 3 muestra un ejemplo de un decodificador de vídeo 30 que se configura para implementar las técnicas de esta solicitud. El decodificador de vídeo 30 se configura para recibir los datos de imagen codificada 21 (por ejemplo flujo de bits codificado 21), por ejemplo codificados por el codificador 20, para obtener una imagen decodificada 331. El flujo de bits o datos de imagen codificados comprenden información para decodificar los  
10 datos de imagen codificados, por ejemplo, datos que representan bloques de imagen de un sector de vídeo codificado y (y/o grupos de mosaicos o mosaicos) elementos de sintaxis asociados.

En el ejemplo de la Fig. 3, el decodificador 30 comprende una unidad de decodificación entrópica 304, una unidad de cuantización inversa 310, una unidad de procesamiento de transformada inversa 312, una unidad de reconstrucción 314 (por ejemplo una sumadora 314), un filtro de bucle 320, un búfer de imágenes decodificadas (DPB) 330, una unidad de aplicación de modo 360, una unidad de interpredicción 344 y una  
15 unidad de intrapredicción 354. La unidad de interpredicción 344 puede ser o incluir una unidad de compensación de movimiento. El decodificador de vídeo 30, en algunos ejemplos, puede realizar una pasada de decodificación generalmente recíproca a la pasada de codificación descrito con respecto al codificador de vídeo 100 de la FIG. 2.

20 Tal como se ha explicado con respecto al codificador 20, la unidad de cuantización inversa 210, la unidad de procesamiento de transformada inversa 212, la unidad de reconstrucción 214 el filtro de bucle 220, el búfer de imágenes decodificadas (DPB) 230, la unidad de interpredicción 344 y la unidad de intrapredicción 354 también son referidas como que forman el "decodificador integrado" del codificador de vídeo 20. Por consiguiente, la  
25 unidad de cuantización inversa 310 puede ser idéntica en función a la unidad de cuantización inversa 110, la unidad de procesamiento de transformada inversa 312 puede ser idéntica en función a la unidad de procesamiento de transformada inversa 212, la unidad de reconstrucción 314 puede ser idéntica en función a la unidad de reconstrucción 214, el filtro de bucle 320 puede ser idéntico en función al filtro de bucle 220, y el búfer de imágenes decodificadas 330 puede ser idéntica en función al búfer de imágenes decodificadas 230.  
30 Por lo tanto, las explicaciones proporcionadas para las unidades y funciones respectivas del codificador de vídeo 20 aplican de manera correspondiente a las unidades y funciones respectivas del decodificador de vídeo 30.

#### Decodificación entrópica

35 La unidad de decodificación entrópica 304 se configura para analizar el flujo de bits 21 (o en general, datos de imagen codificados 21) y realizar, por ejemplo, decodificación entrópica a los datos de imagen codificados 21 para obtener, por ejemplo, coeficientes cuantizados 309 y/o parámetros de codificación decodificados (no mostrados en la Fig. 3), por ejemplo cualquiera o todos los parámetros de interpredicción (por ejemplo, índice de imagen de referencia y vector de movimiento), parámetro de intrapredicción (por ejemplo, modo o índice de  
40 intrapredicción), parámetros de transformada, parámetros de cuantización, parámetros de filtro de bucle y/u otros elementos de sintaxis. La unidad de decodificación entrópica 304 puede configurarse para aplicar los algoritmos o esquemas de descodificación correspondientes a los esquemas de codificación como se describe con relación a la unidad de codificación entrópica 270 del codificador 20. La unidad de decodificación entrópica  
45 304 se puede configurar además para proporcionar parámetros de interpredicción, parámetros de intrapredicción y/u otros elementos de sintaxis a la unidad de aplicación de modo 360 y otros parámetros a otras unidades del decodificador 30. El decodificador de vídeo 30 puede recibir los elementos de sintaxis al nivel de sector de vídeo y/o al nivel de bloque de vídeo. Además o como una alternativa a sectores y elementos de sintaxis respectivos, pueden recibirse y/o utilizarse grupos de mosaicos y/o mosaicos y elementos de  
50 sintaxis respectivos.

#### Cuantización inversa

La unidad de unidad de cuantización inversa 310 puede configurarse para recibir parámetros de cuantización (QP) (o en general información relacionada con la cuantización inversa) y coeficientes cuantizados de los datos de imagen codificados 21 (por ejemplo mediante análisis y/o decodificación, por ejemplo por la unidad de decodificación entrópica 304) y aplicar, con base en los parámetros de cuantización, una cuantización inversa en los coeficientes cuantizados decodificados 309 para obtener coeficientes descuantizados 311, que también pueden denominarse coeficientes de transformada 311. El proceso de cuantización inversa puede incluir el uso  
55 de un parámetro de cuantización determinado por el codificador de vídeo 20 para cada bloque de vídeo en el sector de vídeo (o mosaico o grupo de mosaicos) para determinar un grado de cuantización y, de igual manera, un grado de cuantización inversa que debería aplicarse.  
60

#### Transformada inversa

65 La unidad de procesamiento de transformada inversa 312 puede configurarse para recibir coeficientes

descuantizados 311, también denominados coeficientes de transformada 311, y aplicar una transformada a los coeficientes descuantizados 311 para obtener bloques residuales reconstruidos 213 en el dominio de muestra. Los bloques residuales reconstruidos 213 también pueden denominarse bloques de transformada 313. La transformada puede ser una transformada inversa, por ejemplo, una DCT inversa, una DST inversa, una transformada de entero inversa, o un proceso de transformada inversa conceptualmente similar. La unidad de procesamiento de transformada inversa 312 se puede configurar además para recibir parámetros de transformada o información correspondiente de los datos de imagen codificados 21 (por ejemplo mediante análisis y/o decodificación, por ejemplo por la unidad de decodificación entrópica 304) para determinar la transformada que se va a aplicar a los coeficientes descuantizados 311.

#### Reconstrucción

La unidad de reconstrucción 314 (por ejemplo sumadora 314) puede configurarse para añadir el bloque residual reconstruido 313 al bloque de predicción 365 para obtener un bloque reconstruido 315 en el dominio de muestra, por ejemplo añadiendo los valores de muestra del bloque residual reconstruido 313 y los valores de muestra del bloque de predicción 365.

#### Filtrado

La unidad de filtro de bucle 320 (ya sea en el bucle de codificado o tras el bucle de codificado) se configura para filtrar el bloque reconstruido 315 para obtener un bloque filtrado 321, p. ej. para suavizar transiciones de píxeles o mejorar de otro modo la calidad de vídeo. La unidad de filtro de bucle 320 puede comprender uno o más filtros de bucle tal como un filtro de des-bloqueo, un filtro de desfase adaptable a la muestra (SAO) o uno o más filtros diferentes, por ejemplo, un filtro bilateral, un filtro de bucle adaptable (ALF), un filtro de agudización, un filtro de suavizado o filtros colaborativos, o cualquier combinación de los mismos. Aunque la unidad de filtro de bucle 320 se muestra en la FIG. 3 como que es un filtro en bucle, en otras configuraciones, la unidad de filtro de bucle 320 puede implementarse como un filtro posbucle.

#### Búfer de imágenes decodificadas

Los bloques de vídeo decodificados 321 de una imagen se almacenan entonces en búfer de imágenes decodificadas 330, que almacena las imágenes decodificadas 331 como imágenes de referencia para posterior compensación de movimiento para otras imágenes y/o para emitir una exhibición respectivamente.

El decodificador 30 se configura para emitir la imagen decodificada 311, por ejemplo a través de la salida 312, para presentación o visualización a un usuario.

#### Predicción

La unidad de interpredicción 344 puede ser idéntica a la unidad de interpredicción 244 (en particular a la unidad de compensación de movimiento) y la unidad de intrapredicción 354 puede ser idéntica a la unidad de interpredicción 254 en función, y toma decisiones de división o división y predicción basada en los parámetros de división y/o predicción o información respectiva recibida desde los datos de imagen codificados 21 (por ejemplo, mediante análisis y/o decodificación, por ejemplo, por la unidad de decodificación entrópica 304). La unidad de aplicación de modo 360 puede configurarse para realizar la predicción (intra- o interpredicción) por bloque con base en imágenes reconstruidas, bloques o muestras respectivas (filtradas o no filtradas) para obtener el bloque de predicción 365.

Cuando el sector de vídeo se codifica como un sector intracodificada (I), la unidad de intrapredicción 354 de la unidad de aplicación de modo 360 se configura para generar el bloque de predicción 365 para un bloque de imagen del sector de vídeo actual con base en un modo de intrapredicción señalizado y datos de bloques previamente decodificados de la imagen actual. Cuando la imagen de vídeo se codifica como sector intercodificada (es decir, B, o P), la unidad de interpredicción 344 (por ejemplo unidad de compensación de movimiento) de la unidad de aplicación de modo 360 se configura para producir bloques de predicción 365 para un bloque de vídeo del sector de vídeo actual con base en los vectores de movimiento y otros elementos de sintaxis recibidos desde la unidad de decodificación entrópica 304. Para interpredicción, los bloques de predicción pueden producirse de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El decodificador de vídeo 30 puede construir las listas de fotogramas de referencia, Lista 0 y Lista 1, usando técnicas de construcción predeterminadas basadas en imágenes de referencia almacenadas en DPB 330. Lo mismo o algo similar puede aplicarse para o por realizaciones utilizando grupos de mosaicos (por ejemplo grupos de mosaicos de vídeo) y/o mosaicos (por ejemplo mosaicos de vídeo) además o alternativamente a sectores (por ejemplo sectores de vídeo), por ejemplo un vídeo puede codificarse utilizando grupos de mosaicos I, P o B y/o mosaicos.

La unidad de aplicación de modo 360 se configura para determinar la información de predicción para un bloque de vídeo del sector de vídeo actual analizando los vectores de movimiento o información relacionada y otros

elementos de sintaxis, y utiliza la información de predicción para producir los bloques de predicción para que el bloque de vídeo actual sea decodificado. Por ejemplo la unidad de aplicación de modo 360 utiliza algunos de los elementos de sintaxis recibidos para determinar un modo de predicción (por ejemplo, intra- o interpredicción) utilizado para codificar los bloques de vídeo del sector de vídeo, un tipo de sector de interpredicción (por ejemplo, sector B, sector P, o sector GPB), información de construcción para una o más de las listas de imágenes de referencia para la sector, vectores de movimiento para cada bloque de vídeo intercodificado de la sector, estatus de interpredicción para cada bloque de vídeo intercodificado de la parte, y otra información para decodificar los bloques de vídeo en el sector de vídeo actual. Lo mismo o algo similar puede aplicarse para o por realizaciones utilizando grupos de mosaicos (por ejemplo grupos de mosaicos de vídeo) y/o mosaicos (por ejemplo mosaicos de vídeo) además o alternativamente a sectores (por ejemplo sectores de vídeo), por ejemplo un vídeo puede codificarse utilizando grupos de mosaicos I, P o B y/o mosaicos.

Realizaciones del decodificador de vídeo 30 como se muestra en la Fig. 3 pueden configurarse para dividir y/o decodificar la imagen utilizando sectores (también denominados sectores de vídeo), en donde una imagen puede dividirse o decodificarse utilizando uno o más sectores (típicamente sin superponerse), y cada sector puede comprender uno o más bloques (por ejemplo CTU).

Realizaciones del decodificador de vídeo 30, como se muestra en la Fig. 3, pueden configurarse para dividir y/o decodificar la imagen utilizando grupos de mosaicos (también denominados grupos de mosaicos de vídeo) y/o mosaicos (también denominados mosaicos de vídeo), en donde una imagen puede dividirse o decodificarse utilizando uno o más grupos de mosaicos (típicamente sin superponerse), y cada grupo de mosaicos puede comprender, por ejemplo uno o más bloques (por ejemplo CTU) o uno o más mosaicos, en donde cada mosaico, por ejemplo puede ser de forma rectangular y puede comprender uno o más bloques (por ejemplo CTU), por ejemplo bloques completos o fraccionales.

Otras variaciones del decodificador de vídeo 30 pueden utilizarse para decodificar los datos de imagen codificados 21. Por ejemplo el decodificador 30 puede producir la salida del flujo de vídeo sin la unidad de filtrado de bucle 320. Por ejemplo, un decodificador basado en no transformada 30 puede quantizar a la inversa la señal residual directamente sin la unidad de procesamiento de transformada inversa 312 para ciertos bloques o fotogramas. En otra implementación, el decodificador de vídeo 30 puede tener la unidad de cuantización inversa 310 y la unidad de procesamiento de transformada inversa 312 combinadas en una única unidad.

Se debería entender que, en el codificador 20 y el decodificador 30, un resultado de procesamiento de una etapa actual puede ser procesado adicionalmente y después emitido a la siguiente etapa. Por ejemplo después del filtrado de interpolación, la derivación de vector de movimiento o filtrado de bucle, se puede realizar una operación adicional, tal como Recorte o cambio, en el resultado de procesamiento del filtrado de interpolación, derivación de vector de movimiento o filtrado de bucle.

Se debería observar que operaciones adicionales pueden aplicarse a los vectores de movimiento derivados del bloque actual (incluyendo pero no limitado a vectores de movimiento de punto de control de modo afín, vectores de movimiento de subbloque en modos afín, plano ATMVP, vectores de movimiento temporales, y así sucesivamente). Por ejemplo, el valor del vector de movimiento queda restringido a un intervalo predefinido según su bit representativo. Si el bit representativo del vector de movimiento es bitDepth, entonces el intervalo es  $-2^A(\text{bitDepth}-1) \sim 2^A(\text{bitDepth}-1)-1$ , donde "A" significa exponenciación. Por ejemplo, si bitDepth se establece igual a 16, el intervalo es  $-32768 \sim 32767$ ; si bitDepth se establece igual a 18, el intervalo es  $-131072 \sim 131071$ . Por ejemplo, el valor del vector de movimiento derivado (por ejemplo los MV de cuatro subbloques 4x4 dentro de un bloque 8x8) se restringe de manera que la diferencia máxima entre partes enteras de los cuatro MV de subbloque 4x4 no es mayor que N muestras, tal como no mayor que 1 muestra.

La FIG. 4 es un diagrama esquemático de un dispositivo de codificado de vídeo 400 según una realización de la divulgación. El dispositivo de codificado de vídeo 400 es adecuado para implementar las realizaciones divulgadas como se describen en esta memoria. En una realización, el dispositivo de codificado de vídeo 400 puede ser un decodificador tal como el decodificador de vídeo 30 de la FIG. 1A o un codificador tal como el codificador de vídeo 20 de la FIG. 1A.

El dispositivo de codificado de vídeo 400 comprende puertos de entrada 410 (o puertos de entrada 410) y unidades de receptor (Rx) 420 para recibir datos; un procesador, unidad lógica, o unidad de procesamiento central (CPU) 430 para procesar los datos; unidades de transmisor (Tx) 440 y puertos de salida 450 (o puertos de salida 450) para transmitir los datos; y una memoria 460 para almacenar los datos. El dispositivo de codificado de vídeo 400 también puede comprender componentes óptico-a-eléctrico (OE) y componentes eléctrico-a-óptico (EO) acoplados a los puertos de entrada 410, las unidades de receptor 420, las unidades de transmisor 440, y los puertos de salida 450 para la salida o entrada de señales ópticas o eléctricas.

El procesador 430 se implementa por hardware y software. El procesador 430 puede implementarse como uno o más chips de CPU, núcleos (p. ej., un procesador multinúcleo), FPGA, ASIC y DSP. El procesador 430 está en comunicación con los puertos de entrada 410, las unidades de receptor 420, las unidades de transmisor



440, los puertos de salida 450 y la memoria 460. El procesador 430 comprende un módulo de codificado 470. El módulo de codificado 470 implementa las realizaciones divulgadas antes descritas. Por ejemplo, el módulo de codificado 470 implementa, procesa, prepara o proporciona las diversas operaciones de codificado. La inclusión del módulo de codificado 470 entonces proporciona una mejora sustancial a la funcionalidad del dispositivo de codificado de vídeo 400 y efectúa una transformación del dispositivo de codificado de vídeo 400 a un estado diferente. Alternativamente, el módulo de codificado 470 se implementa como instrucciones almacenadas en la memoria 460 y ejecutadas por el procesador 430.

La memoria 460 puede comprender uno o más discos, unidades de cinta, y unidades de estado sólido y puede utilizarse como un dispositivo de almacenamiento de datos de sobreflujo, para almacenar programas cuando dichos programas se seleccionan para ejecución, y almacenar instrucciones y datos que son leídos durante la ejecución de programa. La memoria 460 puede ser, por ejemplo, volátil y/o no volátil y puede ser una memoria de solo lectura (ROM), memoria de acceso aleatorio (RAM), memoria de contenido direccionable ternaria (TCAM), y/o memoria de acceso aleatorio estática (SRAM).

La Fig. 5 es un diagrama de bloques simplificado de un aparato 500 que puede utilizarse como cualquiera o ambos del dispositivo fuente 12 y el dispositivo de destino 14 de la Fig. 1A según una realización ejemplar.

Un procesador 502 en el aparato 500 puede ser una unidad de procesamiento central. Como alternativa, el procesador 502 puede ser cualquier otro tipo de dispositivo, o múltiples dispositivos, capaces de manipular o procesar información existente ahora o desarrollada más adelante. Aunque las implementaciones descritas se pueden poner en práctica con un único procesador como se muestra, p. ej., el procesador 502, se pueden lograr ventajas en velocidad y eficiencia usando más de un procesador.

Una memoria 504 en el aparato 500 puede ser un dispositivo de memoria de solo lectura (ROM) o un dispositivo de memoria de acceso aleatorio (RAM) en una implementación. Cualquier otro tipo conveniente de dispositivo de almacenamiento puede utilizarse como la memoria 504. La memoria 504 puede incluir un código y datos 506 a los que se tiene acceso a través del procesador 502 utilizando un bus 512. La memoria 504 puede además incluir un sistema operativo 508 y programas de aplicación 510, los programas de aplicación 510 incluyen al menos un programa que permita al procesador 502 realizar los métodos descritos en esta memoria. Por ejemplo los programas de aplicación 510 pueden incluir las aplicaciones 1 a N, que además incluyen una aplicación de codificado de vídeo que realiza los métodos descritos aquí.

El aparato 500 también puede incluir uno o más dispositivos de salida, tal como una pantalla 518. La pantalla 518 puede ser, en un ejemplo, una pantalla sensible al tacto que combina una pantalla con un elemento sensible al tacto que opera para detectar entradas táctiles. La pantalla 518 puede acoplarse al procesador 502 a través del bus 512.

Aunque aquí se representa como un solo bus, el bus 512 del aparato 500 puede componerse de múltiples buses. Además, el almacenamiento secundario 514 puede acoplarse directamente a los otros componentes del aparato 500 o se puede tener acceso al mismo a través de una red y puede comprender una sola unidad integrada tal como una tarjeta de memoria o múltiples unidades tal como múltiples tarjetas de memoria. El aparato 500 entonces puede implementarse en una gran variedad de configuraciones.

Lo siguiente describe primero conceptos en esta solicitud.

#### 1. Modo de interpredicción

En HEVC, se usan dos modos de interpredicción: un modo de predicción de vector de movimiento avanzada (AMVP) y un modo de fusión.

En el modo AMVP, bloques codificados espacialmente o temporalmente vecinos (denotados como bloques vecinos) de un bloque actual son primero atravesados; una lista de vectores de movimiento candidatos (que también se puede denominar lista de candidatos de información de movimiento) se construye con base en información de movimiento de los bloques vecinos; y después un vector de movimiento óptimo se determina de la lista de vectores de movimiento candidatos con base en un coste de tasa-distorsión. Información de movimiento de candidatos con un coste de tasa-distorsión mínimo se utiliza como predictor de vector de movimiento (MVP) del bloque actual. Ambas ubicaciones de los bloques vecinos y un orden transversal de los mismos se predefinen. El coste de tasa-distorsión se calcula según la fórmula (1), donde J representa el coste de tasa-distorsión (coste RD), SAD es una suma de diferencias absolutas (SAD) entre valores de muestra originales y valores de muestra predichos obtenidos a través de estimación de movimiento utilizando un predictor de vector de movimiento candidato, R representa una tasa de bits, y  $\lambda$  representa un multiplicador de Lagrange. Un lado del codificador transfiere un valor de índice del predictor de vector de movimiento seleccionado en la lista de vectores de movimiento candidatos y un valor de índice de fotograma de referencia a un lado del decodificador. Además, se realiza búsqueda de movimiento en las inmediaciones del MVP, para obtener un vector de movimiento real del bloque actual. El lado del codificador transfiere una diferencia

(diferencia de vector de movimiento) entre el MVP y el vector de movimiento real al lado del decodificador.

$$J = SAD + \lambda R \quad (1)$$

- 5 En el modo de fusión, primero se construye una lista de vectores de movimiento candidatos con base en información de movimiento de bloques codificados espacialmente o temporalmente vecinos de un bloque actual. Después se determina información de movimiento óptima a partir de la lista de vectores de movimiento candidatos como información de movimiento del bloque actual con base en un coste de tasa-distorsión. Un valor de índice (denotado como un índice de fusión en lo sucesivo) de la ubicación de información de movimiento óptima en la lista de vectores de movimiento candidatos se transfiere a un lado del decodificador. Información de movimiento de candidato espacial y temporal del bloque actual se muestra en la FIG. 6. La información de movimiento de candidato espacial es de cinco bloques espacialmente vecinos (A0, A1, B0, B1, y B2). Si un bloque vecino no está disponible (el bloque vecino no existe, o el bloque vecino no está codificado, o un modo de predicción utilizado para el bloque vecino no es un modo de interpredicción), información de movimiento de este bloque vecino no se añade a la lista de vectores de movimiento candidatos. La información de movimiento de candidato temporal del bloque actual se obtiene escalando un MV de un bloque en la ubicación correspondiente en un fotograma de referencia con base en recuentos de orden de imagen (POC) del fotograma de referencia y un fotograma actual. Primero se determina si un bloque en una ubicación T en el fotograma de referencia está disponible, y si el bloque no está disponible, se selecciona un bloque en una ubicación C.

Similar al modo AMVP, en el modo de fusión, se predefinen ambas ubicaciones de los bloques vecinos y un orden transversal del mismo también. Además, las ubicaciones de los bloques vecinos y el orden transversal de los mismos pueden ser diferentes en diferentes modos.

- 25 Se puede conocer que, una lista de vectores de movimiento candidatos (también referida como lista de candidatos, que puede denominarse lista de candidatos para abreviar) necesita ser mantenida en ambos del modo AMVP y el modo de fusión. Cada vez antes de añadir nueva información de movimiento a una lista de candidatos, primero se verifica si la misma información de movimiento ya existe en la lista. Si la misma información de movimiento ya existe, la información de movimiento no se añade a la lista. Este proceso de verificación se denomina recorte de la lista de vectores de movimiento candidatos. El recorte de la lista es para evitar que la misma información de movimiento sea incluida en la lista, y así evitar un cálculo redundante de coste de tasa-distorsión.

- 35 En interpredicción en HEVC, la misma información de movimiento se utiliza para todas las muestras en un bloque de codificado, y después se realiza compensación de movimiento con base en la información de movimiento, para obtener predictores de las muestras del bloque de codificado. En el bloque de codificado, no obstante, no todas las muestras tienen las mismas características de movimiento. El uso de la misma información de movimiento para el bloque de codificado puede tener como resultado predicción de compensación de movimiento imprecisa y más información residual.

- 45 En normas de codificado de vídeo existentes, se aplica estimación de movimiento de emparejamiento de bloque basada en un modelo de movimiento de traslación, y se asume que el movimiento de todas las muestras en un bloque es consistente. Sin embargo, en el mundo real, existe una variedad de movimiento. Muchos objetos están en movimiento sin traslación, por ejemplo, un objeto rotativo, una montaña rusa que gira en diferentes direcciones, una exhibición de fuegos artificiales, y algunas acrobacias en películas, especialmente un objeto en movimiento en un escenario de Contenido Generado por el Usuario (UGC). Para estos objetos en movimiento, si para codificación se utiliza una tecnología de compensación de movimiento de bloque basada en un modelo de movimiento de traslación en las normas de codificación existentes, la eficiencia de la codificación puede ser afectada en gran medida. Debido a esto, se introduce un modelo de movimiento sin traslación, por ejemplo, un modelo de movimiento afín, para mejorar adicionalmente la eficiencia de la codificación.

- 55 Con base en esto, debido a los diferentes modelos de movimiento, el modo AMVP puede clasificarse en un modo AMVP basado en modelo de traslación y un modo AMVP basado en modelo de no-traslación (por ejemplo, un modo AMVP basado en modelo afín), y el modo de fusión puede clasificarse en un modo de fusión basado en modelo de traslación y un modo de fusión basado en modelo de no-traslación (por ejemplo, un modo de fusión basado en modelo afín).

## 60 2. Modelo de movimiento de no-traslación

- 65 Predicción basada en un modelo de movimiento de no-traslación se refiere a que, se utiliza un mismo modelo de movimiento en el lado del codificador y del decodificador para derivar información de movimiento de cada subbloque (también referida como una unidad de compensación de submovimiento o una unidad de compensación de movimiento básica) en un bloque actual, y se realiza compensación de movimiento con base en la información de movimiento del subbloque para obtener un bloque de predicción, mejorando así la

eficiencia de la predicción. Modelos de movimiento de no-traslación comunes incluyen un modelo de movimiento afín de 4 parámetros y un modelo de movimiento afín de 6 parámetros.

- La unidad de compensación de submovimiento (también referida como un subbloque) en esta realización de esta solicitud puede ser una muestra o un bloque de muestras  $N_1 \times N_2$  obtenido con base en un método de división particular, donde ambos  $N_1$  y  $N_2$  son enteros positivos, y  $N_1$  puede ser igual a  $N_2$  o puede no ser igual a  $N_2$ .

El modelo de movimiento afín de 4 parámetros se expresa como la fórmula (2):

$$\begin{cases} vx = a_1 + a_3x + a_4y \\ vy = a_2 + a_4x + a_3y \end{cases} \quad (2)$$

- El modelo de movimiento afín de 4 parámetros puede representarse por vectores de movimiento de dos muestras y sus coordenadas con relación a una muestra superior-izquierda del bloque actual. Una muestra utilizada para representar un parámetro de modelo de movimiento se denomina un punto de control. Si la muestra superior-izquierda (0, 0) y una muestra superior-derecha (W, 0) se utilizan como puntos de control, primero se determinan vectores de movimiento respectivos ( $vx_0, vy_0$ ) y ( $vx_1, vy_1$ ) del punto de control superior-izquierdo y un punto de control superior-derecho del bloque actual. Entonces, se obtiene información de movimiento de cada unidad de compensación de submovimiento del bloque actual según la fórmula (3), donde (x, y) es una coordenada (tal como una coordenada de una muestra superior-izquierda) de una unidad de compensación de submovimiento con relación a la muestra superior-izquierda del bloque actual, y W representa una anchura del bloque actual. Se debería entender que otros puntos de control pueden utilizarse de manera alternativa. Por ejemplo, pueden utilizarse muestras en las ubicaciones (2, 2) y (W+2, 2), o (-2, -2) y (W-2, -2) como los puntos de control. La selección de los puntos de control no queda limitada por los ejemplos enumerados en esta memoria.

$$\begin{cases} vx = \frac{vx_1 - vx_0}{W}x - \frac{vy_1 - vy_0}{W}y + vx_0 \\ vy = \frac{vy_1 - vy_0}{W}x + \frac{vx_1 - vx_0}{W}y + vy_0 \end{cases} \quad (3)$$

El modelo de movimiento afín de 6 parámetros se expresa como la fórmula (4):

$$\begin{cases} vx = a_1 + a_3x + a_4y \\ vy = a_2 + a_5x + a_6y \end{cases} \quad (4)$$

- El modelo de movimiento afín de 6 parámetros puede representarse por vectores de movimiento de tres muestras y sus coordenadas con relación a una muestra superior-izquierda del bloque actual. Si la muestra superior-izquierda (0, 0), la muestra superior-derecha (W, 0), y la muestra inferior-izquierdo (0, H) del bloque actual se utilizan como puntos de control, primero se determinan vectores de movimiento respectivos ( $vx_0, vy_0$ ), ( $vx_1, vy_1$ ), y ( $vx_2, vy_2$ ) del punto de control superior-izquierdo, el punto de control superior-derecho, y el punto de control inferior-izquierdo del bloque actual. Luego, se obtiene información de movimiento de cada unidad de compensación de submovimiento del bloque actual según la fórmula (5), donde (x, y) es coordenadas de una unidad de compensación de submovimiento con relación a la muestra superior-izquierda del bloque actual, y W y H representan una anchura y una altura del bloque actual, respectivamente. Se debería entender que otros puntos de control pueden utilizarse de manera alternativa. Por ejemplo, muestras en las ubicaciones (2, 2), (W+2, 2), y (2, H+2), o (-2, -2), (W-2, -2), y (-2, H-2) pueden utilizarse como puntos de control. Estos ejemplos no son limitadores.

$$\begin{cases} vx = \frac{vx_1 - vx_0}{W}x + \frac{vx_2 - vx_0}{H}y + vx_0 \\ vy = \frac{vy_1 - vy_0}{W}x + \frac{vy_2 - vy_0}{H}y + vy_0 \end{cases} \quad (5)$$

Un bloque de codificado que se predice utilizando el modelo de movimiento afín se denomina bloque codificado afín.

- Generalmente, información de movimiento de un punto de control de un bloque codificado afín se puede obtener utilizando un modo de predicción de vector de movimiento avanzada (AMVP) basado en modelo de movimiento afín o un modo de fusión basado en modelo de movimiento afín.

- La información de movimiento del punto de control del bloque de codificado actual puede obtenerse usando un método de predicción de vector de movimiento de punto de control heredado o un método de predicción de vector de movimiento de punto de control construido.

### 3. Método de predicción de vector de movimiento de punto de control heredado

El método de predicción de vector de movimiento de punto de control heredado se refiere al uso de un modelo de movimiento de un bloque codificado afín vecino para determinar vectores de movimiento de punto de control candidatos de un bloque actual.

- 5 Un bloque actual mostrado en la FIG. 7 se utiliza como ejemplo. Bloques en ubicaciones vecinas alrededor del bloque actual son atravesados en un orden especificado, por ejemplo, A1->B1->B0->A0->B2, para encontrar un bloque codificado afín en el que se ubica un bloque en una ubicación vecina del bloque actual, y obtener información de movimiento de punto de control del bloque codificado afín. Además, un vector de movimiento de punto de control (para el modo de fusión) o un predictor de vector de movimiento de punto de control (para el modo AMVP) del bloque actual se deriva utilizando un modelo de movimiento construido con base en la información de movimiento de punto de control del bloque codificado afín. El orden A1->B1->B0->A0->B2 antes mencionado se utiliza simplemente como ejemplo y no debería ser interpretado como una limitación. También se puede utilizar otro orden. Además, los bloques en ubicaciones vecinas no se limitan a A1, B1, B0, A0, y B2 y se pueden utilizar diversos bloques en ubicaciones vecinas.

- 15 Un bloque en una ubicación vecina puede ser una muestra o un bloque de muestras de un tamaño preestablecido obtenido con base en un método de división particular. Por ejemplo, el bloque de muestras puede ser un bloque de muestras 4x4, un bloque de muestras 4x2, o un bloque de muestras de otro tamaño. Estos tamaños de bloque son para propósitos de ilustración y no deberían interpretarse como una limitación.

- 20 Lo siguiente describe un proceso de determinación utilizando A1 como un ejemplo, y un proceso similar puede emplearse para otros casos.

- 25 Como se muestra en la FIG. 7, si un bloque de codificado en el que se ubica A1 es un bloque codificado afín de 4 parámetros, se obtiene el vector de movimiento (vx4, vy4) de la muestra superior-izquierda (x4, y4) y el vector de movimiento (vx5, vy5) de la muestra superior-derecha (x5, y5) del bloque codificado afín. El vector de movimiento (vx0, vy0) de la muestra superior-izquierda (x0, y0) del bloque codificado afín actual se calcula según la fórmula (6), y el vector de movimiento (vx1, vy1) de la muestra superior-derecha (x1, y1) del bloque codificado afín actual se calcula según la fórmula (7).

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_0 - y_4) \\ vy_0 = vy_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_0 - y_4) \end{cases} \quad (6)$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_1 - y_4) \\ vy_1 = vy_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_1 - y_4) \end{cases} \quad (7)$$

- 35 Una combinación del vector de movimiento (vx0, vy0) de la muestra superior-izquierda (x0, y0) y el vector de movimiento (vx1, vy1) de la muestra superior-derecha (x1, y1) del bloque actual que se obtienen con base en el bloque codificado afín en el que se ubica A1 es los vectores de movimiento de punto de control candidatos del bloque actual.

- 40 Si un bloque de codificado en el que se ubica A1 es un bloque codificado afín de 6 parámetros, se obtiene el vector de movimiento (vx4, vy4) de la muestra superior-izquierda (x4, y4), el vector de movimiento (vx5, vy5) de la muestra superior-derecha (x5, y5), y el vector de movimiento (vx6, vy6) de la muestra inferior-izquierda (x6, y6) del bloque codificado afín. El vector de movimiento (vx0, vy0) de la muestra superior-izquierda (x0, y0) del bloque actual se calcula según la fórmula (8). El vector de movimiento (vx1, vy1) de la muestra superior-derecha (x1, y1) del bloque actual se calcula según la fórmula (9). El vector de movimiento (vx2, vy2) de la muestra inferior-izquierda (x2, y2) del bloque actual se calcula según la fórmula (10).

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_6 - vx_4)}{y_6 - y_4} \times (y_0 - y_4) \\ vy_0 = vy_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vy_6 - vy_4)}{y_6 - y_4} \times (y_0 - y_4) \end{cases} \quad (8)$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_6 - vx_4)}{y_6 - y_4} \times (y_1 - y_4) \\ vy_1 = vy_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vy_6 - vy_4)}{y_6 - y_4} \times (y_1 - y_4) \end{cases} \quad (9)$$

$$\begin{cases} vx_2 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_2 - x_4) + \frac{(vx_6 - vx_4)}{y_6 - y_4} \times (y_2 - y_4) \\ vy_2 = vy_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_2 - x_4) + \frac{(vy_6 - vy_4)}{y_6 - y_4} \times (y_2 - y_4) \end{cases} \quad (10)$$

Una combinación del vector de movimiento (vx0, vy0) de la muestra superior-izquierda (x0, y0), el vector de movimiento (vx1, vy1) de la muestra superior-derecha (x1, y1), y el vector de movimiento (vx2, vy2) de la muestra inferior-izquierda (x2, y2) del bloque actual que se obtienen con base en el bloque codificado afin en el que se ubica A1 es el vector de movimiento de punto de control candidato del bloque actual.

5

Se debería observar que otros modelos de movimiento, ubicaciones de candidatos, y órdenes de búsqueda y transversales también son aplicables a esta solicitud. No se describen detalles en esta realización de esta solicitud.

10 Se debería observar que métodos en los que se utilizan otros puntos de control para representar modelos de movimiento de bloques de codificado actuales y vecinos también son aplicables a esta solicitud. En esta memoria no se no se describen detalles.

#### 4. Método de predicción de vectores de movimiento de punto de control contruidos 1

15

El método de predicción de vector de movimiento de punto de control construido se refiere a una combinación de vectores de movimiento de bloques codificados vecinos alrededor de un punto de control de un bloque actual como un vector de movimiento de punto de control de un bloque codificado afin actual, sin considerar si los bloques codificados vecinos son bloques codificados afin.

20

Vectores de movimiento de la muestra superior-izquierda y la muestra superior-derecha del bloque actual se determinan utilizando información de movimiento de los bloques codificados vecinos alrededor del bloque de codificado actual. La FIG. 8A se utiliza como ejemplo para describir el método de predicción de vector de movimiento de punto de control construido. Se debería observar que la FIG. 8A es simplemente un ejemplo y no debería interpretarse como una limitación.

25

Como se muestra en la FIG. 8A, los vectores de movimiento de bloques codificados vecinos A2, B2, y B3 de la muestra superior-izquierda se utilizan como vectores de movimiento candidatos para el vector de movimiento de la muestra superior-izquierda del bloque actual, y vectores de movimiento de bloques codificados vecinos B1 y B0 de la muestra superior-derecha se utilizan como vectores de movimiento candidatos para el vector de movimiento de la muestra superior-derecha del bloque actual. Los vectores de movimiento candidatos de la muestra superior-izquierda y la muestra superior-derecha son combinados para constituir una pluralidad de tuplas de 2. Vectores de movimiento de dos bloques codificados incluidos en una tupla de 2 pueden utilizarse como vectores de movimiento de punto de control candidatos del bloque actual, como se muestra en la siguiente fórmula (11A):

35

$$\{v_{A2}, v_{B1}\}, \{v_{A2}, v_{B0}\}, \{v_{B2}, v_{B1}\}, \{v_{B2}, v_{B0}\}, \{v_{B3}, v_{B1}\}, \{v_{B3}, v_{B0}\} \quad (11A)$$

donde  $v_{A2}$  representa un vector de movimiento de A2,  $v_{B1}$  representa un vector de movimiento de B1,  $v_{B0}$  representa un vector de movimiento de B0,  $v_{B2}$  representa un vector de movimiento de B2, y  $v_{B3}$  representa un vector de movimiento de B3.

40

Tal como se muestra en la FIG. 8A, vectores de movimiento de bloques codificados vecinos A2, B2, y B3 de la muestra superior-izquierda se utilizan como vectores de movimiento candidatos para el vector de movimiento de la muestra superior-izquierda del bloque actual, vectores de movimiento de bloques codificados vecinos B1 y B0 de la muestra superior-derecha se utilizan como vectores de movimiento candidatos para el vector de movimiento de la muestra superior-derecha del bloque actual, y vectores de movimiento de bloques codificados vecinos A0 y A1 de la muestra inferior-izquierda se utilizan como vectores de movimiento candidatos para el vector de movimiento de la muestra inferior-izquierda del bloque actual. Los vectores de movimiento candidatos de la muestra superior-izquierda, la muestra superior-derecha, y la muestra inferior-izquierda son combinados para constituir una tupla de 3. Vectores de movimiento de tres bloques codificados incluidos en una tupla de 3 pueden utilizarse como vectores de movimiento de punto de control candidatos del bloque actual, como se muestra en las siguientes fórmulas (11B) y (11C):

50

$$\{v_{A2}, v_{B1}, v_{A0}\}, \{v_{A2}, v_{B0}, v_{A0}\}, \{v_{B2}, v_{B1}, v_{A0}\}, \{v_{B2}, v_{B0}, v_{A0}\}, \{v_{B3}, v_{B1}, v_{A0}\}, \{v_{B3}, v_{B0}, v_{A0}\} \quad (11B)$$

55

$$\{v_{A2}, v_{B1}, v_{A1}\}, \{v_{A2}, v_{B0}, v_{A1}\}, \{v_{B2}, v_{B1}, v_{A1}\}, \{v_{B2}, v_{B0}, v_{A1}\}, \{v_{B3}, v_{B1}, v_{A1}\}, \{v_{B3}, v_{B0}, v_{A1}\} \quad (11C)$$

donde  $v_{A2}$  representa un vector de movimiento de A2,  $v_{B1}$  representa un vector de movimiento de B1,  $v_{B0}$  representa un vector de movimiento de B0,  $v_{B2}$  representa un vector de movimiento de B2,  $v_{B3}$  representa un vector de movimiento de B3,  $v_{A0}$  representa un vector de movimiento de A0, y  $v_{A1}$  representa un vector de movimiento de A1.

60

Se debería observar que otros métodos para combinar vectores de movimiento de punto de control también son aplicables a esta solicitud. En esta memoria no se no se describen detalles.

65

Se debería observar que métodos en los que se utilizan otros puntos de control para representar modelos de movimiento de bloques de codificado actuales y vecinos también son aplicables a esta solicitud. En esta memoria no se describen detalles.

5

5. Método de predicción de vectores de movimiento de punto de control contruidos 2, tal como se muestra en la FIG. 8B:

Etapa 501: Obtener información de movimiento de puntos de control de un bloque actual.

10

Por ejemplo, en la FIG. 8A,  $CP_k$  ( $k=1, 2, 3, 4$ ) representa un  $k$ -ésimo punto de control.  $A0, A1, A2, B0, B1, B2$ , y  $B3$  son ubicaciones espacialmente vecinas del bloque actual y se utilizan para predecir  $CP1, CP2$ , o  $CP3$ , y  $T$  es una ubicación temporalmente vecina del bloque actual y se utiliza para predecir  $CP4$ .

15

Se asume que coordenadas de  $CP1, CP2, CP3$ , y  $CP4$  son  $(0, 0)$ ,  $(W, 0)$ ,  $(H, 0)$ , y  $(W, H)$ , respectivamente, donde  $W$  y  $H$  representan una anchura y una altura del bloque actual.

Para cada punto de control, información de movimiento del mismo se obtiene en el siguiente orden:

20

(1) Para  $CP1$ , un orden de comprobación es  $B2 \rightarrow A2 \rightarrow B3$ . Si  $B2$  está disponible, información de movimiento de  $B2$  se utiliza para  $CP1$ . De otra manera,  $A2$  y  $B3$  se comprueban en secuencia. Si información de movimiento de todas las tres ubicaciones no está disponible, no puede ser obtenerse información de movimiento de  $CP1$ .

25

(2) Para  $CP2$ , un orden de comprobación es  $B0 \rightarrow B1$ . Si  $B0$  está disponible, información de movimiento de  $B0$  se utiliza para  $CP2$ . De lo contrario, se comprueba  $B1$ . Si no está disponible información de movimiento de ambas de las ubicaciones, no se puede obtener información de movimiento de  $CP2$ .

30

(3) Para  $CP3$ , un orden de comprobación es  $A0 \rightarrow A1$ . Si  $A0$  está disponible, información de movimiento de  $A0$  se utiliza para  $CP3$ . De otra manera, se comprueba  $A1$ . Si información de movimiento de ambas ubicaciones no está disponible, no se puede obtener información de movimiento de  $CP3$ .

(4) Para  $CP4$ , se utiliza información de movimiento de  $T$ .

35

En esta memoria, que  $X$  esté disponible significa que el bloque  $X$  (por ejemplo,  $A0, A1, A2, B0, B1, B2, B3$ , o  $T$ ) ya está codificado y se utiliza un modo de interpredicción. De otra manera,  $X$  no está disponible.

Se debería observar que otros métodos para obtener información de movimiento de un punto de control también son aplicables a esta solicitud. En esta memoria no se describen detalles.

40

Etapa 502: Combinar la información de movimiento de los puntos de control, para obtener información de movimiento de punto de control construido.

45

Información de movimiento de dos puntos de control se combina para constituir una tupla de 2, para construir un modelo de movimiento afín de 4 parámetros. Combinaciones de información de movimiento de los dos puntos de control pueden ser  $\{CP1, CP4\}$ ,  $\{CP2, CP3\}$ ,  $\{CP1, CP2\}$ ,  $\{CP2, CP4\}$ ,  $\{CP1, CP3\}$ , y  $\{CP3, CP4\}$ . Por ejemplo, un modelo de movimiento afín de 4 parámetros construido mediante el uso de una tupla de 2 incluyendo información de movimiento de los puntos de control  $CP1$  y  $CP2$  puede denotarse como Afín ( $CP1, CP2$ ).

50

Información de movimiento de tres puntos de control se combina para constituir una tupla de 3, para construir un modelo de movimiento afín de 6 parámetros. Combinaciones de información de movimiento de los tres puntos de control pueden ser  $\{CP1, CP2, CP4\}$ ,  $\{CP1, CP2, CP3\}$ ,  $\{CP2, CP3, CP4\}$ , y  $\{CP1, CP3, CP4\}$ . Por ejemplo, un modelo de movimiento afín de 6 parámetros construido mediante el uso de una tupla de 3 incluyendo información de movimiento de los puntos de control  $CP1, CP2$ , y  $CP3$  puede denotarse como Afín ( $CP1, CP2, CP3$ ).

55

Información de movimiento de cuatro puntos de control se combina para constituir una tupla de 4, para construir un modelo de movimiento bilineal de 8 parámetros. Un modelo de movimiento bilineal de 8 parámetros construido mediante el uso de una tupla de 4 incluyendo información de movimiento de los puntos de control  $CP1, CP2, CP3$ , y  $CP4$  puede denotarse como Bilineal ( $CP1, CP2, CP3, CP4$ ).

60

En esta realización de esta solicitud, para facilidad de descripción, una combinación de información de movimiento de dos puntos de control (o dos bloques codificados) se denomina simplemente tupla de 2, una combinación de información de movimiento de tres puntos de control (o tres bloques codificados) se denomina simplemente tupla de 3, y una combinación de información de movimiento de cuatro puntos de control (o cuatro bloques codificados) se denomina simplemente tupla de 4.

65

Estos modelos son atravesados en un orden preestablecido. Si información de movimiento de un punto de control correspondiente a un modelo de combinación no está disponible, se considera que el modelo no está disponible. De otra manera, se determina un índice de fotograma de referencia del modelo, y se escala un vector de movimiento de punto de control. Si información de movimiento de todos los puntos de control después del escalado es consistente, el modelo es inválido. Si toda la información de movimiento de puntos de control que controlan el modelo está disponible y el modelo es válido, información de movimiento de los puntos de control que construyen el modelo se añade a una lista de candidatos de información de movimiento.

Un método de escalado de vector de movimiento de punto de control se muestra en la fórmula (12):

$$\mathbf{MV}_s = \frac{\text{CurPoc} - \text{DesPoc}}{\text{CurPoc} - \text{SrcPoc}} \times \mathbf{MV} \quad (12)$$

donde CurPoc representa el número POC de un fotograma actual, DesPoc representa el número POC de un fotograma de referencia del bloque actual, SrcPoc representa el número POC de un fotograma de referencia de un punto de control, MV representa el vector de movimiento obtenido después del escalado, y MV representa el vector de movimiento de un punto de control.

Se debería observar que una combinación de diferentes puntos de control puede convertirse en puntos de control en una misma ubicación.

Por ejemplo, un modelo de movimiento afín de 4 parámetros obtenido a través de una combinación {CP1, CP4}, {CP2, CP3}, {CP2, CP4}, {CP1, CP3}, o {CP3, CP4} se convierte en una representación por {CP1, CP2} o {CP1, CP2, CP3}. El método de conversión incluye: sustituir los vectores de movimiento y la información de coordenadas de los puntos de control {CP1, CP4}, {CP2, CP3}, {CP2, CP4}, {CP1, CP3}, o {CP3, CP4} en la fórmula (2) para obtener los parámetros de modelo; y después sustituir la información de coordenadas de {CP1, CP2} en la fórmula (3), para obtener un vector de movimiento de los puntos de control {CP1, CP2}.

De manera más directa, la conversión puede realizarse según las siguientes fórmulas (13) a (21), donde W representa la anchura del bloque actual, y H representa la altura del bloque actual. En las fórmulas (13) a (21),  $(vx_0, vy_0)$  representa un vector de movimiento de CP1,  $(vx_1, vy_1)$  representa un vector de movimiento de CP2,  $(vx_2, vy_2)$  representa un vector de movimiento de CP3, y  $(vx_3, vy_3)$  representa un vector de movimiento de CP4.

{CP1, CP2} puede convertirse en {CP1, CP2, CP3} utilizando la siguiente fórmula (13). En otras palabras, el vector de movimiento de CP3 en {CP1, CP2, CP3} puede determinarse utilizando la fórmula (13):

$$\begin{cases} vx_2 = -\frac{vy_1 - vy_0}{W} H + vx_0 \\ vy_2 = +\frac{vx_1 - vx_0}{W} H + vy_0 \end{cases} \quad (13)$$

{CP1, CP3} puede convertirse en {CP1, CP2} o {CP1, CP2, CP3} utilizando la siguiente fórmula (14):

$$\begin{cases} vx_1 = +\frac{vy_2 - vy_0}{H} W + vx_0 \\ vy_1 = -\frac{vx_2 - vx_0}{H} W + vy_0 \end{cases} \quad (14)$$

{CP2, CP3} puede convertirse en {CP1, CP2} o {CP1, CP2, CP3} utilizando la siguiente fórmula (15):

$$\begin{cases} vx_0 = \frac{vx_2 - vx_1}{W * W + H * H} W * W - \frac{vy_2 - vy_1}{W * W + H * H} H * W + vx_1 \\ vy_0 = \frac{vy_2 - vy_1}{W * W + H * H} W * W + \frac{vx_2 - vx_1}{W * W + H * H} H * W + vy_1 \end{cases} \quad (15)$$

{CP1, CP4} puede convertirse en {CP1, CP2} o {CP1, CP2, CP3} utilizando la siguiente fórmula (16) o (17):

$$\begin{cases} vx_1 = \frac{vx_3 - vx_0}{W * W + H * H} W * W + \frac{vy_3 - vy_0}{W * W + H * H} H * W + vx_0 \\ vy_1 = \frac{vy_3 - vy_0}{W * W + H * H} W * W - \frac{vx_3 - vx_0}{W * W + H * H} H * W + vy_0 \end{cases} \quad (16)$$

$$\begin{cases} vx_2 = \frac{vx_3 - vx_0}{W * W + H * H} H * H - \frac{vy_3 - vy_0}{W * W + H * H} H * W + vx_0 \\ vy_2 = \frac{vy_3 - vy_0}{W * W + H * H} W * H + \frac{vx_3 - vx_0}{W * W + H * H} H * H + vy_0 \end{cases} \quad (17)$$

{CP2, CP4} puede convertirse en {CP1, CP2} utilizando la siguiente fórmula (18), y {CP2, CP4} puede convertirse en {CP1, CP2, CP3} utilizando las siguientes fórmulas (18) y (19):

$$\begin{cases} vx_0 = -\frac{vy_3 - vy_1}{H} W + vx_1 \\ vy_0 = +\frac{vx_3 - vx_1}{H} W + vy_1 \end{cases} \quad (18)$$

$$\begin{cases} vx_2 = -\frac{vy_3 - vy_1}{H} W + vx_3 \\ vy_2 = +\frac{vx_3 - vx_1}{H} W + vy_3 \end{cases} \quad (19)$$

- 5 {CP3, CP4} puede convertirse en {CP1, CP2} utilizando la siguiente fórmula (20), y {CP3, CP4} puede convertirse en {CP1, CP2, CP3} utilizando las siguientes fórmulas (20) y (21):

$$\begin{cases} vx_0 = +\frac{vy_3 - vy_2}{W} H + vx_2 \\ vy_0 = -\frac{vx_3 - vx_2}{W} H + vy_2 \end{cases} \quad (20)$$

$$\begin{cases} vx_1 = +\frac{vy_3 - vy_2}{W} H + vx_3 \\ vy_1 = -\frac{vx_3 - vx_2}{W} H + vy_3 \end{cases} \quad (21)$$

10

Por ejemplo, un modelo de movimiento afín de 6 parámetros obtenido a través de una combinación {CP1, CP2, CP4}, {CP2, CP3, CP4}, o {CP1, CP3, CP4} puede convertirse en una representación por {CP1, CP2, CP3}. El método de conversión incluye: sustituir un vector de movimiento e información de coordenadas de los puntos de control {CP1, CP2, CP4}, {CP2, CP3, CP4}, o {CP1, CP3, CP4} en la fórmula (4), para obtener un parámetro de modelo; y después sustituir información de coordenadas de {CP1, CP2, CP3} en la fórmula (5), para obtener los vectores de movimiento de {CP1, CP2, CP3}.

15

De manera más directa, la conversión puede realizarse según las siguientes fórmulas (22) a (24), donde W representa la anchura del bloque actual, y H representa la altura del bloque actual. En las fórmulas (13) a (21), (vx<sub>0</sub>, vy<sub>0</sub>) representa un vector de movimiento de CP1, (vx<sub>1</sub>, vy<sub>1</sub>) representa un vector de movimiento de CP2, (vx<sub>2</sub>, vy<sub>2</sub>) representa un vector de movimiento de CP3, y (vx<sub>3</sub>, vy<sub>3</sub>) representa un vector de movimiento de CP4.

20

{CP1, CP2, CP4} puede convertirse en {CP1, CP2, CP3} utilizando la siguiente fórmula (22):

25

$$\begin{cases} vx_2 = vx_3 + vx_0 - vx_1 \\ vy_2 = vy_3 + vy_0 - vy_1 \end{cases} \quad (22)$$

{CP2, CP3, CP4} puede convertirse en {CP1, CP2, CP3} utilizando la siguiente fórmula (23):

30

$$\begin{cases} vx_0 = vx_1 + vx_2 - vx_3 \\ vy_0 = vy_1 + vy_2 - vy_3 \end{cases} \quad (23)$$

{CP1, CP3, CP4} puede convertirse en {CP1, CP2, CP3} utilizando la siguiente fórmula (24):

35

$$\begin{cases} vx_1 = vx_3 + vx_0 - vx_2 \\ vy_1 = vy_3 + vy_0 - vy_2 \end{cases} \quad (24)$$

6. Modo de predicción de vector de movimiento avanzada basado en modelo de movimiento afín (modo AMVP Afín)

(1) Construir una lista de vectores de movimiento candidatos

40

Una lista de vectores de movimiento candidatos para el modo AMVP basado en modelo de movimiento afín se construye usando el método de predicción de vector de movimiento de punto de control heredado y/o el método de predicción de vector de movimiento de punto de control construido descrito anteriormente. En esta realización de esta solicitud, la lista de vectores de movimiento candidatos para el modo AMVP basado en modelo de movimiento afín puede denominarse lista de candidatos de predictor de vector de movimiento de punto de control. El predictor de vector de movimiento de cada punto de control incluye vectores de movimiento de dos puntos de control (modelo de movimiento afín de 4 parámetros) o vectores de movimiento de tres puntos de control (modelo de movimiento afín de 6 parámetros).

45

Opcionalmente, la lista de candidatos de predictor de vector de movimiento de punto de control se recorta y clasifica según una regla particular, y puede truncarse o rellenarse para incluir una cantidad particular de candidatos de predictor de vector de movimiento de punto de control.

50

(2) Determinar un candidato de predictor de vector de movimiento de punto de control óptimo



En un lado del codificador, el vector de movimiento de cada unidad de compensación de submovimiento en el bloque de codificado actual se obtiene con base en cada uno de los candidatos de predictor de vector de movimiento de punto de control (por ejemplo, un candidato de X tuplas) en la lista de candidatos de predictor de vector de movimiento de punto de control utilizando la fórmula (3) o (5). El vector de movimiento obtenido puede utilizarse para obtener un valor de muestra en una ubicación correspondiente en un fotograma de referencia al que apunta el vector de movimiento de la unidad de compensación de submovimiento. Este valor de muestra se utiliza como un predictor para realizar compensación de movimiento utilizando el modelo de movimiento afín. Se calcula una diferencia promedio entre un valor original y un valor de predicción de cada muestra del bloque de codificado actual. Un predictor de vector de movimiento de punto de control candidato correspondiente a la diferencia promedio mínima se selecciona como el predictor óptimo de vector de movimiento de punto de control candidato, y se usan predictores de vector de movimiento de dos o tres puntos de control del bloque de codificado actual. Un número de índice que representa la ubicación del candidato de predictor de vector de movimiento de punto de control óptimo (por ejemplo, un candidato de X tuplas) en la lista de candidatos de predictor de vector de movimiento de punto de control se codifica en un flujo de bits y enviado a un decodificador.

En un lado del decodificador, el número de índice se analiza, y los predictores de vector de movimiento de punto de control (CPMVP) (por ejemplo candidato de X tuplas) se determinan de la lista de candidatos de predictor de vector de movimiento de punto de control con base en el número de índice.

#### (3) Determinar vectores de movimiento de punto de control

En el lado del codificador, el predictor de vector de movimiento de punto de control se utiliza como punto de inicio de búsqueda para búsqueda de movimiento dentro de un intervalo de búsqueda específico, para obtener vectores de movimiento de punto de control (CPMV). Las diferencias (diferencias de vectores de movimiento de punto de control, CPMVD) entre los vectores de movimiento de punto de control respectivos y los predictores de vector de movimiento de punto de control son transferidas al lado del decodificador.

En el lado del decodificador, las diferencias de vector de movimiento de punto de control se obtienen analizando el flujo de bits y se añaden a los predictores de vector de movimiento de punto de control respectivamente, para obtener los vectores de movimiento de punto de control respectivos.

#### 7. Modo de fusión afín

Una lista de candidatos de fusión de vectores de movimiento de punto de control se construye utilizando el método de predicción de vector de movimiento de punto de control heredado y/o el método de predicción de vector de movimiento de punto de control construido descrito anteriormente.

Opcionalmente, la lista de candidatos de fusión de vectores de movimiento de punto de control se recorta y clasifica según una regla particular, y puede ser truncada o rellenada a una cantidad particular.

En el lado del codificador, un vector de movimiento de cada unidad de compensación de submovimiento (una muestra o un bloque de muestra  $N_1 \times N_2$  obtenido con base en un método de división particular) en el bloque de codificado actual se obtiene con base en cada candidato de vectores de movimiento de punto de control (por ejemplo Candidato de X tuplas) en la lista de candidatos de fusión utilizando la fórmula ((3) o (5). El vector de movimiento obtenido puede utilizarse para obtener valores de muestra en una ubicación en un fotograma de referencia al que apunta el vector de movimiento de cada unidad de compensación de submovimiento. Estos valores de muestra se utilizan como valores de muestra predichos para realizar compensación de movimiento afín. Se calcula una diferencia promedio entre un valor original y un valor predicho de cada muestra del bloque de codificado actual. El candidato de vectores de movimiento de punto de control (CPMV) (por ejemplo, candidato de 2 tuplas o candidato de 3 tuplas) correspondiente a una diferencia promedio mínima se selecciona como vectores de movimiento de dos o tres puntos de control del bloque de codificado actual. Un número de índice que representa la ubicación del vector de movimiento de punto de control en la lista de candidatos se codifica en el flujo de bits del vídeo y se envía al decodificador.

En el lado de decodificador, el número de índice se analiza, y los vectores de movimiento de punto de control (CPMV) se determinan de la lista de candidatos de fusión de vector de movimiento de punto de control con base en el número de índice.

Además, se debería observar que en esta solicitud, "al menos uno" significa uno o más, y "una pluralidad de" significa al menos dos. El término "y/o" describe una relación de asociación para describir objetos asociados y representa que pueden existir tres relaciones. Por ejemplo, A y/o B pueden representar los siguientes casos: Solamente existe A, existen A y B, y solamente existe B, donde A y B pueden estar en forma singular o plural. El carácter "/" generalmente indica una relación "o" entre los objetos asociados. "Al menos uno (una pieza) de los siguientes [elementos]" o una expresión similar se refieren a cualquier combinación de estos elementos,

incluyendo cualquier combinación de elementos singulares (piezas) o elementos plurales (piezas). Por ejemplo, al menos uno (una pieza) de a, b, o c puede representar: a, b, c, a y b, a y c, b y c, o a y b y c, donde a, b, y c pueden ser singular o plural.

- 5 En esta solicitud, cuando el modo de interpredicción se utiliza para decodificar el bloque actual, se puede utilizar un elemento de sintaxis para señalar el modo de interpredicción.

Para algunas estructuras de sintaxis actualmente utilizadas del modo de interpredicción utilizado para analizar el bloque actual, consúltese la Tabla 1 donde se enumeran algunas sintaxis para el modo de interpredicción.

- 10 Un elemento de sintaxis en una estructura de sintaxis alternativamente puede representarse por otros identificadores.

Tabla 1

coding_unit( x0, y0, cbWidth, cbHeight, treeType) {	Descriptor
if( cu_skip_flag[ x0 ][ y0 ] = 0 )	
merge_flag[ x0 ][ y0 ]	ae(v)
if( merge_flag[ x0 ][ y0 ] ) {	
merge_data(x0, y0, cbWidth, cbHeight)	
} else if ( CuPredMode[ x0 ][ y0 ] == MODE_IBC ) {	
mvd_coding( x0, y0, 0, 0 )	
mvp_10_flag[ x0 ][ y0 ]	ae(v)
if(sps_amvr_enabled_flag && (MvdLO[ x0 ][ y0 ][ 0 ] != 0    MvdLO[ x0 ][ y0 ][ 1 ] != 0 ) ) {	
amvr_precision_flag[ x0 ][ y0 ]	ae(v)
}	
} else {	
if( tile_group_type == B )	
inter_pred_idc[ x0 ][ y0 ]	ae(v)
if( sps_affine_enabled_flag && cbWidth >= 16 && cbHeight >= 16 ) {	
inter_affine_flag[ x0 ][ y0 ]	ae(v)
if(sps_affine_type_flag && inter_affine_flag[ x0 ][ y0 ] )	
cu_affine_type_flag[ x0 ][ y0 ]	ae(v)
}	
if(inter_pred_idc[ x0 ][ y0 ] == PRED_BI && !inter_affine_flag[ x0 ][ y0 ] && RefIdxSymL0 > -1 && RefIdxSymL1 > -1 )	
sym_mvd_flag[ x0 ][ y0 ]	ae(v)
if(inter_pred_idc[ x0 ][ y0 ] != PRED_L1 ) {	
if(NumRefIdxActive[ 0 ] > 1 && ! sym_mvd_flag[ x0 ][ y0 ] )	
ref_idx_10[ x0 ][ y0 ]	ae(v)
mvd_coding( x0, y0, 0, 0 )	
if(MotionModelIdc[ x0 ][ y0 ] > 0 )	
mvd_coding( x0, y0, 0, 1 )	
if(MotionModelIdc[ x0 ][ y0 ] > 1 )	
mvd_coding( x0, y0, 0, 2 )	
mvp_10_flag[ x0 ][ y0 ]	ae(v)
} else {	

MvdL0[ x0 ][ y0 ][ 0 ] = 0	
MvdL0[ x0 ][ y0 ][ 1 ] = 0	
}	
if(inter_pred_idc[x0][y0] != PRED_L0){	
if( NumRefIdxActive[ 1 ] > 1 && !sym_mvd_flag[ x0 ][ y0 ] )	
ref_idx_l1[ x0 ][ y0 ]	ae(v)
if(mvd_11_zero_flag && inter_pred_idc[ x0 ][ y0 ] == PRED_BI ) {	
MvdL1[ x0 ][ y0 ][ 0 ] = 0	
MvdL1[ x0 ][ y0 ][ 1 ] = 0	
MvdCpL1[ x0 ][ y0 ][ 0 ][ 0 ] = 0	
MvdCpL1[ x0 ][ y0 ][ 0 ][ 1 ] = 0	
MvdCpL1[ x0 ][ y0 ][ 1 ][ 0 ] = 0	
MvdCpL1[ x0 ][ y0 ][ 1 ][ 1 ] = 0	
MvdCpL1[ x0 ][ y0 ][ 2 ][ 0 ] = 0	
MvdCpL1[ x0 ][ y0 ][ 2 ][ 1 ] = 0	
} else {	
if(sym_mvd_flag[ x0 ][ y0 ] ) {	
MvdL1[ x0 ][ y0 ][ 0 ] = -MvdL0 [ x0 ][ y0 ][ 0 ]	
MvdL1[ x0 ][ y0 ][ 1 ] = -MvdL0[ x0 ][ y0 ][ 1 ]	
} else	
mvd_coding(x0, y0, 1, 0 )	
if(MotionModelIdc[ x0 ][ y0 ] > 0 )	
mvd_coding(x0, y0, 1, 1)	
if(MotionModelIdc[ x0 ][ y0 ] > 1 )	
mvd_coding(x0, y0, 1, 2 )	
mvp_l1_flag[ x0 ][ y0 ]	ae(v)
}	
} else {	
MvdL1[ x0 ][ y0 ][ 0 ] = 0	
MvdL1[ x0 ][ y0 ][ 1 ] = 0	
}	
.....	
}	
}	
.....	
}	
}	

- En la Tabla 1, inter\_affine\_flag[ x0 ][ y0 ] igual a 1 especifica que para la unidad de codificación actual, cuando se decodifica un grupo de P o B mosaicos, se utiliza compensación de movimiento basada en modelo afín para generar las muestras de predicción de la unidad de codificación actual. inter\_affine\_flag[ x0 ][ y0 ] igual a 0 especifica que la unidad de codificación no se predice por compensación de movimiento basada en modelo afín. Cuando inter\_affine\_flag[ x0 ][ y0 ] no está presente, se infiere que es igual a 0.
- 5

inter\_pred\_idc[ x0 ][ y0 ] especifica si lista0, lista1, o se utiliza bi-predicción para la unidad de codificación actual según la Tabla 2. Los índices de distribución x0, y0 especifican la ubicación (x0, y0) de la muestra de luminancia superior-izquierda del bloque de codificado considerado con relación a la muestra de luminancia superior-izquierda de la imagen.

5

Cuando inter\_pred\_idc[ x0 ][ y0 ] no está presente, se infiere que es igual a PRED\_L0.

Tabla 2

inter_pred_idc	Name of inter_pred_idc	
	( cbWidth + cbHeight ) != 8	( cbWidth + cbHeight ) = 8
0	PRED_L0	PRED_L0
1	PRED_L1	PRED_L1
2	PRED_BI	n.d.

- 10 sps\_afin\_enabled\_flag especifica si puede utilizarse compensación de movimiento basada en modelo afín para interpredicción. Si sps\_afin\_enabled\_flag es igual a 0, la sintaxis debería estar restringida de manera que ninguna compensación de movimiento basada en modelo afín se utiliza en el CVS, e inter\_affine\_flag y cu\_affine\_type\_flag no están presentes en la sintaxis de unidad de codificación del CVS. De otra manera  
15 (sps\_afin\_enabled\_flag es igual a 1), puede utilizarse compensación de movimiento basada en modelo afín en el CVS.

Un elemento de sintaxis inter\_affine\_flag[x0][y0] (o affine\_inter\_flag[x0][y0]) puede utilizarse para indicar si el modo AMVP basado en modelo de movimiento afín se utiliza para el bloque actual cuando el sector en el que se ubica el bloque actual es un sector de tipo P o un sector de tipo B. Cuando este elemento de sintaxis no  
20 aparece en el flujo de bits, el elemento de sintaxis es 0 por omisión. Por ejemplo, inter\_affine\_flag[x0][y0] = 1 indica que el modo AMVP basado en modelo de movimiento afín se utiliza para el bloque actual; e inter\_affine\_flag[x0][y0] = 0 indica que el modo AMVP basado en modelo de movimiento afín no se utiliza para el bloque actual, y se puede utilizar un modo AMVP basado en modelo de movimiento de traslación. Es decir, inter\_affine\_flag[ x0 ][ y0 ] igual a 1 especifica que para la unidad de codificación actual, cuando se decodifica  
25 un grupo de P o B mosaicos, se utiliza compensación de movimiento basada en modelo afín para generar las muestras de predicción de la unidad de codificación actual. inter\_affine\_flag[ x0 ][ y0 ] igual a 0 especifica que la unidad de codificación no se predice por compensación de movimiento basada en modelo afín. Cuando inter\_affine\_flag[ x0 ][ y0 ] no está presente, se infiere que es igual a 0.

- 30 El elemento de sintaxis cu\_affine\_typeflag[x0][y0] puede utilizarse para indicar si el modelo de movimiento afín de 6 parámetros se utiliza para realizar compensación de movimiento para el bloque actual cuando el sector en la cual se ubica el bloque actual es un sector de tipo P o un sector de tipo B. cu\_affine\_type\_flag[x0][y0] = 0 indica que el modelo de movimiento afín de 6 parámetros no se utiliza para realizar compensación de movimiento para el bloque actual, y solamente el modelo de movimiento afín de 4 parámetros puede utilizarse  
35 para realizar compensación de movimiento; y cu\_affine\_type\_flag[x0][y0] = 1 indica que el modelo de movimiento afín de 6 parámetros se utiliza para realizar compensación de movimiento para el bloque actual. Es decir, cu\_affine\_type\_flag[ x0 ][ y0 ] igual a 1 especifica que para la unidad de codificación actual, cuando se decodifica un grupo de P o B mosaicos, se utiliza compensación de movimiento basada en modelo afín de 6 parámetros para generar las muestras de predicción de la unidad de codificación actual.  
40 cu\_affine\_type\_flag[ x0 ][ y0 ] igual a 0 especifica que compensación de movimiento basada en modelo afín de 4 parámetros se utiliza para generar las muestras de predicción de la unidad de codificación actual.

Tal como se muestra en la Tabla 3, MotionModelldc[x0][y0] = 1 indica que el modelo de movimiento afín de 4 parámetros se utiliza, MotionModelldc[x0][y0] = 2 indica que el modelo de movimiento afín de 6 parámetros se  
45 utiliza, y MotionModelldc[x0][y0] = 0 indica que el modelo de movimiento de traslación se utiliza.

Tabla 3

	Modelo de movimiento para compensación de movimiento (motion model for motion compensation)
0	Traslacional movimiento (translational motion)
1	Movimiento afín de 4 parámetros (4-parameter affine motion)
2	Movimiento afín de 6 parámetros (6-parameter affine motion)

Variables MaxNumMergeCand y MaxAffineNumMrgCand se utilizan para representar una longitud máxima de lista, e indican la longitud máxima de una lista de vectores de movimiento candidatos construida.  
50 inter\_pred\_idc[x0][y0] se utiliza para indicar una dirección de predicción. PRED\_L1 se utiliza para indicar la

- predicción hacia atrás. num\_ref\_idx\_10\_activo\_minus1 indica el número de fotogramas de referencia en una lista de fotogramas de referencia hacia delante, y ref\_idx\_10[x0][y0] indica el valor de índice de fotograma de referencia hacia delante del bloque actual. mvd\_coding(x0, y0, 0, 0) indica la primera diferencia de vector de movimiento. mvp\_10\_flag[x0][y0] indica un valor de índice de lista de candidatos MVP hacia delante. PRED\_L0 indica predicción hacia delante. num\_ref\_idx\_11\_activo\_minus1 indica el número de fotogramas de referencia en una lista de fotogramas de referencia hacia atrás. ref\_idx\_11[x0][y0] indica el valor de índice de fotograma de referencia hacia atrás del bloque actual, y mvp\_11\_flag[x0][y0] indica un valor de índice de lista de candidatos MVP hacia atrás.
- 10 En la Tabla 1, ae(v) representa un elemento de sintaxis codificado mediante el uso de codificación aritmética binaria adaptable basada en contexto (cabac).
- La FIG. 9A es un diagrama de flujo que muestra un proceso de un método de decodificación según una realización de esta solicitud. El proceso puede realizarse por una unidad de interpredicción 344 de un decodificador de vídeo 30. El proceso se describe como una serie de etapas u operaciones. Se debería entender que el proceso puede realizarse en diversos órdenes y/o concurrentemente, y no se limita al orden de ejecución mostrado en la FIG. 9A. Se asume que se emplea un decodificador de vídeo para decodificar un flujo de datos de vídeo que tiene una pluralidad de fotogramas de vídeo utilizando un proceso que incluye el proceso de interpredicción mostrado en la FIG. 9A.
- 20 Etapa 601: Analizar un flujo de bits con base en una estructura de sintaxis mostrada en la Tabla 1, para determinar el modo de interpredicción del bloque actual.
- 25 En caso de que se determine que el modo de interpredicción del bloque actual es un modo AMVP basado en modelo de movimiento afín, realizar la etapa 602a.
- Por ejemplo, los elementos de sintaxis merge\_flag = 0 e inter\_affine\_flag = 1 indican que el modo de interpredicción del bloque actual es el modo AMVP basado en modelo de movimiento afín.
- 30 Si se determine que el modo de interpredicción del bloque actual es un modo fusionar basado en modelo de movimiento afín, realizar la etapa 602b.
- Por ejemplo, los elementos de sintaxis merge\_flag = 1 e inter\_affine\_flag = 1 indican que el modo de interpredicción del bloque actual es el modo AMVP basado en modelo de movimiento afín.
- 35 Etapa 602a: Construir una lista de vectores de movimiento candidatos correspondiente al modo AMVP basado en modelo de movimiento afín.
- Uno o más candidatos de vectores de movimiento de punto de control (por ejemplo uno o más candidatos de X tuplas) del bloque actual pueden derivarse utilizando un método de predicción de vector de movimiento de punto de control heredado y/o un método de predicción de vector de movimiento de punto de control construido. Estos candidatos de vectores de movimiento de punto de control pueden añadirse a la lista de vectores de movimiento candidatos.
- 40 La lista de vectores de movimiento candidatos puede incluir una lista de tuplas de 2 (se usa un modelo de movimiento afín de 4 parámetros para el bloque de codificado actual) o una lista de tuplas de 3. La lista de tuplas de 2 incluye una o más tuplas de 2 utilizadas para construir un modelo de movimiento afín de 4 parámetros. La lista de tuplas de 3 incluye una o más tuplas de 3 utilizadas para construir un modelo de movimiento afín de 6 parámetros. Se puede entender que cada candidato de tuplas de 2 incluye dos vectores de movimiento de punto de control candidatos del bloque actual.
- 50 Opcionalmente, la lista de vectores de movimiento candidatos de tupla de 2/tupla de 3 se recorta y clasifica según una regla particular, y puede ser truncada o rellenada para incluir una cantidad particular de tuplas de 2 o tuplas de 3.
- 55 A1: Se describe un proceso para construir la lista de vectores de movimiento candidatos utilizando el método de predicción de vector de movimiento de punto de control heredado.
- La FIG. 7 se utiliza como ejemplo. En este ejemplo, bloques en ubicaciones vecinas alrededor del bloque actual son atravesados en un orden de A1->B1->B0->A0->B2 para encontrar un bloque codificado afín que contiene un bloque en una ubicación vecina del bloque actual, y obtener información de movimiento de punto de control del bloque codificado afín. La información de movimiento de punto de control del bloque codificado afín puede utilizarse para construir un modelo de movimiento para derivar la información de movimiento de punto de control de candidatos del bloque actual. Detalles de este proceso se proporcionan antes en las descripciones del método de predicción de vector de movimiento de punto de control heredado en 3.
- 60
- 65

- En un ejemplo, el modelo de movimiento afín utilizado para el bloque actual es un modelo de movimiento afín de 4 parámetros (es decir,  $\text{MotionModelIdc} = 1$ ). En este ejemplo, si el modelo de movimiento afín de 4 parámetros se utiliza para un bloque de decodificación afín vecino, se obtienen vectores de movimiento de dos puntos de control del bloque de decodificación afín: el vector de movimiento ( $v_{x4}$ ,  $v_{y4}$ ) del punto de control superior-izquierdo ( $x_4$ ,  $y_4$ ), y el vector de movimiento ( $v_{x5}$ ,  $v_{y5}$ ) del punto de control superior-derecho ( $x_5$ ,  $y_5$ ). El bloque de decodificación afín es un bloque codificado afín predicho en una fase de codificación utilizando un modelo de movimiento afín.
- Los vectores de movimiento de los dos puntos de control, concretamente, los puntos de control superior-izquierdo y superior-derecho, del bloque actual se derivan según las fórmulas (6) y (7) del modelo de movimiento afín de 4 parámetros, respectivamente, utilizando un modelo de movimiento afín de 4 parámetros incluyendo dos puntos de control del bloque de decodificación afín vecino.
- Si un modelo de movimiento afín de 6 parámetros se utiliza para el bloque de decodificación afín vecino, se obtienen vectores de movimiento de tres puntos de control del bloque de decodificación afín vecino, por ejemplo, el vector de movimiento ( $v_{x4}$ ,  $v_{y4}$ ) del punto de control superior-izquierdo ( $x_4$ ,  $y_4$ ), el vector de movimiento ( $v_{x5}$ ,  $v_{y5}$ ) del punto de control superior-derecho ( $x_5$ ,  $y_5$ ), y el vector de movimiento ( $v_{x6}$ ,  $v_{y6}$ ) del punto de control inferior-izquierdo ( $x_6$ ,  $y_6$ ) en la FIG. 7.
- Los vectores de movimiento de los dos puntos de control, concretamente, los puntos de control superior-izquierdo y superior-derecho, del bloque actual se derivan según las fórmulas (8) y (9) del modelo de movimiento afín de 6 parámetros, respectivamente, utilizando el modelo de movimiento afín de 6 parámetros incluyendo tres puntos de control del bloque de decodificación afín vecino.
- En otro ejemplo, el modelo de movimiento afín utilizado para el bloque de decodificación actual es un modelo de movimiento afín de 6 parámetros (es decir,  $\text{MotionModelIdc} = 2$ ).
- Si el modelo de movimiento afín utilizado para el bloque de decodificación afín vecino es el modelo de movimiento afín de 6 parámetros, se obtienen vectores de movimiento de tres puntos de control del bloque de decodificación afín vecino, por ejemplo, el vector de movimiento ( $v_{x4}$ ,  $v_{y4}$ ) del punto de control superior-izquierdo ( $x_4$ ,  $y_4$ ), el vector de movimiento ( $v_{x5}$ ,  $v_{y5}$ ) del punto de control superior-derecho ( $x_5$ ,  $y_5$ ), y el vector de movimiento ( $v_{x6}$ ,  $v_{y6}$ ) del punto de control inferior-izquierdo ( $x_6$ ,  $y_6$ ) en la FIG. 7.
- Los vectores de movimiento de los tres puntos de control, concretamente, los puntos de control superior-izquierdo, superior-derecho, e inferior-izquierdo, del bloque actual se derivan según las fórmulas (8), (9), y (10) correspondientes al modelo de movimiento afín de 6 parámetros, respectivamente, utilizando el modelo de movimiento afín de 6 parámetros incluyendo tres puntos de control del bloque de decodificación afín vecino.
- Si el modelo de movimiento afín utilizado para el bloque de decodificación afín vecino es un modelo de movimiento afín de 4 parámetros, se pueden obtener vectores de movimiento de dos puntos de control del bloque de decodificación afín vecino. Estos vectores de movimiento pueden ser, por ejemplo, el vector de movimiento ( $v_{x4}$ ,  $v_{y4}$ ) del punto de control superior-izquierdo ( $x_4$ ,  $y_4$ ) y el vector de movimiento ( $v_{x5}$ ,  $v_{y5}$ ) del punto de control superior-derecho ( $x_5$ ,  $y_5$ ).
- Los vectores de movimiento de tres puntos de control, tal como los puntos de control superior-izquierdo, superior-derecho e inferior-izquierdo del bloque actual pueden derivarse. Por ejemplo, estos vectores de movimiento pueden derivarse según las fórmulas (6) y (7) del modelo de movimiento afín de 4 parámetros utilizando el modelo de movimiento afín de 4 parámetros representado con base en dos puntos de control del bloque de decodificación afín vecino.
- Se debería observar que otros modelos de movimiento, ubicaciones candidatos, y órdenes de búsqueda también se pueden emplear en esta memoria. Además, también se pueden utilizar métodos para representar los modelos de movimiento de bloques de codificado vecinos y actuales con base en otros puntos de control.
- A2: Se describe un proceso de construir una lista de vectores de movimiento candidatos utilizando el método de predicción de vector de movimiento de punto de control construido.
- En un ejemplo, el modelo de movimiento afín utilizado para un bloque de decodificación actual es un modelo de movimiento afín de 4 parámetros (es decir,  $\text{MotionModelIdc} = 1$ ). En este ejemplo, vectores de movimiento de la muestra superior-izquierda y la muestra superior-derecha del bloque de codificado actual se determinan con base en información de movimiento de un bloque codificado vecino del bloque de codificado actual. Específicamente, la lista de vectores de movimiento candidatos puede construirse utilizando el método de predicción de vector de movimiento de punto de control construido 1 descrito anteriormente con respecto al elemento 4 o el método de predicción de vector de movimiento de punto de control construido 2 descrito anteriormente con respecto al elemento 5.

En otro ejemplo, el modelo de movimiento afín utilizado para el bloque de decodificación actual es un modelo de movimiento afín de 6 parámetros (es decir,  $\text{MotionModelIdc} = 2$ ). En este ejemplo, vectores de movimiento de la muestra superior-izquierda, la muestra superior-derecha, y la muestra inferior-izquierda del bloque de codificado actual se determinan utilizando información de movimiento de un bloque codificado vecino del bloque de codificado actual. Específicamente, la lista de vectores de movimiento candidatos puede construirse utilizando el método de predicción de vector de movimiento de punto de control construido 1 descrito anteriormente con respecto al elemento 4 o el método de predicción de vector de movimiento de punto de control construido 2 descrito anteriormente con respecto al elemento 5.

Se debería observar que también se pueden emplear otras combinaciones de información de movimiento de punto de control.

Etapa 603a: Analizar el flujo de bits y determinar predictores de vector de movimiento de punto de control óptimos (es decir, un candidato de tupla múltiple óptimo).

B1: Si un modelo de movimiento afín utilizado para el bloque de decodificación actual es un modelo de movimiento afín de 4 parámetros ( $\text{MotionModelIdc} = 1$ ), un número de índice se analiza del flujo de bits, y predictores de vector de movimiento óptimos de dos puntos de control se determinan de la lista de vectores de movimiento candidatos con base en el número de índice.

Por ejemplo, el número de índice es `mvp_10_flag` o `mvp_11_flag`.

B2: Si un modelo de movimiento afín utilizado para el bloque de decodificación actual es un modelo de movimiento afín de 6 parámetros ( $\text{MotionModelIdc} = 2$ ), un número de índice se analiza del flujo de bits, y predictores de vector de movimiento óptimos de tres puntos de control se determinan de la lista de vectores de movimiento candidatos con base en el número de índice.

Etapa 604a: Analizar sintácticamente el flujo de bits y determinar vectores de movimiento de punto de control.

C1: Si un modelo de movimiento afín utilizado para el bloque de decodificación actual es un modelo de movimiento afín de 4 parámetros ( $\text{MotionModelIdc} = 1$ ), diferencias de vector de movimiento de dos puntos de control del bloque actual son respectivamente obtenidas decodificando el flujo de bits. Los valores de vector de movimiento de los dos puntos de control se obtienen entonces con base en las diferencias de vector de movimiento y los predictores de vector de movimiento correspondientes de los puntos de control. Utilizando la predicción hacia delante como un ejemplo, diferencias de vector de movimiento de dos puntos de control son `mvd_coding(x0, y0, 0, 0)` y `mvd_coding(x0, y0, 0, 1)`, respectivamente.

Por ejemplo, diferencias de vector de movimiento del punto de control superior-izquierdo y el punto de control superior-derecho se obtienen decodificando el flujo de bits, y se añaden a sus predictores de vector de movimiento respectivos, para obtener vectores de movimiento del punto de control superior-izquierdo y el punto de control superior-derecho del bloque actual.

C2: Un modelo de movimiento afín utilizado para un bloque de decodificación actual es un modelo de movimiento afín de 6 parámetros ( $\text{MotionModelIdc} = 2$ ).

Diferencias de vector de movimiento de tres puntos de control del bloque actual se obtienen decodificando el flujo de bits. Valores de vector de movimiento de estos puntos de control se obtienen con base en diferencias de vector de movimiento y predictores de vector de movimiento respectivos de los puntos de control. Utilizando predicción hacia delante (es decir, lista 0) como un ejemplo, diferencias de vector de movimiento de tres puntos de control son `mvd_coding(x0, y0, 0, 0)`, `mvd_coding(x0, y0, 0, 1)`, y `mvd_coding(x0, y0, 0, 2)`, respectivamente.

Por ejemplo, diferencias de vector de movimiento del punto de control superior-izquierdo, el punto de control superior-derecho, y el punto de control inferior-izquierdo se obtienen decodificando el flujo de bits. Estas diferencias de vector de movimiento se añaden a sus predictores de vector de movimiento respectivos para obtener vectores de movimiento del punto de control superior-izquierdo, el punto de control superior-derecho, y el punto de control inferior-izquierdo del bloque actual.

Etapa 605a: Obtener un vector de movimiento de cada subbloque en el bloque actual con base en información de movimiento de punto de control y un modelo de movimiento afín utilizado para el bloque de decodificación actual.

Un subbloque en el bloque de decodificación afín actual puede ser equivalente a una unidad de compensación de movimiento, y la anchura y la altura del subbloque son menores que la anchura y la altura del bloque actual. Información de movimiento de una muestra en una ubicación preestablecida en un subbloque o unidad de compensación de movimiento puede utilizarse para representar información de movimiento de todas las muestras en el subbloque o la unidad de compensación de movimiento. Asumiendo que el tamaño de la unidad

de compensación de movimiento es  $M \times N$ , la muestra en la ubicación preestablecida puede ser una muestra central ( $M/2, N/2$ ), una muestra superior-izquierda (0, 0), una muestra superior-derecha ( $M-1, 0$ ), o una muestra en otra ubicación en la unidad de compensación de movimiento. La siguiente descripción utiliza la muestra central de la unidad de compensación de movimiento como un ejemplo para descripción. Haciendo referencia a la FIG. 9C,  $V_0$  representa un vector de movimiento del punto de control superior-izquierdo, y  $V_1$  representa un vector de movimiento del punto de control superior-derecho. Cada caja pequeña representa una unidad de compensación de movimiento.

Coordenadas de una muestra central de una unidad de compensación de movimiento con relación a la muestra superior-izquierda en un bloque de decodificación afín actual se calculan según la fórmula (25). En la fórmula (25),  $i$  indica la  $i$ -ésima unidad de compensación de movimiento en la dirección horizontal (izquierda a derecha),  $j$  indica la  $j$ -ésima unidad de compensación de movimiento en la dirección vertical (superior a inferior), y  $(x_{(i,j)}, y_{(i,j)})$  representa las coordenadas de la muestra central de la  $(i, j)$ -ésima unidad de compensación de movimiento con relación a la muestra de punto de control superior-izquierdo en el bloque de decodificación afín actual.

Si el modelo de movimiento afín utilizado para el bloque de decodificación afín actual es un modelo de movimiento afín de 6 parámetros,  $(x_{(i,j)}, y_{(i,j)})$  se sustituye en la fórmula (26) del modelo de movimiento afín de 6 parámetros para obtener el vector de movimiento de la muestra central de cada unidad de compensación de movimiento  $(vx_{(i,j)}, vy_{(i,j)})$ . Tal como se ha analizado antes, el vector de movimiento del píxel central de una unidad de compensación de movimiento se utiliza como los vectores de movimiento de todas las muestras en la unidad de compensación de movimiento.

Si el modelo de movimiento afín utilizado para el bloque de decodificación afín actual es un modelo de movimiento afín de 4 parámetros,  $(x_{(i,j)}, y_{(i,j)})$  se sustituye en la fórmula (27) del modelo de movimiento afín de 4 parámetros, para obtener el vector de movimiento de una muestra central de cada unidad de compensación de movimiento  $(vx_{(i,j)}, vy_{(i,j)})$  que se utiliza como los vectores de movimiento de todas las muestras en la unidad de compensación de movimiento.

$$\begin{cases} x_{(i,j)} = M \times i + \frac{M}{2}, i = 0, 1, \dots \\ y_{(i,j)} = N \times j + \frac{N}{2}, j = 0, 1, \dots \end{cases} \quad (25)$$

$$\begin{cases} vx = \frac{vx_1 - vx_0}{W} x + \frac{vx_2 - vx_0}{H} y + vx_0 \\ vy = \frac{vy_1 - vy_0}{W} x + \frac{vy_2 - vy_0}{H} y + vy_0 \end{cases} \quad (26)$$

$$\begin{cases} vx = \frac{vx_1 - vx_0}{W} x - \frac{vy_1 - vy_0}{W} y + vx_0 \\ vy = \frac{vy_1 - vy_0}{W} x + \frac{vx_1 - vx_0}{W} y + vy_0 \end{cases} \quad (27)$$

Etapla 606a: Realizar compensación de movimiento para cada subbloque con base en el vector de movimiento determinado del subbloque, para obtener valores de muestra de predicción del subbloque.

Tal como se ha analizado antes, si en la etapa 601 se determina que el modo de interpredicción del bloque actual es un modo de fusión (merge) basado en modelo de movimiento afín, se realiza la etapa 602b.

Etapla 602b: Construir una lista de candidatos de información de movimiento correspondiente al modo de fusión basado en modelo de movimiento afín.

Específicamente, la lista de candidatos de información de movimiento correspondiente al modo de fusión basado en modelo de movimiento afín puede construirse utilizando un método de predicción de vector de movimiento de punto de control heredado y/o un método de predicción de vector de movimiento de punto de control construido.

Opcionalmente, la lista de candidatos de información de movimiento se recorta y clasifica según una regla particular, y puede ser truncada o rellenada para contener una cantidad particular de información de movimiento.

D1: Se describe un proceso para construir la lista de vectores de movimiento candidatos utilizando el método de predicción de vector de movimiento de punto de control heredado.

Información de movimiento de punto de control candidata del bloque actual se deriva utilizando el método de predicción de vector de movimiento de punto de control heredado, y se añade a la lista de candidatos de información de movimiento.



En el ejemplo mostrado en la FIG. 8A, bloques en ubicaciones vecinas alrededor del bloque actual son  
atravesados según el orden de A1->B1->B0->A0->B2 para encontrar un bloque codificado afín en el que se  
ubica un bloque vecino, y obtener información de movimiento de punto de control del bloque codificado afín.  
Además, información de movimiento de punto de control candidata del bloque actual se deriva utilizando el  
5 modelo de movimiento del bloque actual.

Si la lista de vectores de movimiento candidatos está vacía, la información de movimiento de punto de control  
candidata anterior se obtiene se añade a la lista de candidatos. De otra manera, información de movimiento en  
la lista de vectores de movimiento candidatos se atraviesa en secuencia, para comprobar si la información de  
10 movimiento que es la misma que la información de movimiento de punto de control candidata existe en la lista  
de vectores de movimiento candidatos. Si ninguna información de movimiento que es la misma que la  
información de movimiento de punto de control candidata existe en la lista de vectores de movimiento  
candidatos, la información de movimiento de punto de control candidata se añade a la lista de vectores de  
movimiento candidatos.

15 Determinar si dos elementos de información de movimiento candidata son iguales puede realizarse mediante  
la determinación respecto a si los fotogramas de referencia hacia delante (lista 0) y hacia atrás (lista 1) de la  
información de movimiento candidata y los componentes horizontal y vertical de cada vector de movimiento  
hacia delante y hacia atrás son iguales. Los dos elementos de información de movimiento candidato se  
20 consideran diferentes únicamente cuando todos estos elementos son diferentes.

Si una cantidad de elementos de información de movimiento en la lista de vectores de movimiento candidatos  
alcanza una longitud máxima de lista MaxAffineNumMrgCand (MaxAffineNumMrgCand es un entero positivo  
tal como 1, 2, 3, 4 o 5, y 5 se utiliza como ejemplo en la siguiente descripción), se ha completado la construcción  
25 de la lista de candidatos. De otra manera, se atraviesa un siguiente bloque vecino.

D2: Información de movimiento de punto de control candidata del bloque actual se deriva utilizando el método  
de predicción de vector de movimiento de punto de control heredado, y se añade a la lista de candidatos de  
información de movimiento. La FIG. 9B muestra un ejemplo de un diagrama de flujo para el método de  
30 predicción de vectores de movimiento de punto de control construido.

Etapa 601c: Obtener información de movimiento de puntos de control del bloque actual. Esta etapa es similar  
a la etapa 501 en "5. Método de predicción de vector de movimiento de punto de control construido 2". En esta  
memoria no se repiten detalles.

35 Etapa 602c: Combinar la información de movimiento de puntos de control, para obtener información de  
movimiento de punto de control construido. Esta etapa es similar a la etapa 501 en la FIG. 8B y los detalles de  
esta etapa no se describen nuevamente en esta memoria.

40 Etapa 603c: Añadir la información de movimiento de punto de control construido a la lista de vectores de  
movimiento candidatos.

Si la longitud de la lista de candidatos es menor que una longitud máxima de lista MaxAffineNumMrgCand, las  
combinaciones de información de movimiento de puntos de control son atravesadas en un orden  
45 preestablecido, y una combinación válida resultante se utiliza como la información de movimiento de punto de  
control candidata. En este caso, si la lista de vectores de movimiento candidatos está vacía, la información de  
movimiento de punto de control candidata se añade a la lista de vectores de movimiento candidatos. De otra  
manera, información de movimiento en la lista de vectores de movimiento candidatos se atraviesa en  
secuencia, para comprobar si la información de movimiento que es la misma que la información de movimiento  
50 de punto de control candidata existe en la lista de vectores de movimiento candidatos. Si ninguna información  
de movimiento que es la misma que la información de movimiento de punto de control candidata existe en la  
lista de vectores de movimiento candidatos, la información de movimiento de punto de control candidata se  
añade a la lista de vectores de movimiento candidatos.

55 Por ejemplo, un orden preestablecido es de la siguiente manera: Afín (CP1, CP2, CP3) -> Afín (CP1, CP2,  
CP4) -> Afín (CP1, CP3, CP4) -> Afín (CP2, CP3, CP4) -> Afín (CP1, CP2) -> Afín (CP1, CP3) -> Afín (CP2,  
CP3) -> Afín (CP1, CP4) -> Afín (CP2, CP4) -> Afín (CP3, CP4). Hay 10 combinaciones en total.

Si no hay disponible información de movimiento de punto de control correspondiente a una combinación, esta  
combinación se considera no disponible. Si hay disponible una combinación, se determina un índice de  
60 fotograma de referencia de la combinación. En un caso de dos puntos de control, el índice de fotograma de  
referencia mínimo se selecciona como el índice de fotograma de referencia de la combinación. En un caso de  
más de dos puntos de control, un índice de fotograma de referencia con una frecuencia de presencia máxima  
se selecciona como el índice de fotograma de referencia de la combinación. Si una pluralidad de índices de  
fotograma de referencia tiene la misma frecuencia de presencia, el índice de fotograma de referencia mínimo  
65 se selecciona como el índice fotograma de referencia. El vector de movimiento de punto de control se escala

aún más. Si información de movimiento de todos los puntos de control después del escalado es consistente, la combinación es inválida.

5 Opcionalmente, en esta realización de esta solicitud, la lista de vectores de movimiento candidatos puede rellenarse. Por ejemplo, después del proceso transversal anterior, si la longitud de la lista de vectores de movimiento candidatos es menor que la longitud máxima de lista MaxAffineNumMrgCand, la lista de vectores de movimiento candidatos puede rellenarse hasta que la longitud de lista es igual a MaxAffineNumMrgCand.

10 El relleno puede realizarse utilizando un método de relleno de vector de movimiento cero, o combinando (por ejemplo, promediando por ponderación) información de movimiento candidata existente en la lista existente. Se debería observar que otros métodos para rellenar la lista de vectores de movimiento candidatos son también aplicables a esta solicitud.

15 Etapa 603b: Analizar sintácticamente el flujo de bits y determinar información de movimiento de punto de control óptima.

Se analiza un número de índice, y se determina la información de movimiento de punto de control óptima de la lista de vectores de movimiento candidatos con base en el número de índice.

20 Etapa 604b: Obtener un vector de movimiento de cada subbloque en el bloque actual con base en información de movimiento de punto de control óptimo y un modelo de movimiento afín utilizado para el bloque de decodificación actual.

Esta etapa es similar a la etapa 605a.

25 Etapa 605b: Realizar compensación de movimiento para cada subbloque con base en el vector de movimiento determinado del subbloque, para obtener valores de muestra de predicción del subbloque.

30 Tal como se ha descrito antes, después que el vector de movimiento de cada subbloque se obtiene en la etapa 605a y la etapa 604b, se realiza compensación de movimiento para el subbloque en la etapa 606a y 605b, respectivamente. Es decir, los detalles de la realización de la compensación de movimiento afín basada en subbloque para un subbloque actual del bloque codificado afín, para obtener valores de muestra de predicción del subbloque actual del bloque codificado afín se describieron anteriormente. En el diseño convencional, el tamaño del subbloque se establece a 4x4, es decir, se realiza compensación de movimiento para cada unidad 35 4x4 utilizando un vector de movimiento respectivo/diferente. Generalmente, un tamaño más pequeño de un subbloque conduce a una complejidad de cálculo de compensación de movimiento superior y un mejor efecto de predicción. Para tomar en consideración la complejidad de cálculo de compensación de movimiento y la precisión de predicción, después de la compensación de movimiento a nivel de subbloque, se proporciona un proceso para refinamiento de señal de predicción con flujo óptico (PROF). Etapas ejemplares del proceso son de la siguiente manera:

40 (1) Después que se obtiene un vector de movimiento de cada subbloque utilizando las etapas 605a y 604b, y después de realizar la compensación de movimiento para el subbloque utilizando las etapas 606a y 605b, obtener una señal de predicción  $I(i,j)$  del subbloque. Se puede observar que la etapa (1) no está incluida en el proceso PROF.

(2) Calcular un valor de gradiente horizontal  $g_x(i,j)$  y un valor de gradiente vertical  $g_y(i,j)$  de la señal de predicción del subbloque, donde un método de cálculo es de la siguiente manera:

50 
$$g_x(i,j) = I(i+1,j) - I(i-1,j)$$

$$g_y(i,j) = I(i,j+1) - I(i,j-1)$$

55 Se puede aprender a partir de las fórmulas que, para obtener los valores de gradiente para un bloque 4x4 (valores de gradiente 4x4), se requiere una ventana de señal de predicción 6x6 900, tal como se muestra en la FIG. 9D.

Esto puede implementarse utilizando los siguientes métodos diferentes:

60 a) Después que se obtiene una matriz de predicción de un subbloque con base en información de movimiento (por ejemplo, un vector de movimiento) del subbloque, obtener una matriz de gradiente horizontal y una matriz de gradiente vertical del subbloque. En otras palabras el bloque de predicción  $(M+2)*(N+2)$  se obtiene mediante interpolación basada en el vector de movimiento del subbloque  $M \times N$ . Por ejemplo, la interpolación se realiza directamente en base al vector de movimiento del sub-bloque, para obtener una señal de predicción 6x6 y 65 calcular un valor de gradiente de 4x4 (es decir, una matriz de gradiente de 4x4).

5 b) Realizar una interpolación basada en el vector de movimiento del sub-bloque, para obtener una señal de predicción 4x4 (es decir, la primera (es decir, la primera matriz de predicción) y, a continuación, extender los bordes de la señal de predicción para obtener una señal de predicción de 6x6 (es decir, la segunda matriz de predicción) y calcular el valor de la señal de predicción. la segunda matriz de predicción) y calcular un valor de gradiente 4x4 (es decir, una matriz de gradiente 4x4).

10 c) Realizar una interpolación basada en el vector de movimiento de cada sub-bloque, para obtener cada señal de predicción 4x4 (es decir, la primera matriz de predicción), y obtener un valor de gradiente 4x4 (es decir, la segunda matriz de predicción). primera matriz de predicción), y obtener una señal de predicción w\*h mediante combinación. A continuación, se realiza la extensión de bordes en la señal de predicción w\*h, para obtener una señal de predicción (w+2)\*(h+2), y calcular un valor de gradiente w\*h (es decir, la matriz de gradiente w\*h ), para obtener cada valor de gradiente 4x4 (es decir, matriz de gradiente 4x4).

15 Se debería observar que, la obtención de un bloque de predicción (M+2)\*(N+2) directamente a través de interpolación con base en el vector de movimiento del subbloque MxN incluye las siguientes implementaciones:

20 a1) para una región circundante (muestras blancas en la FIG. 13), se obtiene una muestra de entero de una muestra superior-izquierda de una ubicación a la que apunta el vector de movimiento. Para una región interior (muestras grises en la FIG. 13), se obtiene una muestra de una ubicación a la que apunta el vector de movimiento. Si la muestra es una muestra fraccional, la muestra se obtiene a través de interpolación utilizando un filtro de interpolación.

25 Tal como se muestra en la FIG. 14, A, B, C y D son muestras de entero, un vector de movimiento de un subbloque MxN es de precisión de muestra 1/16, dx/16 es una distancia horizontal entre una muestra fraccional y la muestra de entero de una muestra superior-izquierda, y dy/16 es una distancia vertical entre la muestra fraccional y la muestra de entero de la muestra superior-izquierda. Para la región circundante, un valor de muestra de A se utiliza como el valor de muestra de predicción de la ubicación de muestra. Para una región interior, el valor de muestra de predicción de la ubicación de muestra se obtiene a través de interpolación utilizando un filtro de interpolación.

30 a2) para una región circundante (muestras blancas en la FIG. 13), se obtiene una muestra de entero más cercana de una ubicación a la que apunta el vector de movimiento. Para una región interior (muestras grises en la FIG. 13), se obtiene una muestra de una ubicación a la que apunta el vector de movimiento. Si la muestra es una muestra fraccional, la muestra se obtiene a través de interpolación utilizando un filtro de interpolación.

35 Tal como se muestra en la FIG. 14, para una región circundante, una muestra de entero más cercana a una ubicación a la que apunta el vector de movimiento se selecciona con base en dx y dy.

40 a3) Para una región circundante y una región interior, se obtiene la muestra de una ubicación a la que apunta el vector de movimiento. Si la muestra es una muestra fraccional, la muestra se obtiene a través de interpolación utilizando un filtro de interpolación.

45 Se debería entender que a), b), y c) son tres implementaciones diferentes.

(3) Calcular un valor de predicción delta, donde un método de cálculo es de la siguiente manera:

$$\Delta I(i, j) = g_x(i, j) * \Delta v_x(i, j) + g_y(i, j) * \Delta v_y(i, j)$$

50 (i, j) representa una muestra actual del subbloque,  $\Delta v(i, j)$  es una diferencia (tal como se muestra en la FIG. 10) entre un vector de movimiento de la muestra actual del subbloque actual y un vector de movimiento de una muestra central del subbloque, y se puede calcular según la fórmula anterior,  $\Delta v_x(i, j)$  y  $\Delta v_y(i, j)$  son un valor de compensación horizontal y un valor de compensación vertical de una diferencia entre un vector de movimiento de la muestra actual del subbloque actual y un vector de movimiento de una muestra central del subbloque.

55 Alternativamente, en un método simplificado, se puede calcular una diferencia de vector de movimiento entre un vector de movimiento de cada bloque de muestras 2x2 al que pertenece la muestra actual y un vector de movimiento de una muestra central del subbloque. En comparación,  $\Delta v(i, j)$ : una diferencia de vector de movimiento necesita calcularse para cada píxel o muestra (por ejemplo, se necesita realizar un cálculo 16 veces para un subbloque 4x4). Sin embargo, en el método simplificado, una diferencia de vector de movimiento se calcula para cada subbloque 2x2 (por ejemplo, el cálculo se realiza cuatro veces para el subbloque 4x4).

60 Se debería observar que el subbloque en esta memoria puede ser un subbloque 4x4 o un subbloque mxn. Por ejemplo, m en esta memoria es mayor o igual a 4, o n aquí es mayor o igual a 4.

$$\begin{cases} \Delta v_x(x, y) = c * x + d * y \\ \Delta v_y(x, y) = e * x + f * y \end{cases}$$

Para un modelo afín de 4 parámetros:

$$\begin{cases} c = f = \frac{v_{1x} - v_{0x}}{w} \\ e = -d = \frac{v_{1y} - v_{0y}}{w} \end{cases}$$

5

Para un modelo afín de 6 parámetros:

$$\begin{cases} c = \frac{v_{1x} - v_{0x}}{w} \\ d = \frac{v_{2x} - v_{0x}}{h} \\ e = \frac{v_{1y} - v_{0y}}{w} \\ f = \frac{v_{2y} - v_{0y}}{h} \end{cases}$$

- 10 donde  $(v_{0x}, v_{0y})$ ,  $(v_{1x}, v_{1y})$  y  $(v_{2x}, v_{2y})$  son vectores de movimiento de puntos de control superior-izquierdo, superior-derecho e inferior-izquierdo, y  $w$  y  $h$  son una anchura y una altura de un bloque codificado afín (CU).

(4) Realizar refinamiento de predicción:

15  $I'(i, j) = I(i, j) + \Delta I(i, j)$

donde  $I(i, j)$  es un valor de predicción de una muestra  $(i, j)$  del subbloque (es decir, valor de predicción de la muestra en la posición  $(i, j)$  del subbloque),  $\Delta I(i, j)$  es un valor de predicción delta de la muestra  $(i, j)$  del subbloque, e  $I'(i, j)$  es un valor de predicción refinado de la muestra  $(i, j)$  del subbloque.

20

A continuación se describe el proceso de refinamiento de predicción con flujo óptico (PROF) que se realiza de manera condicional para refinar la predicción compensada de movimiento afín basada en subbloque con flujo óptico según la realización de la presente divulgación.

25 Realización 1

Un método para obtener valores de predicción delta de un subbloque (que es específicamente un valor de predicción delta de cada muestra del subbloque) con flujo óptico puede aplicarse a un bloque codificado afín unidireccional, o puede aplicarse a un bloque codificado afín bidireccional. Si el método se aplica a un bloque de predicción afín bidireccional, las etapas (1) a (4) antes descritas necesitan realizarse dos veces, teniendo como resultado complejidad computacional relativamente alta. Para reducir la complejidad del método, la presente invención proporciona una restricción para aplicar PROF. Para ser específicos, valores de muestra de predicción se refinan utilizando este método solamente cuando el bloque codificado afín es un bloque codificado afín unidireccional.

35

En el lado del decodificador, un elemento de sintaxis obtenido mediante el análisis de un flujo de bits indica uni- o bi-predicción. Este elemento de sintaxis puede utilizarse para determinar si el bloque codificado afín es un bloque codificado afín unidireccional.

- 40 En el lado del codificador, estructuras de fotogramas B y P se determinan por diferentes casos de usuario, y RDO determina si se utiliza uni- o bi-predicción en el fotograma B. En otras palabras, para el fotograma B, el lado del codificador puede determinar, con base en un coste RDO, si se utiliza uni- o bi-predicción para el bloque de imagen afín actual. Por ejemplo, el lado del codificador intenta seleccionar un mecanismo entre la dirección de avance, la predicción hacia atrás y la predicción bidireccional que minimizan el RDO.

45

Realización 2

Para reducir la complejidad del refinamiento de señal de predicción con flujo óptico, el método para obtener un valor de compensación de predicción de un subbloque con flujo óptico puede utilizarse únicamente cuando el tamaño del subbloque es relativamente grande. Por ejemplo, el tamaño de subbloque de un bloque codificado afín unidireccional puede establecerse a  $4 \times 4$ , y el tamaño de subbloque de un bloque codificado afín bidireccional puede establecerse a  $8 \times 8$ ,  $8 \times 4$ , o  $4 \times 8$ . En este ejemplo, el método se utiliza únicamente cuando el tamaño de subbloque es mayor que  $4 \times 4$ . Para otro ejemplo, el tamaño de subbloque puede seleccionarse de manera adaptable con base en información tal como vectores de movimiento de puntos de control del bloque codificado afín, la anchura y altura del bloque codificado afín, y similar. El método se utiliza únicamente cuando el tamaño de subbloque es mayor que  $4 \times 4$ .

55

Además, en la etapa (2), los métodos a) y b) pueden garantizar que la predicción de cada subbloque 4x4 del bloque codificado afin no tiene dependencia y puede realizarse concurrentemente. Sin embargo, el método a) aumenta la complejidad del cálculo de interpolación. Sin embargo, el método a) incrementa la complejidad del cálculo de interpolación. Aunque el método b) no incrementa la complejidad, el valor de gradiente de la frontera se obtiene a través del cálculo utilizando una muestra extendida, y la precisión no es alta. El método c) puede mejorar la precisión del cálculo de gradiente, pero cada subbloque 4x4 tiene dependencia, es decir, refinamiento con base en flujo óptico puede realizarse solamente cuando se completa la interpolación de una CU completa.

Tal como se muestra en la FIG. 9E, para tomar en consideración la concurrencia y la precisión del cálculo de gradiente, la presente divulgación propone un cálculo de valor de gradiente con base en una granularidad de 16x16. Dado  $\text{size\_w} = \min(w, 16)$ , y  $\text{size\_h} = \min(h, 16)$ , para cada  $\text{size\_w} \times \text{size\_h}$  en el bloque codificado afin, se calcula el predictor (valores de muestra de predicción) de cada subbloque 4x4 en el bloque codificado afin, y la señal de predicción  $\text{size\_w} \times \text{size\_h}$  se obtiene a través de combinación. Después, la extensión de borde se realiza en la señal de predicción  $\text{size\_w} \times \text{size\_h}$  (por ejemplo, dos muestras extendidas hacia fuera, tal como mediante relleno), para obtener una señal de predicción  $(\text{size\_w}+2) \times (\text{size\_h}+2)$ . Se calcula el valor de gradiente  $\text{size\_w} \times \text{size\_h}$ , para obtener cada valor de gradiente 4x4. Se debería entender que una cantidad de muestras extendidas hacia fuera en esta solicitud no queda limitada a dos muestras, y está relacionada con el cálculo de gradiente. Si la resolución de gradiente es 3-derivaciones, dos muestras se extienden hacia fuera. En otras palabras, esto se relaciona con un filtro para cálculo de gradiente. Asumiendo que una cantidad de derivaciones del filtro es T, una región añadida o una región circundante soportada es  $T/2$  (divisible)\*2.

La FIG. 11A muestra un método para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afin según una realización, el método puede realizarse por un aparato de codificación (tal como un aparato de decodificación o un decodificador). El método incluye las siguientes etapas:

S1101. determinar que se cumple una pluralidad de condiciones de decisión de flujo óptico;

Aquí las condiciones de decisión de flujo óptico también pueden denominarse condiciones que permiten la aplicación de PROF. Si se satisfacen todas las condiciones de decisión de flujo óptico, el PROF se aplica para el subbloque actual del bloque codificado afin. A continuación se describen ejemplos de las condiciones de decisión de flujo óptico. En algunos ejemplos, las condiciones de decisión de flujo óptico pueden ser reemplazadas o rephraseadas como condiciones de restricción para aplicar PROF. Si se satisface una condición de restricción para aplicar PROF, entonces el PROF no se aplica al subbloque actual del bloque codificado afin. En esos ejemplos, la etapa S1101 se cambiará para determinar que no se cumple ninguna de una pluralidad de condiciones de restricción para aplicar PROF.

S1102. realizar un proceso PROF para un subbloque actual del bloque codificado afin, para obtener valores de muestra de predicción refinados del subbloque actual del bloque codificado afin, donde se cumple toda la pluralidad de condiciones de decisión de flujo óptico para el bloque codificado afin. Aquí los valores de muestra de predicción refinados del subbloque actual pueden entenderse como valores de muestra de predicción finales del subbloque actual después que se añade el refinamiento de predicción.

En la etapa S1102, realizar procesamiento de flujo óptico (refinamiento de predicción con flujo óptico, PROF) para uno o más subbloques (por ejemplo, un subbloque actual o cada subbloque) en el bloque de imagen afin actual, para obtener un valor de predicción delta (por ejemplo,  $\Delta l(i, j)$ ) de uno o más subbloques (por ejemplo, el subbloque actual o cada subbloque) en el bloque de imagen afin actual.

En la etapa S1102, un valor de muestra de predicción refinado (por ejemplo, una señal de predicción) del subbloque  $l'(i, j)$  se obtiene con base en el valor de predicción delta (por ejemplo,  $\Delta l(i, j)$ ) y un valor de muestra de predicción (por ejemplo, una señal de predicción  $l(i, j)$ ) del subbloque.

Específicamente, la etapa S1102 involucra obtener un predictor refinado (por ejemplo, una señal de predicción  $l'(i, j)$ ) de una muestra actual en el subbloque con base en un valor de predicción delta (por ejemplo,  $\Delta l(i, j)$ ) de la muestra actual en el subbloque y un predictor (por ejemplo, una señal de predicción  $l(i, j)$ ) de la muestra actual en el subbloque.

En un posible diseño, la pluralidad de condiciones de decisión de flujo óptico incluye uno o más de los siguientes:

(a) Información de indicación (por ejemplo, `sps_prof_enabled_flag` o `sps_bdof_enabled_flag`) obtenida a través del análisis o derivación indica que PROF está habilitado para una secuencia, imagen, sector o grupo de mosaicos actual. Por ejemplo, `sps_prof_enabled_flag` o `sps_bdof_enabled_flag` = 1. Se puede entender que si en S1101 se utilizan condiciones de restricción para aplicar PROF en lugar de las condiciones de decisión de flujo óptico, esta condición puede convertirse en una condición de restricción para aplicar PROF como: (a)

Información de indicación que indica que PROF está deshabilitado para una secuencia, imagen, sector o grupo de mosaicos actual. Por ejemplo, `sps_prof_disabled_flag` o `sps_bdof_disabled_flag` = 1.

- 5 Información de indicación obtenida mediante el análisis de un parámetro establecido tal como SPS, PPS, una cabecera de sector, o una cabecera de grupo de mosaicos indica si PROF está habilitado para la secuencia, imagen, sector o grupo de mosaicos actual.

Específicamente, `sps_prof_enabled_flag` puede utilizarse para control, y sintaxis y semántica de `sps_prof_enabled_flag` son de la siguiente manera:

<code>seq_parameter_set_rbsp( ) {</code>	Descriptor
<code>...</code>	
<code>sps_affine_enabled_flag</code>	<code>u(1)</code>
<code>if( sps_affine_enabled_flag) {</code>	
<code>sps_affine_type_flag</code>	<code>u(1)</code>
<code>sps_prof_enabled_flag</code>	<code>u(1)</code>
<code>}</code>	
<code>...</code>	

10

`sps_prof_enabled_flag` igual a 0 especifica que el refinamiento de predicción flujo óptico para compensación de movimiento basada en afín está deshabilitado. `sps_prof_enabled_flag` igual a 1 especifica que el refinamiento de predicción flujo óptico para compensación de movimiento basada en afín está habilitado.

- 15 Como alternativa, `sps_bdof_enabled_flag` se reutiliza para control.

Se debería entender que, en esta realización, otra condición se deriva adicionalmente sobre la premisa de que se cumple la condición anterior (por ejemplo, un interruptor principal determina habilitar PROF). En otras palabras, si PROF está habilitado para la secuencia, imagen, sector o grupo de mosaicos actual, adicionalmente se determina si el bloque de imagen afín actual cumple con otra condición de decisión de flujo óptico descrita a continuación. Si PROF no está habilitado para la secuencia, imagen, sector o grupo de mosaicos actual, es innecesario determinar si el bloque de imagen afín actual cumple con otra condición de decisión de flujo óptico proporcionada en lo sucesivo.

20

- 25 (b) Información de indicación derivada (por ejemplo, la variable `fallbackModeTriggered`) indica que el bloque codificado afín actual va a dividirse. Por ejemplo, `fallbackModeTriggered` = 0. Se puede entender que si en S1101 se utilizan condiciones de restricción para aplicar PROF, en lugar de las condiciones de decisión de flujo óptico, la condición (b) puede convertirse en una condición de restricción para aplicar PROF como: (b) información de indicación derivada indica que el bloque codificado afín actual no está dividido. Por ejemplo, `fallbackModeTriggered` = 1.

30

Una variable `fallbackModeTriggered` se deriva con base en un parámetro afín, y si se utiliza o no PROF depende de `fallbackModeTriggered`. Cuando `fallbackModeTriggered` es 1, esto indica que el bloque codificado afín actual no va a dividirse. Cuando `fallbackModeTriggered` es 0, esto indica que el bloque codificado afín va a dividirse (por ejemplo, el bloque codificado afín se divide en subbloques, por ejemplo, subbloques 4x4). PROF va a utilizarse cuando el bloque codificado afín actual va a dividirse.

35

Específicamente, la variable `fallbackModeTriggered` puede derivarse utilizando el siguiente proceso:

- 40 La variable `fallbackModeTriggered` inicialmente se establece a 1, y se deriva adicionalmente de la siguiente manera:

- Las variables `bxWX4`, `bxHX4`, `bxWXh`, `bxHXh`, `bxWXv` y `bxHXv` se derivan de la siguiente manera:

$$\begin{aligned} \max W_4 &= \text{Max}(0, \text{Max}(4 * (2048 + d\text{HorX}), \\ &\text{Max}(4 * d\text{HorY}, 4 * (2048 + d\text{HorX}) + 4 * d\text{HorY}))) \quad (8-775) \end{aligned}$$

45

$$\begin{aligned} \min W_4 &= \text{Min}(0, \text{Min}(4 * (2048 + d\text{HorX}), \\ &\text{Min}(4 * d\text{HorY}, 4 * (2048 + d\text{HorX}) + 4 * d\text{HorY}))) \quad (8-775) \end{aligned}$$

$$\begin{aligned} \max H_4 &= \text{Max}(0, \text{Max}(4 * dVerX, \\ &\quad \text{Max}(4 * (2048 + dVerY), 4 * dVerX + 4 * (2048 + dVerY)))) \quad (8-775) \end{aligned}$$

$$\begin{aligned} \min H_4 &= \text{Min}(0, \text{Min}(4 * dVerX, \\ &\quad \text{Min}(4 * (2048 + dVerY), 4 * dVerX + 4 * (2048 + dVerY)))) \quad (8-775) \end{aligned}$$

$$5 \quad bxWX_4 = ((\max W_4 - \min W_4) \gg 11) + 9 \quad (8-775)$$

$$bxHX_4 = ((\max H_4 - \min H_4) \gg 11) + 9 \quad (8-775)$$

$$10 \quad bxWX_h = ((\text{Max}(0, 4 * (2048 + dHorX)) - \text{Min}(0, 4 * (2048 + dHorX))) \gg 11) + 9 \quad (8-775)$$

$$bxHX_h = ((\text{Max}(0, 4 * dVerX) - \text{Min}(0, 4 * dVerX)) \gg 11) + 9 \quad (8-775)$$

$$bxWX_v = ((\text{Max}(0, 4 * dVerY) - \text{Min}(0, 4 * dVerY)) \gg 11) + 9 \quad (8-775)$$

$$15 \quad bxHX_v = ((\text{Max}(0, 4 * (2048 + dHorY)) - \text{Min}(0, 4 * (2048 + dHorY))) \gg 11) + 9 \quad (8-775)$$

- Si `inter_pred_idc[xCb][yCb]` es igual a `PRED_BI` y `bxWX4 * bxHX4` es menor o igual a 225, `fallbackModeTriggered` se establece igual a 0.

20 - De otra manera, si ambos `bxWXh * bxHXh` es menor o igual a 165 y `bxWXv * bxHXv` es menor o igual a 165, `fallbackModeTriggered` se establece igual a 0.

(c) El bloque de imagen afín actual es un bloque de imagen afín de uni-predicción.

25 (d) El tamaño del subbloque en el bloque de imagen afín es mayor que  $N \times N$ , donde  $N = 4$ .

(e) El bloque de imagen afín actual es un bloque de imagen afín de uni-predicción y el tamaño del subbloque en el bloque de imagen afín es igual a  $N \times N$ , donde  $N = 4$ .

30 (f) El bloque de imagen afín actual es un bloque de imagen afín de bi-predicción y el tamaño del subbloque en el bloque de imagen afín es mayor que  $N \times N$ , donde  $N = 4$ .

El bloque de imagen afín actual es el bloque de codificación afín actual, y el hecho de que el bloque de imagen afín actual sea un bloque de imagen afín de uni-predicción se determina utilizando el siguiente método:

35 En el lado del codificador, se determina, con base en un criterio de tasa-distorsión RDO, que la uni-predicción se utiliza para el bloque de imagen afín actual.

40 El bloque de imagen afín actual es el bloque de decodificación afín actual, y el hecho de que el bloque de imagen afín actual sea un bloque de imagen afín de uni-predicción se determina utilizando el siguiente método:

45 en el lado del decodificador, en un modo AMVP, se utiliza información de indicación de dirección de predicción para indicar una dirección de uni-predicción (por ejemplo, solamente predicción hacia delante o solamente predicción hacia atrás), y la información de indicación de dirección de predicción se obtiene analizando un flujo de bits o a través de derivación; o

50 en el lado del decodificador, en un modo de fusión, información de movimiento candidata correspondiente a un índice candidato en una lista de candidatos incluye primera información de movimiento correspondiente a una primera lista de fotogramas de referencia, o información de movimiento candidata correspondiente a un índice candidato en una lista de candidatos incluye segunda información de movimiento correspondiente a una segunda lista de fotogramas de referencia.

55 En un posible diseño, la información de indicación de dirección de predicción incluye un elemento de sintaxis `inter_pred_idc[x0][y0]`, donde:

`inter_pred_idc[x0][y0] = PRED_L0`, utilizada para indicar predicción hacia delante;

`inter_pred_idc[x0][y0] = PRED_L1`, utilizada para indicar predicción hacia atrás; o

60 la información de indicación de dirección de predicción incluye `predFlagL0` y/o `predFlagL1`, donde:

predFlagL0= 1, y predFlagL1= 0, utilizadas para indicar predicción hacia delante;

predFlagL1= 1, y predFlagL0= 0, utilizadas para indicar predicción hacia atrás.

5

Se debería observar que las condiciones de decisión de flujo óptico (o condiciones de restricción para aplicar PROF) no se limitan a los ejemplos anteriores, y condiciones de decisión de flujo óptico adicionales o diferentes (o condiciones de restricción para aplicar PROF) pueden ser establecidas con base en diferentes escenarios de aplicación. Por ejemplo, las condiciones anteriores (c) a (f) pueden reemplazarse con otras condiciones, tal como una condición de decisión de flujo óptico: PROF puede aplicarse para un bloque codificado afín si todos los MV de punto de control del bloque codificado afín son diferentes entre sí o una condición de restricción para aplicar PROF: PROF no se va a aplicar para un bloque codificado afín si todos los MV de punto de control del bloque codificado afín son los mismos. Tal como una condición de decisión de flujo óptico: PROF puede aplicarse para un bloque codificado afín si la resolución de la imagen actual y la resolución de la imagen de referencia del bloque codificado afín son las mismas, por ejemplo  $RprConstraintsActive[X][refIdxLX]$  es igual a 0 o una condición de restricción para aplicar PROF: PROF no se va a aplicar para un bloque codificado afín si la resolución de la imagen actual y la resolución de la imagen de referencia del bloque codificado afín son diferentes una de otra, por ejemplo,  $RprConstraintsActive[X][refIdxLX]$  es igual a 1.

10

15

20

En un posible diseño, en la etapa S1102, la realización del procesamiento de flujo óptico (refinamiento de predicción con flujo óptico, PROF) para uno o más subbloques (por ejemplo, cada subbloque o un subbloque actual) en el bloque de imagen afín actual, para obtener un valor de predicción delta (por ejemplo,  $\Delta(i, j)$ ) de uno o más subbloques (por ejemplo, cada subbloque o el subbloque actual) en el bloque de imagen afín actual puede incluir las siguientes etapas:

25

Etapla 1. Obtener una segunda matriz de predicción con base en información de movimiento (por ejemplo, un vector de movimiento) de un subbloque actual en el bloque de imagen afín actual.

30

Por ejemplo, un bloque de predicción  $(M+2)*(N+2)$  (concretamente, la segunda matriz de predicción) se obtiene a través de interpolación con base en un vector de movimiento de un subbloque  $M \times N$ . Se proporcionaron arriba diferentes implementaciones.

35

Etapla 2. Calcular una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical con base en la segunda matriz de predicción, donde el tamaño de la segunda matriz de predicción es mayor o igual a tamaños de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical.

40

Etapla 3. Calcular un valor de predicción delta ( $\Delta(i, j)$ ) de la muestra actual en el subbloque con base en un valor de gradiente de predicción horizontal de la muestra actual en el subbloque en la matriz de gradiente de predicción horizontal, un valor de gradiente de predicción vertical de la muestra actual en la matriz de gradiente de predicción vertical, y una diferencia entre el vector de movimiento de la muestra actual del subbloque actual y el vector de movimiento de la muestra central del subbloque.

45

De manera correspondiente, en la etapa S1102, la obtención de un valor de muestra de predicción refinado (por ejemplo, una señal de predicción  $I'(i, j)$ ) del subbloque con base en el valor de predicción delta (por ejemplo,  $\Delta(i, j)$ ) del subbloque y un valor de muestra de predicción (por ejemplo, una señal de predicción  $I(i, j)$ ) del subbloque puede incluir:

50

obtener un valor de muestra de predicción refinado (por ejemplo, una señal de predicción  $I'(i, j)$ ) de la muestra actual con base en un valor de predicción delta (por ejemplo,  $\Delta(i, j)$ ) de la muestra actual en el subbloque y un valor de muestra de predicción (por ejemplo, una señal de predicción  $I(i, j)$ ) de la muestra actual.

55

Se debería entender que los valores de muestra de predicción (por ejemplo, la señal de predicción  $I(i, j)$ ) del subbloque puede ser un bloque de predicción  $M \times N$  en un bloque de predicción  $(M+2)*(N+2)$ .

60

Para la etapa 3, en una implementación, diferencias de vector de movimiento entre vectores de movimiento de diferentes muestras en el subbloque actual y un vector de movimiento de la muestra central del subbloque son diferentes. En otra implementación, la diferencia de vector de movimiento entre un vector de movimiento de una unidad de muestra actual (por ejemplo, un bloque de muestra  $2 \times 2$ ) que contiene la muestra actual y el vector de movimiento de la muestra central del subbloque se utiliza como una diferencia de vector de movimiento entre el vector de movimiento de la muestra actual del subbloque actual y el vector de movimiento de la muestra central del subbloque. En otras palabras, para equilibrar las sobrecargas de procesamiento y la precisión de predicción, asumiendo que ambas muestras A y B se incluyen en la unidad de muestra actual, una diferencia de vector de movimiento entre un vector de movimiento de una unidad de muestra actual (por ejemplo, un bloque de muestra  $2 \times 2$ ) y un vector de movimiento de una muestra central del subbloque puede utilizarse como una diferencia de vector de movimiento entre el vector de movimiento de la muestra A en el

65



- subbloque y el vector de movimiento de la muestra central del subbloque; también, una diferencia de vector de movimiento entre el vector de movimiento de la unidad de muestra actual y el vector de movimiento de la muestra central del subbloque puede utilizarse como una diferencia de vector de movimiento entre el vector de movimiento de la muestra B en el subbloque y el vector de movimiento de la muestra central del subbloque.
- 5 En una implementación, la segunda matriz de predicción en la etapa 1 anterior se representa por  $I_1(p, q)$ , donde un intervalo de valores de  $p$  es  $[-1, sbW]$ , y un intervalo de valores de  $q$  es  $[-1, sbH]$ ;
- 10 la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$ , donde un intervalo de valores de  $i$  es  $[0, sbW-1]$ , y un intervalo de valores de  $j$  es  $[0, sbH-1]$ ; y
- La matriz de gradiente de predicción vertical se representa por  $Y(i, j)$ , donde un intervalo de valores de  $i$  es  $[0, sbW-1]$ , y un intervalo de valores de  $j$  es  $[0, sbH-1]$ ; donde
- 15  $sbW$  representa una anchura del subbloque actual en el bloque de imagen afín actual,  $sbH$  representa una altura del subbloque actual en el bloque de imagen afín actual, y  $(x, y)$  representa coordenadas de ubicación de cada muestra (también referida como una muestra) en el subbloque actual en el bloque de imagen afín actual, y el elemento ubicado en  $(x, y)$  puede corresponder al elemento ubicado en  $(i, j)$ .
- 20 En otro posible diseño, en la etapa 1102, la realización del procesamiento de flujo óptico (refinamiento de predicción con flujo óptico, PROF) para uno o más subbloques (por ejemplo, cada subbloque) en el bloque de imagen afín actual, para obtener un valor de predicción delta (también denominado un valor de compensación de predictor, por ejemplo,  $\Delta I(i, j)$ ) de uno o más subbloques (por ejemplo, cada subbloque) en el bloque de imagen afín actual incluye, tal como se muestra en la FIG. 12:
- 25 S1202. obtener o generar una segunda matriz de predicción con base en una primera matriz de predicción, la primera matriz de predicción (por ejemplo, una primera señal de predicción  $I(i, j)$  o una predicción  $4 \times 4$ ) del subbloque (por ejemplo, cada subbloque) corresponde a valores de muestra de predicción del subbloque actual. La compensación de movimiento afín basada en subbloque para un subbloque actual del bloque codificado afín se realiza para obtener valores de muestra de predicción del subbloque actual del bloque codificado afín, tal como se muestra en la FIG. 9A.
- 30 S1203. Calcular una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical con base en una segunda matriz de predicción, donde un tamaño de la segunda matriz de predicción es mayor o igual a un tamaño de la primera matriz de predicción, y el tamaño de la segunda matriz de predicción es mayor o igual a tamaños de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical; y
- 35 S1204. Calcular una matriz de valor de predicción delta (por ejemplo,  $\Delta I(i, j)$  de una señal de predicción) del subbloque con base en la matriz de gradiente de predicción horizontal, la matriz de gradiente de predicción vertical, y una diferencia de vector de movimiento entre el vector de movimiento de la unidad de muestra actual (por ejemplo, una muestra actual o un bloque de muestras actual, tal como un bloque de muestra  $2 \times 2$ ) del subbloque y el vector de movimiento de la muestra central del subbloque; y
- 40 la etapa de obtener un valor de muestra de predicción refinado (por ejemplo, una señal de predicción  $I'(i, j)$ ) del subbloque con base en el valor de predicción delta (por ejemplo,  $\Delta I(i, j)$ ) del subbloque y un valor de muestra de predicción (por ejemplo, una señal de predicción  $I(i, j)$ ) del subbloque incluye:
- 45 S1205. Obtener una tercera matriz de predicción refinada (por ejemplo, una señal de predicción  $I'(i, j)$ ) del subbloque con base en la matriz de valor de predicción delta (por ejemplo,  $\Delta I(i, j)$ ) y la primera matriz de predicción (por ejemplo, la señal de predicción  $I(i, j)$ ).
- 50 Se debería entender que,  $I(i, j)$  en esta memoria representa un valor de muestra de predicción (por ejemplo, un valor de muestra de predicción original obtenido a través de compensación de movimiento) de la muestra actual en el subbloque actual,  $\Delta I(i, j)$  representa un valor de predicción delta de la muestra actual en el subbloque actual, e  $I'(i, j)$  representa un valor de muestra de predicción refinado de la muestra actual en el subbloque actual. Por ejemplo, el valor de muestra de predicción original + el valor de predicción delta = el valor de muestra de predicción refinado. Se debería entender que la obtención de valores de muestra de predicción refinados de una pluralidad de muestras (por ejemplo, todas las muestras) en el subbloque actual es equivalente a la obtención de valores de muestra de predicción refinados del subbloque actual.
- 55 En diferentes implementaciones posibles, un valor de gradiente puede calcularse muestra por muestra y un valor de predicción delta puede calcularse muestra por muestra. Alternativamente, se puede obtener una matriz de valor de gradiente y después se puede calcular un valor de predicción delta. Esto no queda limitado en esta solicitud. En una implementación alternativa, la primera matriz de predicción y la segunda matriz de predicción representan una misma matriz de predicción.
- 60
- 65

Para un caso en el que el tamaño de la segunda matriz de predicción es igual al tamaño de la primera matriz de predicción y el tamaño de la segunda matriz de predicción es igual a los tamaños de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical, en una posible implementación, una matriz de gradiente  $(w-2)*(h-2)$  se calcula utilizando una matriz de predicción  $w*h$ , y la matriz de gradiente es rellena para obtener un tamaño de  $w*h$ , donde  $w*h$  representa un tamaño del subbloque actual. Por ejemplo, la primera matriz de predicción y la segunda matriz de predicción son matrices de predicción cuyos tamaños son  $w*h$  o la primera matriz de predicción y la segunda matriz de predicción representan una misma matriz de predicción.

Tal como se muestra en la FIG. 11B, otra realización de esta solicitud proporciona otro método para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afín, incluyendo los siguientes etapas:

S1110. Se determina si una pluralidad de condiciones de decisión de flujo óptico se cumplen o satisfechas o no. Aquí las condiciones de decisión de flujo óptico se refieren a las condiciones que permiten la aplicación de PROF.

S1111. Si la pluralidad de condiciones de decisión de flujo óptico se cumplen, un primer indicador (por ejemplo, applyProfFlag) se establece igual a verdadero, y la realización de un proceso de refinamiento de predicción con flujo óptico (PROF) para un subbloque actual del bloque codificado afín, para obtener valores de muestra de predicción refinados del subbloque actual del bloque codificado afín. En la etapa S1111, se realiza procesamiento de flujo óptico (refinamiento de predicción con flujo óptico, PROF) para uno o más subbloques (por ejemplo, cada subbloque) en el bloque de imagen afín actual, para obtener un valor de predicción delta (también denominado valor de compensación de predictor, por ejemplo,  $\Delta l(i, j)$ ) de uno o más subbloques (por ejemplo, cada subbloque) en el bloque de imagen afín actual.

En la etapa S1111, un valor de muestra de predicción refinado (por ejemplo, una señal de predicción  $l'(i, j)$ ) del subbloque se obtiene con base en el valor de predicción delta (por ejemplo,  $\Delta l(i, j)$ ) y un valor de muestra de predicción (por ejemplo, una señal de predicción  $l(i, j)$ ) del subbloque.

Se debería entender que,  $l(i, j)$  en esta memoria representa un valor de muestra de predicción (por ejemplo, un valor de muestra de predicción original obtenido a través de compensación de movimiento) de la muestra actual en el subbloque actual,  $\Delta l(i, j)$  representa un valor de predicción delta de la muestra actual en el subbloque actual, e  $l'(i, j)$  representa un valor de muestra de predicción refinado de la muestra actual en el subbloque actual. Por ejemplo, el valor de muestra de predicción original + el valor de predicción delta = el valor de muestra de predicción refinado. Se debería entender que la obtención de valores de muestra de predicción refinados de una pluralidad de muestras (por ejemplo, todas las muestras) en el subbloque actual es equivalente a la obtención de valores de muestra de predicción refinados del subbloque actual.

Se puede entender que cuando se generan los valores de muestra de predicción refinados de cada subbloque del bloque codificado afín, los valores de muestra de predicción refinados del bloque codificado afín se generan naturalmente. S1113, cuando al menos una de la pluralidad de condiciones de decisión de flujo óptico no se cumple o satisface, un primer indicador (por ejemplo, applyProfFlag) se establece igual a falso, y el proceso de PROF se omite.

Se puede entender que si se utilizan condiciones de restricción para aplicar PROF para determinar si se aplica PROF, la etapa S1110 se cambiará para determinar si ninguna de una pluralidad de las condiciones de restricción para aplicar PROF se cumple. En este caso, la etapa S1111 se cambiará a: si ninguna de una pluralidad de las condiciones de restricción para aplicar PROF se cumple o satisface, un primer indicador (por ejemplo, applyProfFlag) se establece igual a verdadero, y realizando un proceso de refinamiento de predicción con flujo óptico (PROF) para un subbloque actual del bloque codificado afín, para obtener valores de muestra de predicción refinados del subbloque actual del bloque codificado afín. Por consiguiente, la etapa S1113 se cambiará a: si se cumple al menos una de la pluralidad de las condiciones de restricción para aplicar PROF, el primer indicador (por ejemplo, applyProfFlag) se establece igual a falso, y el proceso PROF se omite.

En una implementación, si el primer indicador (por ejemplo, applyProfFlag) es un primer valor (por ejemplo, 1), realizar procesamiento de flujo óptico (por ejemplo, PROF) para uno o más subbloques (por ejemplo, cada subbloque) en el bloque de imagen afín actual; o,

De otra manera, si el primer indicador (por ejemplo, applyProfFlag) es un segundo valor (por ejemplo, 0), omitir la realización del procesamiento de flujo óptico (por ejemplo, PROF) para uno o más subbloques (por ejemplo, cada subbloque) en el bloque de imagen afín actual.

En una implementación, un valor del primer indicador depende de si se cumplen o no las condiciones de decisión de flujo óptico, donde las condiciones de decisión de flujo óptico incluyen uno o más de los siguientes:

- primera información de indicación (por ejemplo, `sps_prof_enabled_flag` o `sps_bdof_enabled_flag`) se utiliza para indicar que PROF está habilitado para una unidad de imagen actual. Se debería observar que la unidad de imagen actual en esta memoria puede ser, por ejemplo, una secuencia actual, una imagen actual, un sector actual, o un grupo de mosaicos actual. Estos ejemplos de la unidad de imagen actual no son una limitación.
- segunda información de indicación (por ejemplo, `fallbackModeTriggered`) se utiliza para indicar la división del bloque de imagen afín actual;
- el bloque de imagen afín actual es un bloque de imagen afín de uni-predicción;
- un tamaño del subbloque en el bloque de imagen afín es mayor que  $N \times N$ , en donde  $N = 4$ ;
- el bloque de imagen afín actual es un bloque de imagen afín de uni-predicción y un tamaño del subbloque en el bloque de imagen afín es igual a  $N \times N$ , en donde  $N = 4$ ; o
- el bloque de imagen afín actual es un bloque de imagen afín de bi-predicción y un tamaño del subbloque en el bloque de imagen afín es mayor que  $N \times N$ , en donde  $N = 4$ .
- Se debería observar que esta solicitud incluye, pero no se limita a las condiciones de decisión de flujo óptico anteriores, y condiciones de decisión de flujo óptico adicionales o diferentes pueden ser establecidas con base en diferentes escenarios de aplicación.
- En esta realización de esta solicitud, por ejemplo, cuando se cumplen todas las siguientes condiciones, `applyProfFlag` se establece a 1:
- `sps_prof_enabled_flag == 1`
  - `fallbackModeTriggered == 0`
  - `inter_pred_idc[x0][y0] = PRED_L0 o PRED_L1`; (o `predFlagL0 = 1`, `predFlagL1 = 0`; o `predFlagL1 = 1`, `predFlagL0 = 0`)
  - Otras condiciones
- En otra realización, por ejemplo, cuando se cumplen todas las siguientes condiciones, `applyProfFlag` se establece a 1:
- `sps_prof_enabled_flag == 1`
  - `fallbackModeTriggered == 0`
  - otras condiciones
- Se puede entender que si se usan condiciones de restricción para aplicar PROF en S1110 en lugar las condiciones de decisión de flujo óptico, cuando no se cumple ninguna de todas las condiciones de restricción siguientes, `applyProfFlag` se establece a 1:
- `sps_prof_disabled_flag == 1`
  - `fallbackModeTriggered == 1`
  - otras condiciones
- Se debería entender que, para detalles referentes a una entidad de ejecución de las etapas en el método de predicción proporcionado en esta realización de esta solicitud y extensiones y variaciones de estas etapas, hacer referencia a las descripciones anteriores de los métodos correspondientes. Por brevedad, los detalles no se describen en esta memoria descriptiva nuevamente.
- Otra realización de esta solicitud proporciona además otro proceso PROF, que incluye:
- obtener una primera matriz de predicción del bloque  $M \times N$  con base en información de movimiento (por ejemplo, un vector de movimiento) de una pluralidad de subbloques en un bloque de imagen afín actual; por ejemplo, el bloque  $M \times N$  es  $16 \times 16$ , tal como se muestra en la FIG. 9E. Por ejemplo, el bloque  $16 \times 16$  (o la ventana  $16 \times 16$ ) incluye 16 subbloques  $4 \times 4$ .

- calcular una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical con base en una segunda matriz de predicción, donde un tamaño de la segunda matriz de predicción es mayor o igual a un tamaño de la primera matriz de predicción, y el tamaño de la segunda matriz de predicción es mayor o igual a tamaños de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical; y
- calcular una matriz de valor de predicción delta (por ejemplo,  $\Delta l(i, j)$  de una señal de predicción) del bloque  $M \times N$  con base en la matriz de gradiente de predicción horizontal, la matriz de gradiente de predicción vertical, y una diferencia de vector de movimiento entre un vector de movimiento de la unidad de píxel actual (por ejemplo, un píxel actual o un bloque de píxel actual, tal como un bloque de píxel  $2 \times 2$ ) en el bloque  $M \times N$  y un vector de movimiento de un píxel central del bloque  $M \times N$ ; y
- obtener una tercera matriz de predicción refinada (por ejemplo, una señal de predicción  $l'(i, j)$ ) del bloque  $M \times N$  con base en la matriz de valor de predicción delta (por ejemplo,  $\Delta l(i, j)$ ) y la primera matriz de predicción (por ejemplo, la señal de predicción  $l(i, j)$ ).
- En otro posible diseño, la primera matriz de predicción se representa por  $l_1(i, j)$ , donde un intervalo de valores de  $i$  es  $[0, \text{size\_w}-1]$ , y un intervalo de valores de  $j$  es  $[0, \text{size\_h}-1]$ ;
- la segunda matriz de predictor se representa por  $l_2(i, j)$ , donde un intervalo de valores de  $i$  es  $[-1, \text{size\_w}]$ , y un intervalo de valores de  $j$  es  $[-1, \text{size\_h}]$ , donde  $\text{size\_w} = \min(W, m)$ ,  $\text{size\_h} = \min(H, m)$ , y  $m = 16$ ;
- la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$ , donde un intervalo de valores de  $i$  es  $[0, \text{size\_w}-1]$ , y un intervalo de valores de  $j$  es  $[0, \text{size\_h}-1]$ ; y
- la matriz de gradiente de predicción vertical se representa por  $Y(i, j)$ , donde un intervalo de valores de  $i$  es  $[0, \text{size\_w}-1]$ , y un intervalo de valores de  $j$  es  $[0, \text{size\_h}-1]$ ; donde
- $W$  representa una anchura del bloque de imagen afín actual,  $H$  representa una altura del bloque de imagen afín actual, y  $(x, y)$  representa coordenadas de ubicación de cada muestra en el bloque de imagen afín actual.
- Se debería entender que, tal como se muestra en la FIG. 9E, en algunos ejemplos, un bloque de imagen afín se divide de manera implícita en bloques  $16 \times 16$ , y se calcula una matriz de gradiente para cada bloque  $16 \times 16$ . De manera correspondiente, la segunda matriz de predicción se representa por  $l_2(i, j)$ , donde un intervalo de valores de  $i$  es  $[-1, \text{size\_w}]$ , y un intervalo de valores de  $j$  es  $[-1, \text{size\_h}]$ , donde  $\text{size\_w} = \min(w, m)$ ,  $\text{size\_h} = \min(h, m)$ , y  $m = 16$ . La matriz de gradiente de predicción horizontal se representa por  $X(i, j)$ , donde un intervalo de valores de  $i$  es  $[0, \text{size\_w}-1]$ , y un intervalo de valores de  $j$  es  $[0, \text{size\_h}-1]$ . La matriz de gradiente de predicción vertical se representa por  $Y(i, j)$ , donde un intervalo de valores de  $i$  es  $[0, \text{size\_w}-1]$ , y un intervalo de valores de  $j$  es  $[0, \text{size\_h}-1]$ .
- En otro diseño posible, el cálculo de una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical (por ejemplo, valores de gradiente  $\text{size\_w} \times \text{size\_h}$ ) con base en una segunda matriz de predicción (por ejemplo, una señal de predicción  $(\text{size\_w}+2) \times (\text{size\_h}+2)$ ) incluye:
- calcular la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical con base en la segunda matriz de predicción, donde la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical incluyen: una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical del subbloque, respectivamente, donde
- la segunda matriz de predicción se representa por  $l_2(i, j)$ , donde un intervalo de valores de  $i$  es  $[-1, \text{size\_w}]$ , y un intervalo de valores de  $j$  es  $[-1, \text{size\_h}]$ , donde  $\text{size\_w} = \min(W, m)$ ,  $\text{size\_h} = \min(H, m)$ , y  $m = 16$ ;
- la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$ , donde un intervalo de valores de  $i$  es  $[0, \text{size\_w}-1]$ , y un intervalo de valores de  $j$  es  $[0, \text{size\_h}-1]$ ; y
- la matriz de gradiente de predicción vertical se representa por  $Y(i, j)$ , donde un intervalo de valores de  $i$  es  $[0, \text{size\_w}-1]$ , y un intervalo de valores de  $j$  es  $[0, \text{size\_h}-1]$ ; donde
- $W$  representa una anchura del bloque de imagen afín actual,  $H$  representa una altura del bloque de imagen afín actual, y  $(i, j)$  representa coordenadas de ubicación de cada muestra en el bloque de imagen afín actual.
- Se puede aprender a partir de la descripción anterior que el bloque de imagen afín actual se divide implícitamente en bloques  $16 \times 16$ , y se calcula una matriz de gradiente para cada bloque  $16 \times 16$ . Se debería entender que  $m = 16$  se utiliza simplemente como un ejemplo en esta memoria y no debería interpretarse como una limitación. Se pueden utilizar otros diversos valores de  $m$ , tal como  $m = 32$ .

En un posible diseño, el método se utiliza para bi-predicción, y la información de movimiento incluye primera información de movimiento correspondiente a una primera lista de fotogramas de referencia y segunda información de movimiento correspondiente a una segunda lista de fotogramas de referencia; y

5 la primera matriz de predicción incluye (es) una primera matriz de predicción inicial o una segunda matriz de predicción inicial, donde la primera matriz de predicción inicial se obtiene con base en la primera información de movimiento, y la segunda matriz de predicción inicial se obtiene con base en la segunda información de movimiento;

10 la matriz de gradiente de predicción horizontal incluye (es) una primera matriz de gradiente de predicción horizontal o una segunda matriz de gradiente de predicción horizontal, donde la primera matriz de gradiente de predicción horizontal se obtiene a través del cálculo con base en una primera matriz de predicción inicial extendida, y la segunda matriz de gradiente de predicción horizontal se obtiene a través del cálculo con base en una segunda matriz de predicción inicial extendida;

15 la matriz de gradiente de predicción vertical incluye (es) una primera matriz de gradiente de predicción vertical o una segunda matriz de gradiente de predicción vertical, donde la primera matriz de gradiente de predicción vertical se obtiene a través del cálculo con base en una primera matriz de predicción inicial extendida y la segunda matriz de gradiente de predicción vertical se obtiene a través de cálculo con base en una segunda matriz de predicción inicial extendida; y

20 la matriz de valor de predicción delta incluye (es) una primera matriz de valor de predicción delta correspondiente a una primera lista de fotogramas de referencia o una segunda matriz de valor de predicción delta correspondiente a la segunda lista de fotogramas de referencia, donde la primera matriz de valor de predicción delta se obtiene a través del cálculo con base en la primera matriz de gradiente de predicción horizontal, la primera matriz de gradiente de predicción vertical, y una primera diferencia de vector de movimiento (por ejemplo, una diferencia de vector de movimiento de avance) de cada unidad de muestra en el subbloque con relación a la muestra central del subbloque, y la segunda matriz de valor de predicción delta se obtiene a través de cálculo con base en la segunda matriz de gradiente de predicción horizontal, la segunda matriz de gradiente de predicción vertical, y una segunda diferencia de vector de movimiento (por ejemplo, una diferencia de vector de movimiento de retroceso) de cada unidad de muestra en el subbloque con relación a la muestra central del subbloque.

25 En un posible diseño, el método se utiliza para bi-predicción, y la información de movimiento incluye primera información de movimiento correspondiente a una primera lista de fotogramas de referencia y segunda información de movimiento correspondiente a una segunda lista de fotogramas de referencia; y

30 la primera matriz de predicción incluye una primera matriz de predicción inicial y una segunda matriz de predicción inicial, donde la primera matriz de predicción inicial se obtiene con base en la primera información de movimiento, y la segunda matriz de predicción inicial se obtiene con base en la segunda información de movimiento;

35 la matriz de gradiente de predicción horizontal incluye una primera matriz de gradiente de predicción horizontal y una segunda matriz de gradiente de predicción horizontal, la primera matriz de gradiente de predicción horizontal se obtiene a través del cálculo con base en una primera matriz de predicción inicial extendida, y la segunda matriz de gradiente de predicción horizontal se obtiene a través del cálculo con base en una segunda matriz de predicción inicial extendida;

40 la matriz de gradiente de predicción vertical incluye una primera matriz de gradiente de predicción vertical y una segunda matriz de gradiente de predicción vertical, donde la primera matriz de gradiente de predicción vertical se obtiene a través del cálculo con base en una primera matriz de predicción inicial extendida, y la segunda matriz de gradiente de predicción vertical se obtiene a través del cálculo con base en una segunda matriz de predicción inicial extendida; y

45 la matriz de valor de predicción delta incluye una primera matriz de valor de predicción delta correspondiente a la primera lista de fotogramas de referencia y una segunda matriz de valor de predicción delta correspondiente a la segunda lista de fotogramas de referencia, donde la primera matriz de valor de predicción delta se obtiene a través del cálculo con base en la primera matriz de gradiente de predicción horizontal, la primera matriz de gradiente de predicción vertical, y una primera diferencia de vector de movimiento (por ejemplo, una diferencia de vector de movimiento de avance) de cada unidad de muestra en el subbloque con relación a la muestra central del subbloque, y la segunda matriz de valor de predicción delta se obtiene a través del cálculo con base en la segunda matriz de gradiente de predicción horizontal, la segunda matriz de gradiente de predicción vertical, y una segunda diferencia de vector de movimiento (por ejemplo, una diferencia de vector de movimiento de retroceso) de cada unidad de muestra en el subbloque con relación a la muestra central del subbloque.

En un posible diseño, el método se utiliza para uni-predicción;

la información de movimiento incluye primera información de movimiento correspondiente a una primera lista de fotogramas de referencia o segunda información de movimiento correspondiente a una segunda lista de fotogramas de referencia; y

la primera matriz de predicción incluye (es) una primera matriz de predicción inicial o una segunda matriz de predicción inicial, donde la primera matriz de predicción inicial se obtiene con base en la primera información de movimiento, y la segunda matriz de predicción inicial se obtiene con base en la segunda información de movimiento.

En un posible diseño, el método se utiliza para bi-predicción;

la información de movimiento incluye primera información de movimiento correspondiente a una primera lista de fotogramas de referencia y segunda información de movimiento correspondiente a una segunda lista de fotogramas de referencia; y

la primera matriz de predicción incluye la primera matriz de predicción inicial y la segunda matriz de predicción inicial, donde la primera matriz de predicción inicial se obtiene con base en la primera información de movimiento, y la segunda matriz de predicción inicial se obtiene con base en la segunda información de movimiento; y

la obtención de una matriz de predicción del subbloque con base en la información de movimiento del subbloque incluye:

realizar la suma ponderada de valores de muestra en una misma ubicación en la primera matriz de predicción inicial y la segunda matriz de predicción inicial, para obtener la matriz de predicción del subbloque. Se debería entender que, antes que se ejecute la suma ponderada en esta memoria, valores de muestra en la primera matriz de predicción inicial y la segunda matriz de predicción inicial pueden refinarse de manera separada.

En otro posible diseño, el proceso PROF se describe como en las siguientes cuatro etapas.

Etapla 1) La compensación de movimiento afín basada en subbloque se realiza para generar predicción de subbloque  $I(i,j)$ . Por ejemplo,  $i$  tiene un valor de  $[0, \text{subW}+1]$  o  $[-1, \text{subW}]$ , y  $j$  tiene un valor de  $[0, \text{subH}+1]$  o  $[-1, \text{subH}]$ . Se puede entender que la muestra superior-izquierda (o el origen de la coordenadas) se ubica en  $(1, 1)$  debido a que  $i$  tiene un valor de  $[0, \text{subW}+1]$  y  $j$  tiene un valor de  $[0, \text{subH}+1]$ ; mientras que, la muestra superior-izquierda se ubica en  $(0, 0)$  debido a que  $i$  tiene un valor de  $[-1, \text{subW}]$  y  $j$  tiene un valor de  $[-1, \text{subH}]$ .

Etapla 2) Los gradientes espaciales  $g_x(i, j)$  y  $g_y(i, j)$  de la predicción de subbloque son calculados en cada ubicación de muestra utilizando un filtro de 3-derivaciones  $[-1, 0, 1]$ .

$$g_x(i, j) = I(i + 1, j) - I(i - 1, j)$$

$$g_y(i, j) = I(i, j + 1) - I(i, j - 1)$$

La predicción de subbloque es extendida por una muestra en cada lado para el cálculo de gradiente. Para reducir el ancho de banda de memoria y la complejidad, las muestras en los bordes extendidos son copiadas desde la posición de muestra de entero más cercana en la imagen de referencia. Por lo tanto, se evita interpolación adicional para la región de relleno.

Etapla 3) El refinamiento de predicción de luminancia se calcula a través de la ecuación de flujo óptico.

$$\Delta I(i, j) = g_x(i, j) * \Delta v_x(i, j) + g_y(i, j) * \Delta v_y(i, j)$$

donde  $\Delta v(i, j)$  es la diferencia entre el MV de muestra calculado para la ubicación de muestra  $(i, j)$ , denotado por  $v(i, j)$ , y el MV de subbloque para el subbloque al que pertenece la muestra  $(i, j)$ , como se muestra en la Figura 10.

En otras palabras, se calcula un MV de cada muestra central  $4 \times 4$ , y después se calcula un MV de cada muestra del subbloque. La diferencia  $\Delta v(i, j)$  entre el MV de cada muestra y el MV de la muestra central se puede obtener.

Debido a que los parámetros de modelo afín y la ubicación de muestra con relación al centro del subbloque no son cambiados de subbloque a subbloque,  $\Delta v(i, j)$  puede calcularse para el primer subbloque, y reutilizarse para otros subbloques en la misma CU. Asumir que  $x$  e  $y$  son la compensación horizontal y vertical desde la

ubicación de muestra al centro del subbloque,  $\Delta v(x, y)$  puede derivarse a través de la siguiente ecuación,

$$\begin{cases} \Delta v_x(x, y) = c * x + d * y \\ \Delta v_y(x, y) = e * x + f * y \end{cases}$$

5 Para modelo afín de 4 parámetros,

$$\begin{cases} c = f = \frac{v_{1x} - v_{0x}}{w} \\ e = -d = \frac{v_{1y} - v_{0y}}{w} \end{cases}$$

Para modelo afín de 6 parámetros,

$$\begin{cases} c = \frac{v_{1x} - v_{0x}}{w} \\ d = \frac{v_{2x} - v_{0x}}{h} \\ e = \frac{v_{1y} - v_{0y}}{w} \\ f = \frac{v_{2y} - v_{0y}}{h} \end{cases}$$

donde  $(v_{0x}, v_{0y})$ ,  $(v_{1x}, v_{1y})$ ,  $(v_{2x}, v_{2y})$  son los vectores de movimiento de punto de control superior-izquierdo, superior-derecho e inferior-izquierdo,  $w$  y  $h$  son la anchura y la altura de la CU.

15 Etapa 4) Finalmente, el refinamiento de predicción de luminancia se añade a la predicción de subbloque  $I(i, j)$ . La predicción final  $I'$  se genera como se muestra en la siguiente ecuación.

$$I'(i, j) = I(i, j) + \Delta I(i, j)$$

20 La Fig. 15 ilustra un aparato 1500 para refinamiento de predicción con flujo óptico (PROF) para un bloque codificado afín según otro aspecto de la divulgación. En un ejemplo, el aparato 1500 comprende:

25 una unidad de determinación 1501 configurada para determinar que ninguna de la pluralidad de condiciones de restricción para aplicar PROF se cumple;

30 una unidad de procesamiento de predicción 1503 configurada para realizar un proceso de refinamiento de predicción con flujo óptico, PROF para un subbloque actual del bloque codificado afín, para obtener valores de muestra de predicción refinados del subbloque actual del bloque codificado afín. Se puede entender que cuando se generan los valores de muestra de predicción refinados de cada subbloque del bloque codificado afín, los valores de muestra de predicción refinados del bloque codificado afín se generan naturalmente.

En otro ejemplo, el aparato 1500 comprende:

35 una unidad de determinación 1501 configurada para determinar que se cumple una pluralidad de condiciones de decisión de flujo óptico; aquí, la pluralidad de condiciones de decisión de flujo óptico se refiere a las condiciones que permiten la aplicación de PROF.

40 una unidad de procesamiento de predicción 1503 configurada para realizar un proceso PROF para un subbloque actual del bloque codificado afín, para obtener valores de muestra de predicción refinados del subbloque actual del bloque codificado afín. Se puede entender que cuando se generan los valores de muestra de predicción refinados de cada subbloque del bloque codificado afín, los valores de muestra de predicción refinados del bloque codificado afín se generan naturalmente.

45 De manera correspondiente, en un ejemplo, una estructura ejemplar del aparato 1500 puede ser correspondiente al codificador 20 en la FIG. 2. En otro ejemplo, una estructura ejemplar del aparato 1500 puede ser correspondiente al decodificador 30 en la FIG. 3.

50 En otro ejemplo, una estructura ejemplar del aparato 1500 puede ser correspondiente a la unidad de interpredicción 244 en la FIG. 2. En otro ejemplo, una estructura ejemplar del aparato 1500 puede ser correspondiente a la unidad de interpredicción 344 en la FIG. 3.

55 Se puede entender que la unidad de determinación y la unidad de procesamiento de predicción (correspondiente al módulo de interpredicción) en el codificador 20 o el decodificador 30 proporcionados en esta realización de esta solicitud es una entidad funcional para implementar diversas etapas de ejecución

incluidos en el método correspondiente anterior, es decir, tiene una entidad funcional para implementar completamente etapas en el método en esta solicitud y extensiones y variaciones de estas etapas. Para detalles, hacer referencia a las descripciones anteriores del método correspondiente. Por brevedad, en esta memoria no se describen de nuevo detalles..

5

A continuación se proporciona una explicación de las aplicaciones del método de codificación así como del método de decodificación como se muestra en las realizaciones antes mencionadas, y el sistema que las utiliza.

La FIG. 16 es un diagrama de bloques que muestra un sistema de suministro de contenido 3100 para realizar el servicio de distribución de contenido. Este sistema de suministro de contenido 3100 incluye el dispositivo de captura 3102, el dispositivo terminal 3106, y opcionalmente incluye la pantalla 3126. El dispositivo de captura 3102 se comunica con el dispositivo terminal 3106 sobre el enlace de comunicación 3104. El enlace de comunicación puede incluir el canal de comunicación 13 como se ha descrito anteriormente. El enlace de comunicación 3104 incluye, pero no se limita a WIFI, Ethernet, Cable, inalámbrico (3G/4G/5G), USB, o cualquier tipo de combinación de los mismos, o similar.

El dispositivo de captura 3102 genera datos, y puede codificar los datos a través del método de codificación como se muestra en las realizaciones anteriores. Alternativamente, el dispositivo de captura 3102 puede distribuir los datos a un servidor de transmisión por secuencias (que no se muestra en las Figuras), y el servidor codifica los datos y transmite los datos codificados al dispositivo terminal 3106. El dispositivo de captura 3102 incluye pero no se limita a una cámara, teléfono inteligente o Pad, ordenador o portátil, sistema de conferencias de vídeo, PDA, dispositivo montado en vehículo, o una combinación de cualquiera de los anteriores, o similar. Por ejemplo, el dispositivo de captura 3102 puede incluir el dispositivo fuente 12 como se ha descrito antes. Cuando los datos incluyen vídeo, el codificador de vídeo 20 incluido en el dispositivo de captura 3102 realmente puede ejecutar procesamiento de codificación de vídeo. Cuando los datos incluyen audio (es decir, voz), un codificador de audio incluido en el dispositivo de captura 3102 realmente puede realizar procesamiento de codificación de audio. Para algunos escenarios prácticos, el dispositivo de captura 3102 distribuye los datos de audio y vídeo codificados multiplexándolos juntos. Para otros escenarios prácticos, por ejemplo en el sistema de conferencias de vídeo, los datos de audio codificados y los datos de vídeo codificados no se multiplexan. El dispositivo de captura 3102 distribuye los datos de audio codificados y los datos de vídeo codificados al dispositivo terminal 3106 separadamente.

En el sistema de suministro de contenido 3100, el dispositivo terminal 310 recibe y reproduce los datos codificados. El dispositivo terminal 3106 podría ser un dispositivo con capacidad de recepción y recuperación de datos, tal como teléfono inteligente o Pad 3108, ordenador o portátil 3110, grabadora de vídeo de red (NVR)/ grabadora de vídeo digital (DVR) 3112, TV 3114, descodificador (STB) 3116, sistema de conferencia de vídeo 3118, sistema de vigilancia de vídeo 3120, asistente digital personal (PDA) 3122, dispositivo montado en vehículo 3124, o una combinación de cualquiera de los anteriores, o similar con la capacidad para decodificar los datos codificados mencionado anteriormente. Por ejemplo, el dispositivo terminal 3106 puede incluir el dispositivo de destino 14 como se ha descrito antes. Cuando los datos codificados incluyen vídeo, el decodificador de vídeo 30 incluido en el dispositivo terminal es priorizado para ejecutar decodificación de vídeo. Cuando los datos codificados incluyen audio, un decodificador de audio incluido en el dispositivo terminal es priorizado para realizar procesamiento de decodificación de audio.

Para un dispositivo terminal con su pantalla, por ejemplo, teléfono inteligente o Pad 3108, ordenador o portátil 3110, la grabadora de vídeo de red (NVR)/ grabadora de vídeo digital (DVR) 3112, TV 3114, asistente digital personal (PDA) 3122, o dispositivo montado en vehículo 3124, el dispositivo terminal puede alimentar los datos decodificados a su pantalla. Para un dispositivo terminal que no se equipa con pantalla, por ejemplo, STB 3116, sistema de conferencia de vídeo 3118, o sistema de vigilancia de vídeo 3120, una pantalla externa 3126 es contactada ahí para recibir y mostrar los datos decodificados.

Cuando cada dispositivo en este sistema realiza codificación o decodificación, se puede utilizar el dispositivo de codificación de imagen o el dispositivo de decodificación de imagen, tal como se muestra en las realizaciones antes mencionadas.

55

La FIG. 17 es un diagrama que muestra una estructura de un ejemplo del dispositivo terminal 3106. Después que el dispositivo terminal 3106 recibe el flujo desde el dispositivo de captura 3102, la unidad de procedimiento de protocolo 3202 analiza el protocolo de transmisión del flujo. El protocolo incluye pero no se limita a Protocolo de Transmisión por Secuencias en Tiempo Real (RTSP), Protocolo de Transferencia de Hiper Texto (HTTP), protocolo de transmisión por secuencias HTTP en Vivo (HLS), MPEG-DASH, protocolo de Transporte en Tiempo Real (RTP), Protocolo de Mensajería en Tiempo Real (RTMP), o cualquier tipo de combinación de los mismos, o similar.

Después que la unidad de procedimiento de protocolo 3202 procesa el flujo, se genera el archivo de flujo. El archivo se emite a una unidad de desmultiplexión 3204. La unidad de desmultiplexión 3204 puede separar los datos multiplexados en los datos de audio codificados y los datos de vídeo codificados. Tal como se ha descrito

65



antes, para algunos escenarios prácticos, por ejemplo en el sistema de conferencia de vídeo, los datos de audio codificados y los datos de vídeo codificados no se multiplexan. En esta situación, los datos codificados son transmitidos al decodificador de vídeo 3206 y al decodificador de audio 3208 sin hacerlo a través de la unidad de desmultiplexión 3204.

5

A través del procesamiento de desmultiplexión, se genera el flujo elemental de vídeo (ES), la ES de audio, y opcionalmente el subtítulo. El decodificador de vídeo 3206, que incluye el decodificador de vídeo 30 como se ha explicado en las realizaciones antes mencionadas, decodifica la ES de vídeo a través del método de decodificación como se muestra en las realizaciones antes mencionadas para generar el fotograma de vídeo, y alimenta estos datos a la unidad de sincronización 3212. El decodificador de audio 3208 decodifica el audio ES para generar un fotograma de audio y alimenta estos datos a la unidad sincrónica 3212. Alternativamente, el fotograma de vídeo puede almacenarse en un búfer (no mostrado en la FIG. Y) antes de enviarlo a la unidad sincrónica 3212. De manera similar, el fotograma de audio puede almacenarse en un búfer (que no se muestra en la FIG. Y) antes de alimentarse a la unidad de sincronización 3212.

10

15

La unidad de sincronización 3212 sincroniza el fotograma de vídeo y el fotograma de audio, y suministra el vídeo/audio a una pantalla de vídeo/audio 3214. Por ejemplo, la unidad de sincronización 3212 sincroniza la presentación de la información de vídeo y audio. La información se puede codificar en la sintaxis utilizando marcas de tiempo concernientes a la presentación de datos de audio y visuales codificadas y marcas de tiempo concernientes a la entrega del flujo de datos en sí mismo.

20

Si el subtítulo se incluye en el flujo, el decodificador de subtítulo 3210 decodifica el subtítulo, y lo sincroniza con el fotograma de vídeo y el fotograma de audio, y suministra el vídeo/audio/subtítulo a una pantalla de vídeo/audio/subtítulo 3216.

25

La presente invención no se limita al sistema antes mencionado, y cualquiera del dispositivo de codificación de imágenes o el dispositivo de decodificación de imágenes en las realizaciones antes mencionadas se puede incorporar en otro sistema, por ejemplo, un sistema de automóvil.

30

Los operadores matemáticos utilizados en esta solicitud son similares a aquellos utilizados en el lenguaje de programación C. Sin embargo, los resultados de las operaciones de división de entero y cambio aritmético se definen de manera más precisa, y se definen operaciones adicionales, tal como exponenciación y división de valor real.

35

Para explicaciones de contenido relacionado en esta realización, implementaciones de etapas relacionadas, efectos benéficos y similares, se refieren a las partes correspondientes anteriores, o se pueden realizar modificaciones simples basadas en las partes correspondientes anteriores. En esta memoria no se describen de nuevo detalles.

40

Se debería observar que, en un caso en el que no ocurre un conflicto, algunas características en cualesquiera dos o más de las realizaciones anteriores se pueden combinar para formar una nueva realización. Además, algunas características en cualquiera de las realizaciones anteriores pueden utilizarse de manera independiente como una realización.

45

Lo anterior describe principalmente las soluciones proporcionadas en las realizaciones de esta solicitud desde una perspectiva de los métodos. Para implementar las funciones anteriores, se incluyen estructuras de hardware y/o módulos de software correspondientes para realizar las funciones. Un experto en la técnica debería tener fácilmente conocimiento de que, en combinación con los ejemplos descritos en las realizaciones divulgadas en esta solicitud, unidades y etapas de algoritmo pueden implementarse por hardware o una combinación de hardware y software informático en esta solicitud. Si una función se realiza por hardware o hardware accionado por software informático, esto depende de aplicaciones particulares y restricciones de diseño de las soluciones técnicas. Un experto en la técnica puede utilizar diferentes métodos para implementar las funciones descritas para cada aplicación particular, pero no se debería considerar que la implementación va más allá del alcance de esta solicitud.

50

55

La división de un codificador/un decodificador en módulos funcionales en las realizaciones de esta solicitud puede realizarse con base en los ejemplos del método anterior. Por ejemplo, cada módulo funcional se puede obtener a través de división en correspondencia con cada función, o al menos dos funciones pueden integrarse en un módulo de procesamiento. El módulo integrado puede implementarse en una forma de hardware, o puede implementarse en una forma de un módulo funcional de software. Se debería observar que, en las realizaciones de esta solicitud, la división de módulo es un ejemplo, y es simplemente una división de función lógica. En una implementación real, se puede utilizar otra manera de división.

60

Aunque se han descrito realizaciones de la invención principalmente con base en el codificado de vídeo, se debería observar que realizaciones del sistema de codificado 10, el codificador 20 y el decodificador 30 (y de manera correspondiente el sistema 10) y las otras realizaciones descritas en esta memoria también se pueden

65

configurar para procesamiento o codificado de imágenes estáticas, es decir, el procesamiento o codificado de una imagen individual independiente de cualquier imagen precedente o consecutiva como en codificado de vídeo. En general solamente las unidades de interpredicción 244 (codificador) y 344 (decodificador) pueden no estar disponibles en el caso en que la codificación de procesamiento de imagen esté limitada a una sola imagen

5 17. Todas las otras funcionalidades (también denominadas herramientas o tecnologías) del codificador de vídeo 20 y decodificador de vídeo 30 igualmente se pueden utilizar para procesamiento de imágenes estáticas, por ejemplo cálculo residual 204/304, transformada 206, cuantización 208, cuantización inversa 210/310, transformada (inversa) 212/312, división 262/362, intrapredicción 254/354, y/o filtrado de bucle 220, 320, y codificación entrópica 270 y decodificación entrópica 304.

10 Realizaciones, por ejemplo del codificador 20 y el decodificador 30, y funciones descritas en esta memoria, por ejemplo con referencia al codificador 20 y el decodificador 30, pueden implementarse en hardware, software, firmware, o cualquier combinación de los mismos. Si se implementan en software, las funciones pueden almacenarse en un medio legible por ordenador o transmitirse sobre el medio de comunicación como una o

15 más instrucciones o código y ejecutadas por una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir soportes de almacenamiento legibles por ordenador, que corresponde a un medio tangible tal como soportes de almacenamiento de datos, o medios de comunicación incluyendo cualquier medio que facilite la transferencia de un programa informático de un lugar a otro, por ejemplo, según un protocolo de comunicación. De esta manera, los medios legibles por ordenador generalmente pueden

20 corresponder a (1) soportes de almacenamiento legibles por ordenador tangibles que son no-transitorios o (2) un medio de comunicación tal como una señal u onda portadora. Los soportes de almacenamiento de datos pueden ser cualesquiera medios disponibles a los que se pueda tener acceso a través de uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede

25 incluir un medio legible por ordenador.

A manera de ejemplo, y no limitación, dichos soportes de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético, otros dispositivos de almacenamiento magnético, memoria flash, o cualquier otro medio que pueda

30 utilizarse para almacenar un código de programa deseado en la forma de instrucciones o estructuras de datos y a las que se puede tener acceso a través de un ordenador. También, cualquier conexión se determina de manera apropiada un medio legible por ordenador. Por ejemplo si las instrucciones son transmitidas desde un sitio Web, servidor, u otra fuente remota utilizando un cable coaxial, cable de fibra óptica, par torcido, línea de suscriptor digital (DSL), o tecnologías inalámbricas tal como infrarrojo, radio y microondas, entonces el cable

35 coaxial, cable de fibra óptica, par torcido, DSL, o tecnologías inalámbricas tales como infrarrojo, radio, y microondas se incluyen en la definición de medio. Sin embargo, se debería entender que los soportes de almacenamiento legibles por ordenador y los soportes de almacenamiento de datos no incluyen conexiones, ondas de portadora, señales u otros medios transitorios, sino que más bien se dirigen a soportes de almacenamiento tangibles no transitorios. Disco (disk) y disco (disc), tal como se utilizan en esta memoria,

40 incluyen disco compacto (CD), disco láser, disco óptico, disco versátil digital (DVD), disco (disk) flexible y disco Blu-ray, donde los discos (disks) generalmente reproducen datos de forma magnética, mientras que los discos (discs) reproducen datos de manera óptica con láser. Combinaciones de lo anterior también se deberían incluir dentro del alcance de medios legibles por ordenador.

45 Las instrucciones pueden ser ejecutadas por uno o más procesadores, tal como uno o más procesadores de señal digital (DSP), microprocesadores de propósito general, circuitos integrados de aplicación específica (ASIC), matrices lógicas programables en campo (FPGA), u otros circuitos lógicos discretos o integrados equivalentes. Por consiguiente, el término "procesador" como se utiliza en esta memoria se puede referir a cualquiera de la estructura anterior o cualquier otra estructura adecuada para implementación de las técnicas

50 aquí descritas. Además, en algunos aspectos, la funcionalidad descrita en esta memoria puede proporcionarse dentro de los módulos de hardware y/o software dedicados configurados para codificación y decodificación, o incorporados en un códec combinado. También, las técnicas podrían ser implementadas completamente en uno o más circuitos o elementos lógicos.

55 Las técnicas de esta divulgación pueden implementarse en una gran variedad de dispositivos o aparatos, incluyendo un equipo 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 necesariamente requieren la realización a través de diferentes unidades de hardware. Más bien, como se ha descrito antes,

60 diversas unidades se pueden combinar en una unidad de hardware de códec o pueden ser proporcionadas por un agrupamiento de unidades de hardware interoperativas, incluyendo uno o más procesadores como se ha descrito antes, en conjunto con software y/o firmware conveniente.

REIVINDICACIONES

1. Un método par refinamiento de predicción con flujo óptico, PROF, para un bloque codificado afín en codificación de vídeo o decodificación de vídeo, el método es realizado por un aparato de codificación o un  
5 aparato de decodificación, que comprende:  
  
realizar (1102) un proceso PROF para un subbloque actual del bloque codificado afín para obtener valores de muestra de predicción refinados del subbloque actual del bloque codificado afín, en donde todas de una pluralidad de condiciones de restricción para aplicar PROF no se cumplen para el bloque codificado afín;  
10 en donde realizar (1102) un proceso PROF para un subbloque actual del bloque codificado afín comprende:  
  
realizar un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual; y  
15 obtener un valor de muestra de predicción refinada de la muestra actual en función del valor de predicción delta de la muestra actual y un valor de predicción de muestra de la muestra actual del subbloque actual;  
  
en donde la pluralidad de condiciones de restricción para aplicar PROF comprende:  
20 primera información de indicación indica que PROF está inhabilitado para una imagen que contiene el bloque codificado afín; y  
  
segunda información de indicación indica no división del bloque codificado afín; y  
25 en donde realizar un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual comprende:  
  
obtener (1202) una segunda matriz de predicción, en donde la segunda matriz de predicción se genera en función de información de movimiento del subbloque actual;  
30 generar (1203) una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical con base en la segunda matriz de predicción, en donde la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical tiene el mismo tamaño y el tamaño de la segunda matriz de predicción es mayor o igual al tamaño de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical; y  
35 calcular (1204) un valor de predicción delta de la muestra actual del subbloque actual con base en un valor de gradiente de predicción horizontal de la muestra actual en la matriz de gradiente de predicción horizontal, un valor de gradiente de predicción vertical de la muestra actual en la matriz de gradiente de predicción vertical, y una diferencia entre un vector de movimiento de la muestra actual del subbloque actual y un vector de movimiento de una muestra central del subbloque.  
40  
  
2. El método según la reivindicación 1, en donde antes de la realización de un proceso PROF para un subbloque actual del bloque codificado afín, el método comprende además:  
45 determinar (1101) que la pluralidad de condiciones de restricción para aplicar PROF no se cumplen para el bloque codificado afín.  
  
3. El método según la reivindicación 1 o 2, en donde una de la pluralidad de condiciones de restricción para aplicar PROF es una variable fallbackModeTriggered que se establece a 1.  
50  
  
4. El método según la reivindicación 1, en donde obtener una segunda matriz de predicción (1202) comprende:  
55 generar una primera matriz de predicción con base en información de movimiento del subbloque actual, en donde elementos de la primera matriz de predicción corresponden a valores de muestra de predicción del subbloque actual, y generar la segunda matriz de predicción con base en la primera matriz de predicción; o  
  
generar la segunda matriz de predicción con base en la información de movimiento del subbloque actual.  
60  
  
5. El método según la reivindicación 1 o 4, en donde  
  
un elemento de la segunda matriz de predicción se representa por  $I_1(p, q)$ , en donde un intervalo de valores de p es [-1, sbV], y un intervalo de valores de q es [-1, sbH];  
65 un elemento de la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$  y corresponde a la

muestra (i, j) del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de i es [0, sbW-1], y un intervalo de valores de j es [0, sbH-1]; y

- 5 un elemento de la matriz de gradiente de predicción vertical se representa por  $Y(i, j)$  y corresponde a la muestra (i, j) del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de i es [0, sbW-1], y un intervalo de valores de j es [0, sbH-1]; en donde

sbW representa una anchura del subbloque actual en el bloque codificado afín, sbH representa una altura del subbloque actual en el bloque codificado afín.

- 10 6. El método según cualquiera de las reivindicaciones 1 a 5, antes de la realización de un proceso PROF para un subbloque actual del bloque codificado afín, el método comprende además:

- 15 realizar compensación de movimiento afín basada en subbloque para un subbloque actual del bloque codificado afín para obtener valores de muestra de predicción del subbloque actual.

- 20 7. Un método par refinamiento de predicción con flujo óptico, PROF, para un bloque codificado afín en codificación de vídeo o decodificación de vídeo, el método es realizado por un aparato de codificación o un aparato de decodificación, que comprende:

realizar (1102) un proceso PROF para un subbloque actual del bloque codificado afín para obtener valores de muestra de predicción refinados del subbloque actual del bloque codificado afín, en donde se cumplen todas la pluralidad de condiciones de decisión de flujo óptico para el bloque codificado afín;

- 25 en donde realizar (1102) un proceso PROF para un subbloque actual del bloque codificado afín comprende:

realizar un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual; y

- 30 obtener un valor de muestra de predicción refinada de la muestra actual en función del valor de predicción delta de la muestra actual y un valor de predicción de muestra de la muestra actual del subbloque actual;

en donde la pluralidad de condiciones de decisión de flujo óptico comprende:

- 35 primera información de indicación indica que PROF está habilitada para una imagen que contiene el bloque codificado afín; y

segunda información de indicación indica que se aplica división al bloque codificado afín;

- 40 en donde realizar un procesamiento de flujo óptico para el subbloque actual para obtener un valor de predicción delta de una muestra actual del subbloque actual comprende:

obtener (1202) una segunda matriz de predicción, en donde la segunda matriz de predicción se genera en función de información de movimiento del subbloque actual;

- 45 generar (1203) una matriz de gradiente de predicción horizontal y una matriz de gradiente de predicción vertical con base en la segunda matriz de predicción, en donde la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical tiene el mismo tamaño y el tamaño de la segunda matriz de predicción es mayor o igual al tamaño de la matriz de gradiente de predicción horizontal y la matriz de gradiente de predicción vertical; y

- 50 calcular (1204) un valor de predicción delta de una muestra actual del subbloque actual con base en un valor de gradiente de predicción horizontal de la muestra actual en la matriz de gradiente de predicción horizontal, un valor de gradiente de predicción vertical de la muestra actual en la matriz de gradiente de predicción vertical, y una diferencia entre un vector de movimiento de la muestra actual del subbloque actual y un vector de movimiento de una muestra central del subbloque.

- 60 8. El método según la reivindicación 7, en donde antes de la realización de un proceso PROF para un subbloque actual del bloque codificado afín, el método comprende además:

determinar (1101) que la pluralidad de condiciones de decisión de flujo óptico se cumplen para el bloque codificado afín.

- 65 9. El método según la reivindicación 7 o 8, en donde una de la pluralidad de condiciones de decisión de flujo óptico es una variable fallbackModeTriggered que se establece a 0.

10. El método según la reivindicación 7, en donde obtener una segunda matriz de predicción (1202), comprende:
- 5 generar una primera matriz de predicción con base en información de movimiento del subbloque actual, en donde elementos de la primera matriz de predicción corresponden a valores de muestra de predicción del subbloque actual, y generar la segunda matriz de predicción con base en la primera matriz de predicción; o
- generar la segunda matriz de predicción con base en la información de movimiento del subbloque actual.
- 10 11. El método según la reivindicación 7 o 10, en donde
- un elemento de la segunda matriz de predicción se representa por  $I_1(p, q)$ , en donde un intervalo de valores de  $p$  es  $[-1, sbW]$ , y un intervalo de valores de  $q$  es  $[-1, sbH]$ ;
- 15 un elemento de la matriz de gradiente de predicción horizontal se representa por  $X(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de  $i$  es  $[0, sbW-1]$ , y un intervalo de valores de  $j$  es  $[0, sbH-1]$ ; y
- 20 un elemento de la matriz de gradiente de predicción vertical se representa por  $Y(i, j)$  y corresponde a la muestra  $(i, j)$  del subbloque actual en el bloque codificado afín, en donde un intervalo de valores de  $i$  es  $[0, sbW-1]$ , y un intervalo de valores de  $j$  es  $[0, sbH-1]$ ; en donde
- $sbW$  representa una anchura del subbloque actual en el bloque codificado afín,  $sbH$  representa una altura del subbloque actual en el bloque codificado afín.
- 25 12. El método según cualquiera de las reivindicaciones 7 a 11, antes de la realización de un proceso PROF para un subbloque actual del bloque codificado afín, el método comprende además:
- 30 realizar compensación de movimiento afín basada en subbloque para un subbloque actual del bloque codificado afín para obtener valores de muestra de predicción del subbloque actual.
13. Un codificador (20) caracterizado porque comprende circuitos de procesamiento para llevar a cabo el método según cualquiera de las reivindicaciones 1 a 12.
- 35 14. Un decodificador (30) caracterizado porque comprende circuitos de procesamiento para llevar a cabo el método según cualquiera de las reivindicaciones 1 a 12.
15. Un producto de programa informático caracterizado porque comprende un código de programa para llevar a cabo el método según cualquiera de las reivindicaciones 1 a 12.

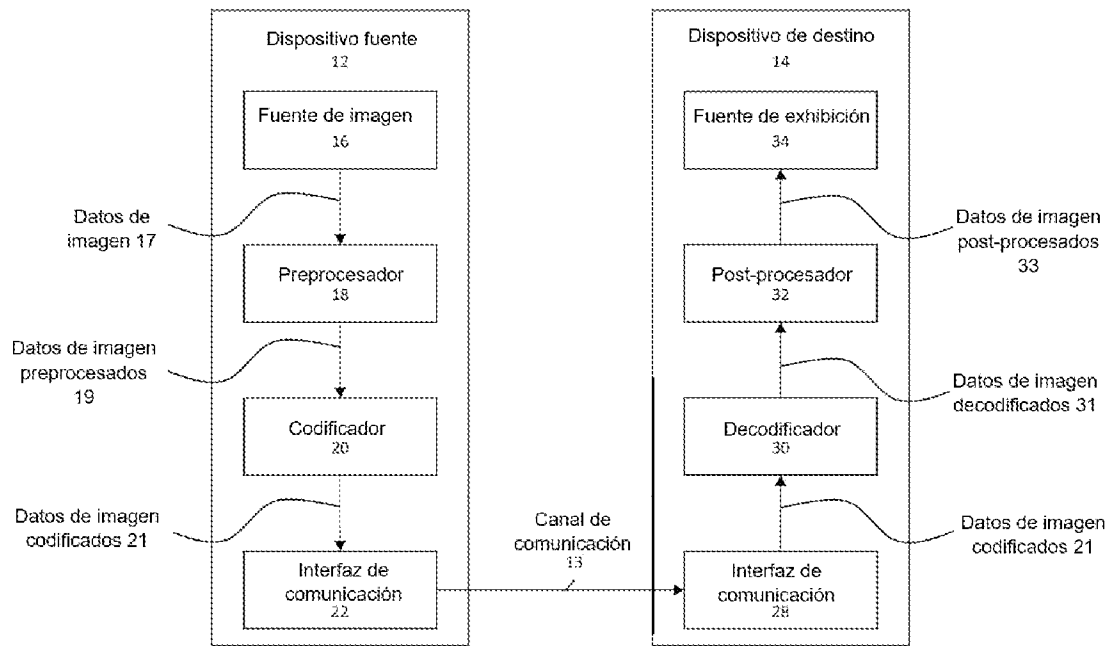


FIGURA 1A

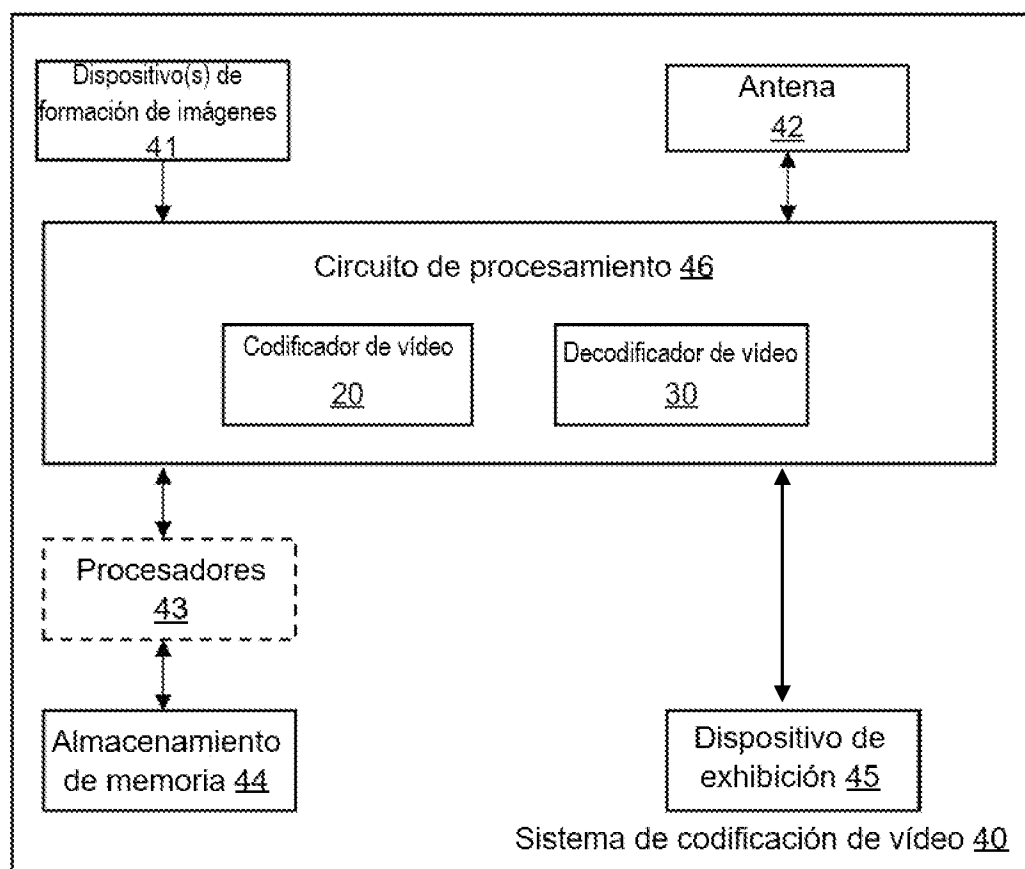


FIGURA 1B

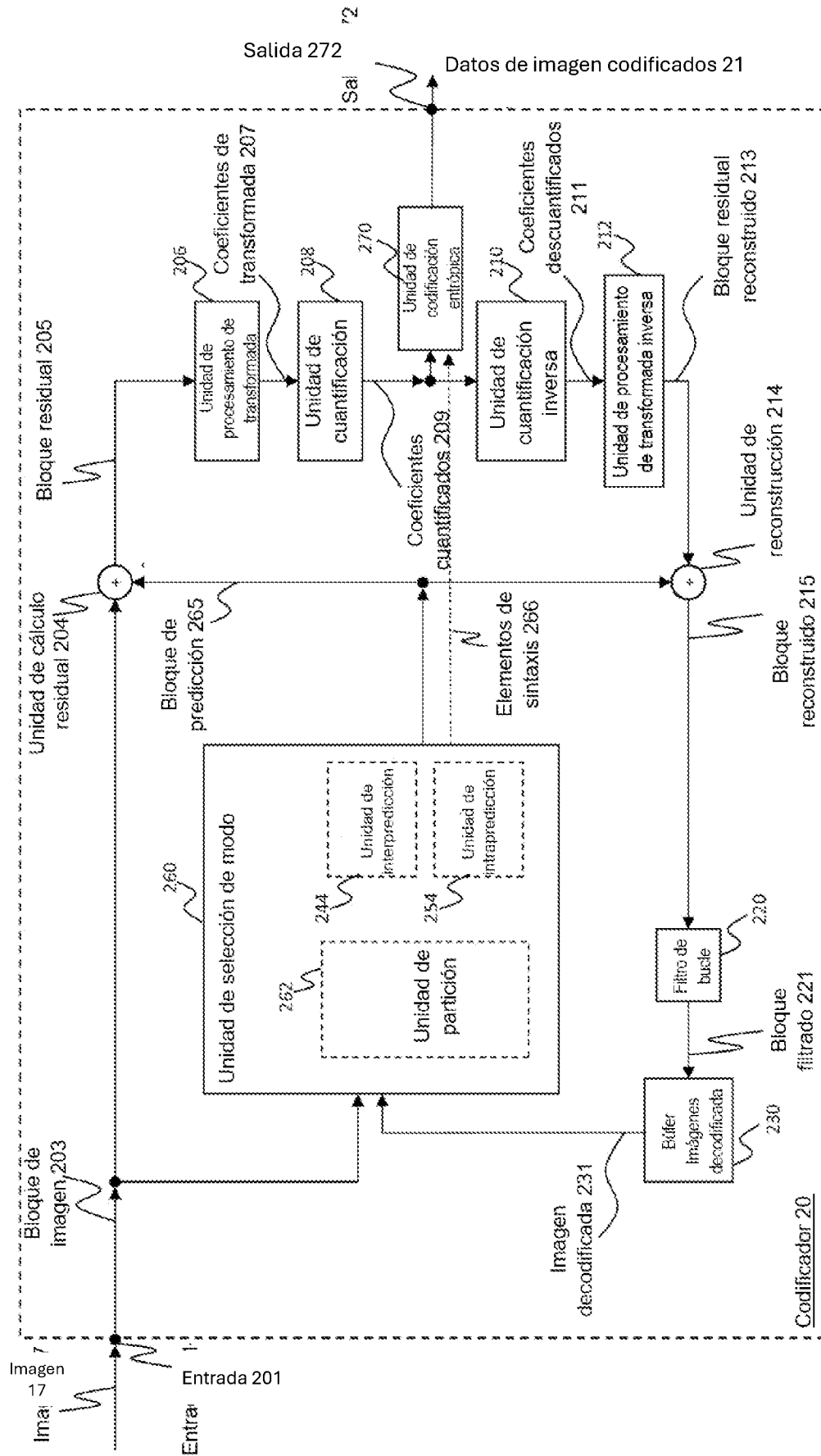


FIGURA 2



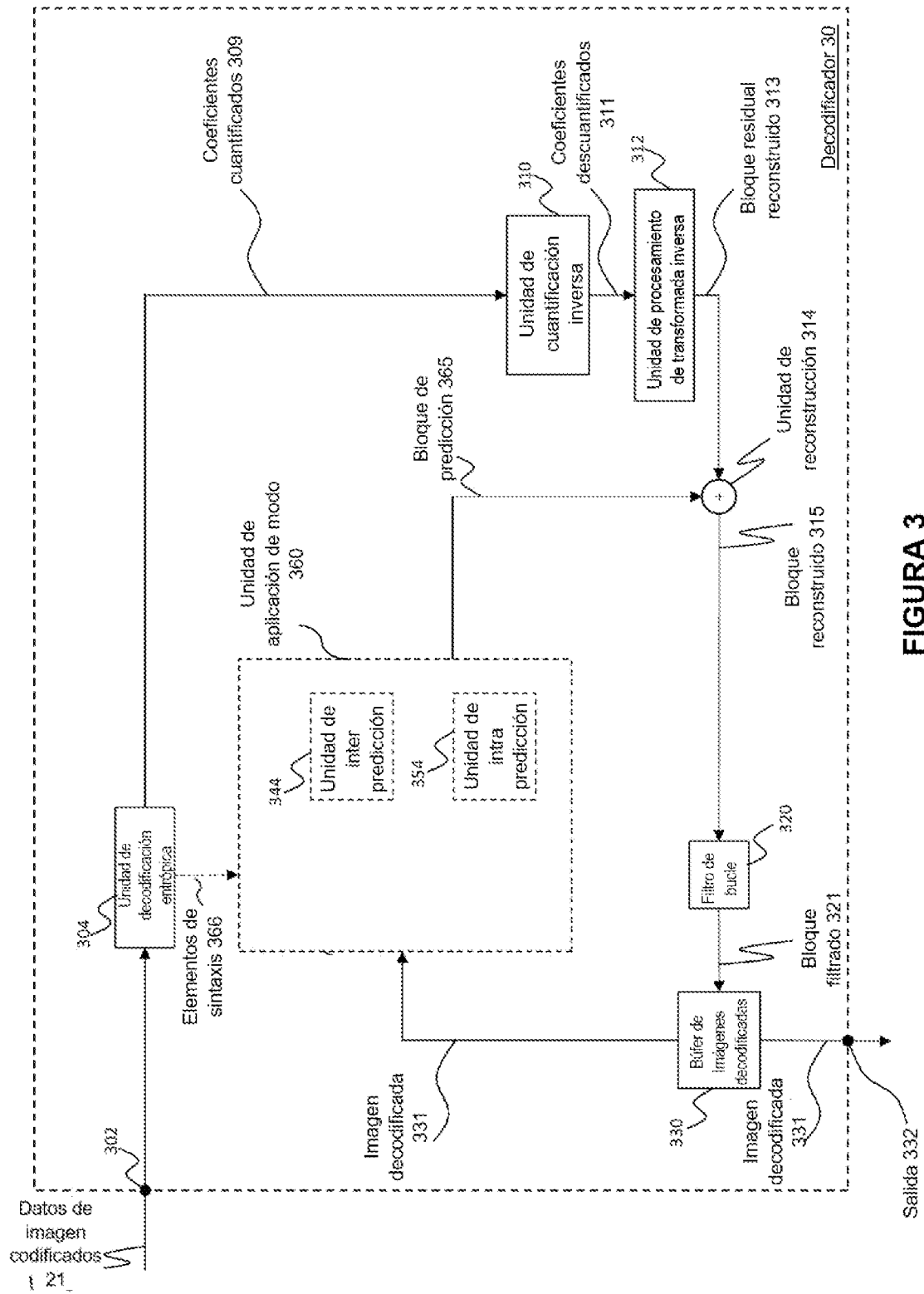


FIGURA 3

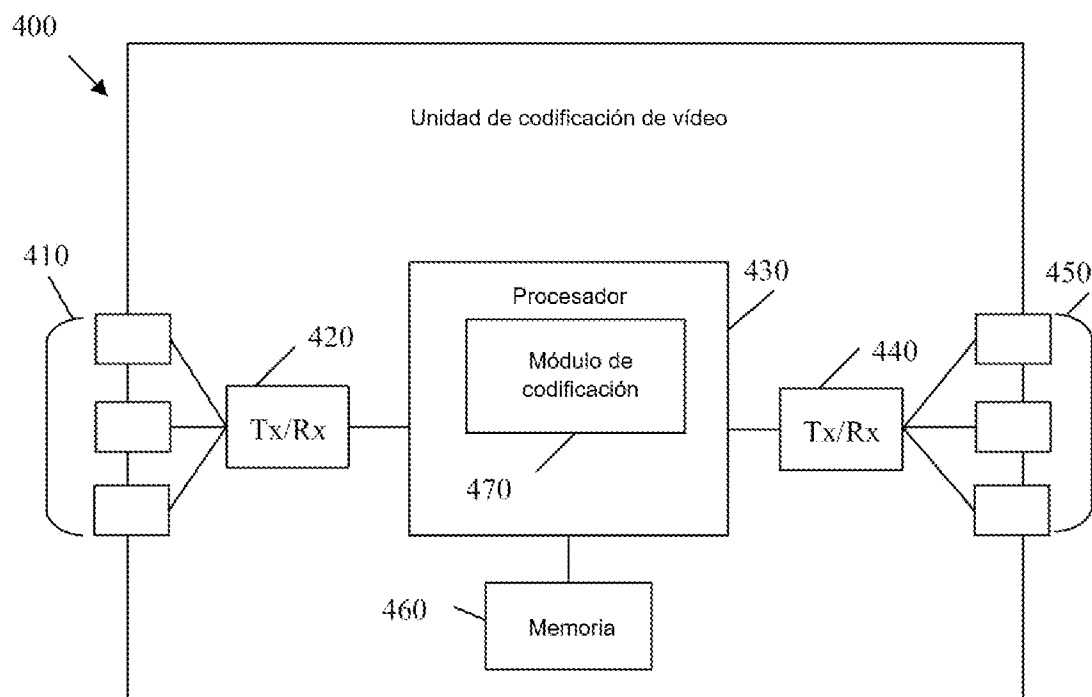
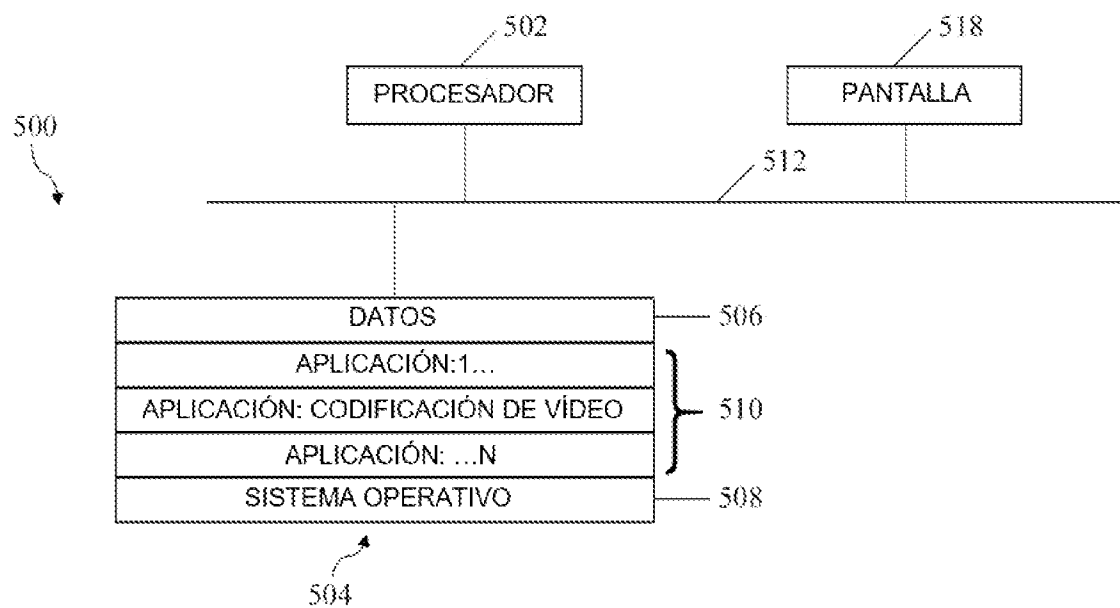
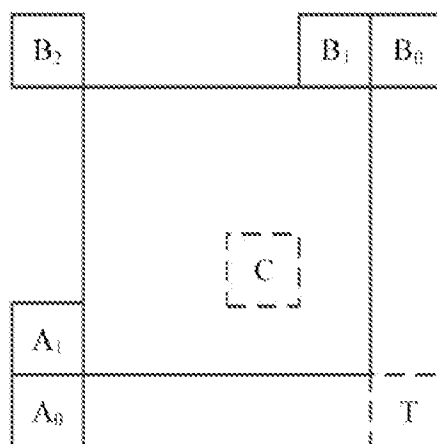


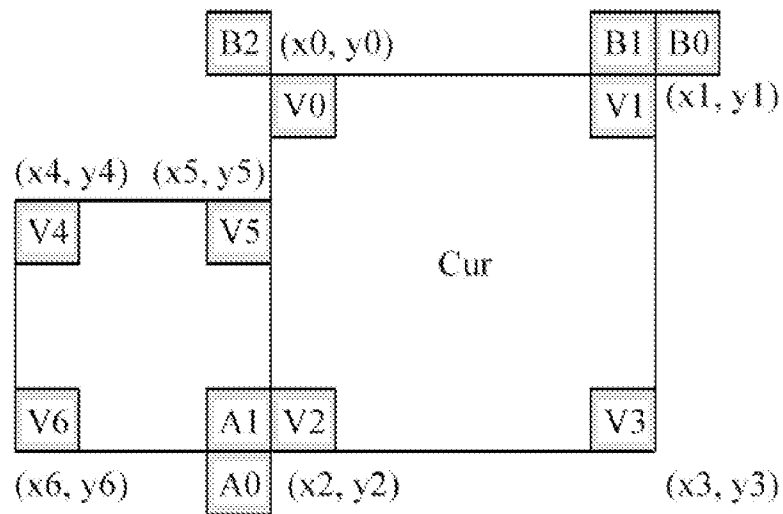
FIGURA 4



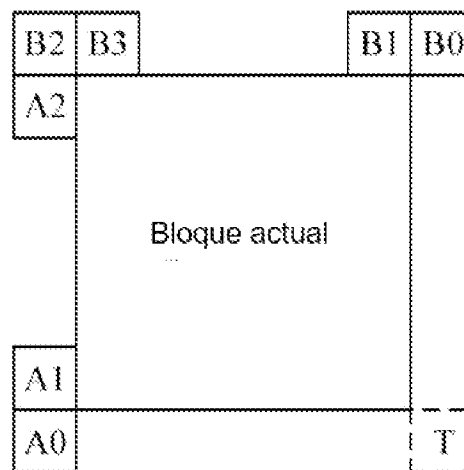
**FIGURA 5**



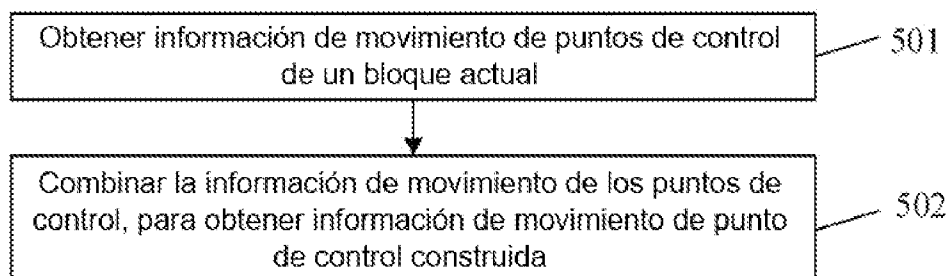
**FIGURA 6**



**FIGURA 7**



**FIGURA 8A**



**FIGURA 8B**

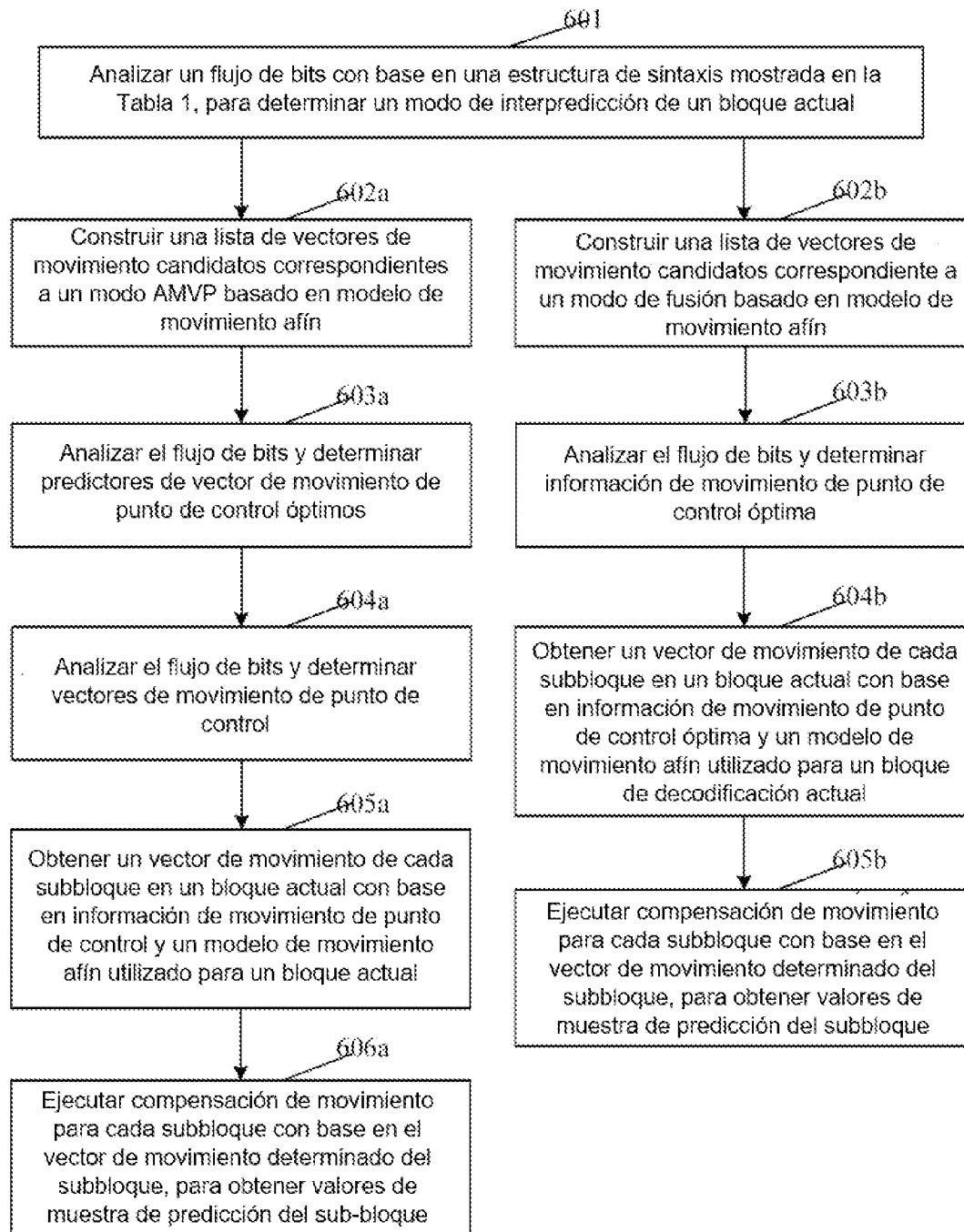
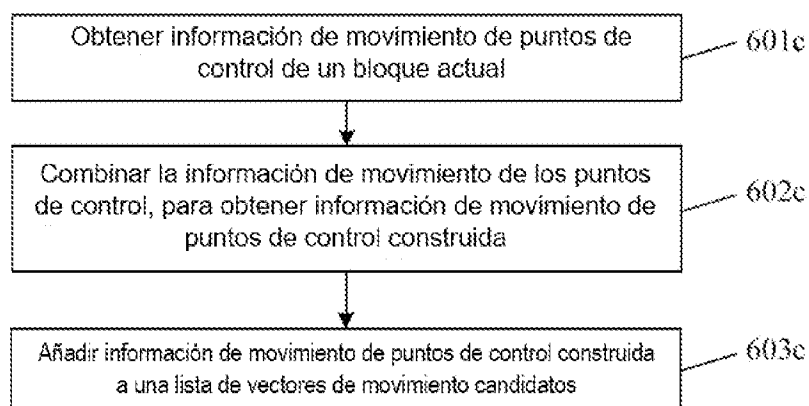
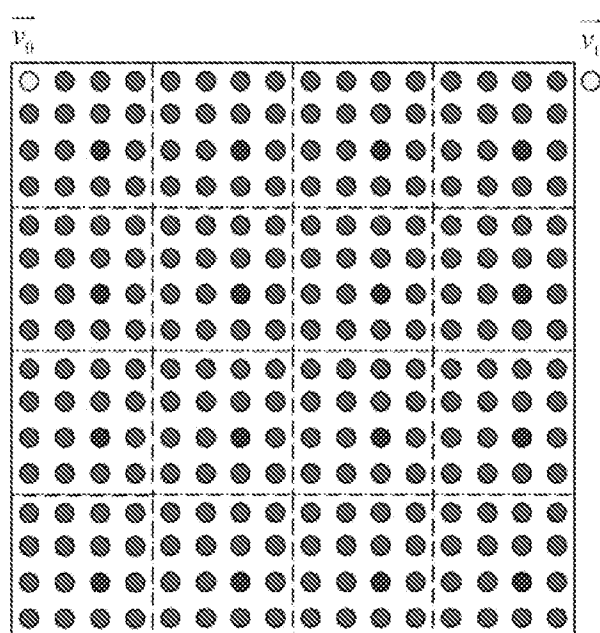


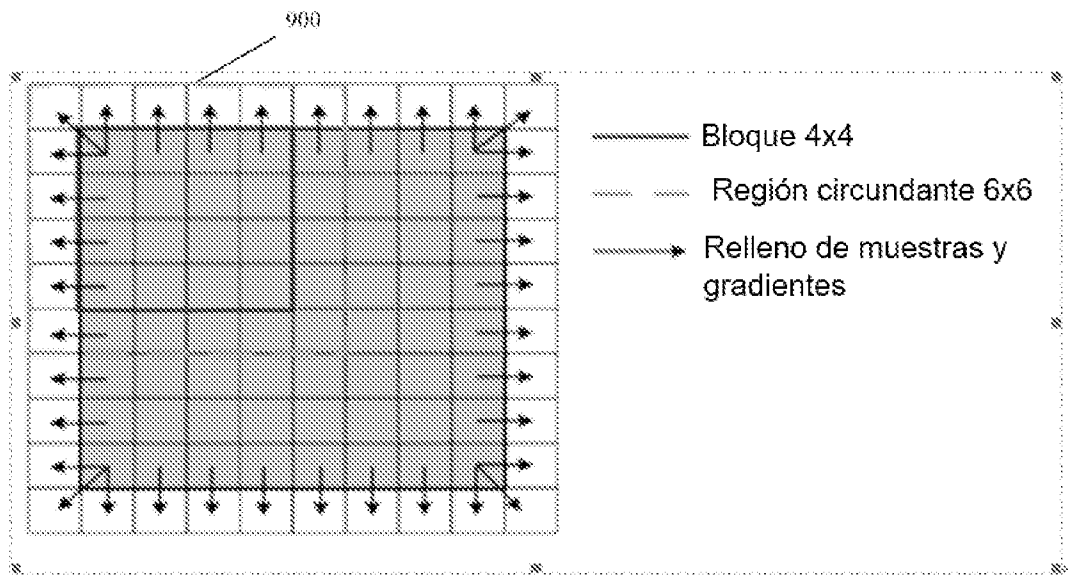
FIGURA 9A



**FIGURA 9B**



**FIGURA 9C**



**FIGURA 9D**

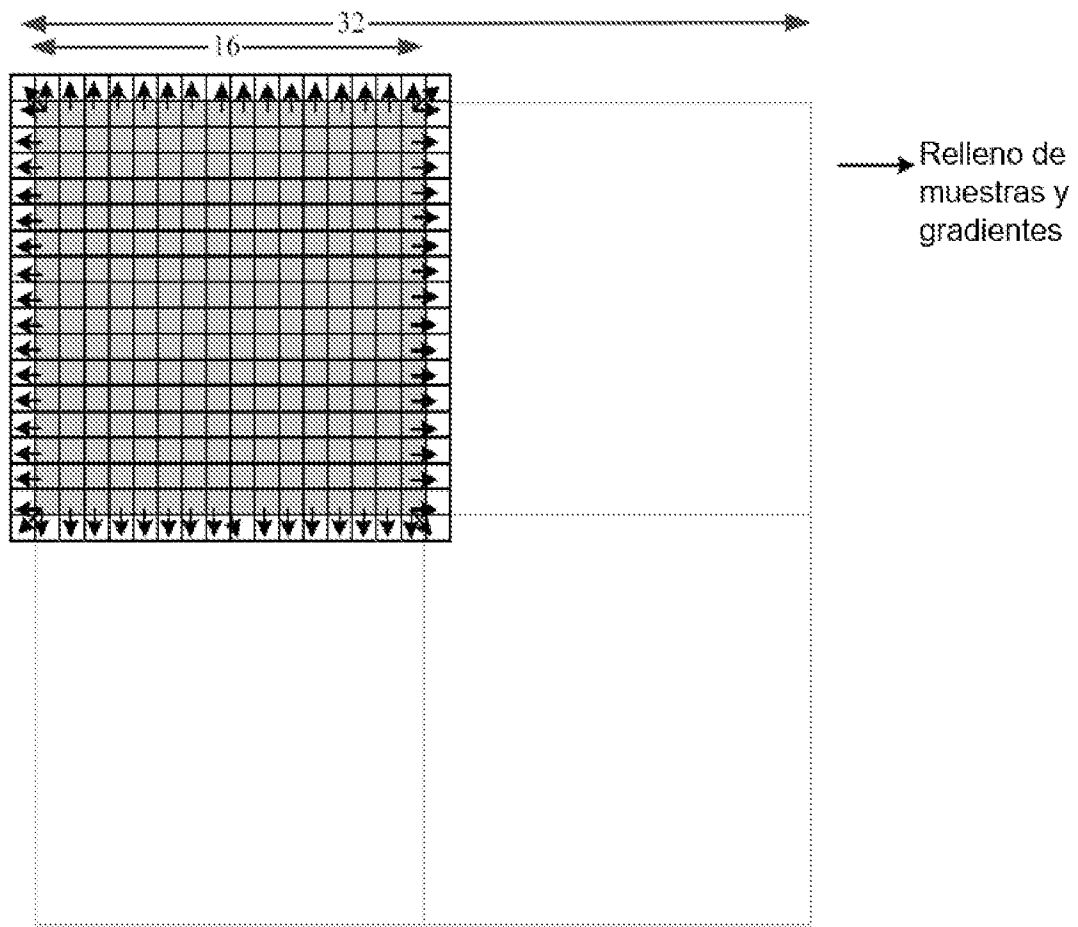


FIGURA 9E

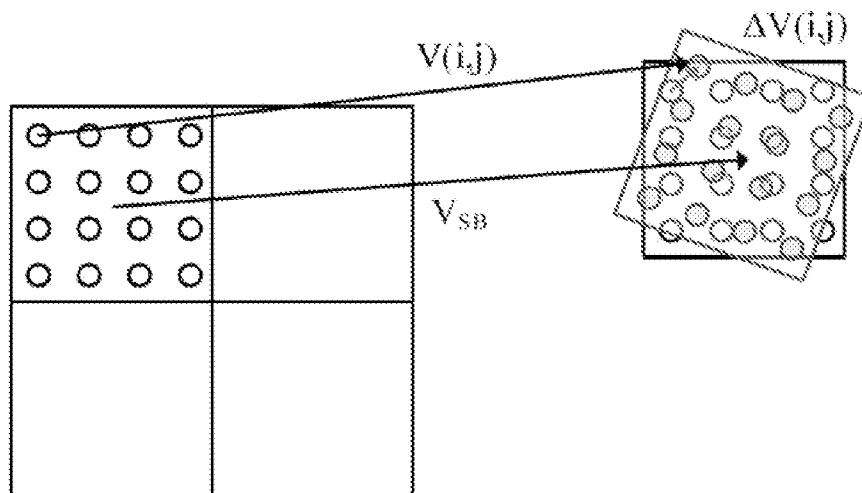
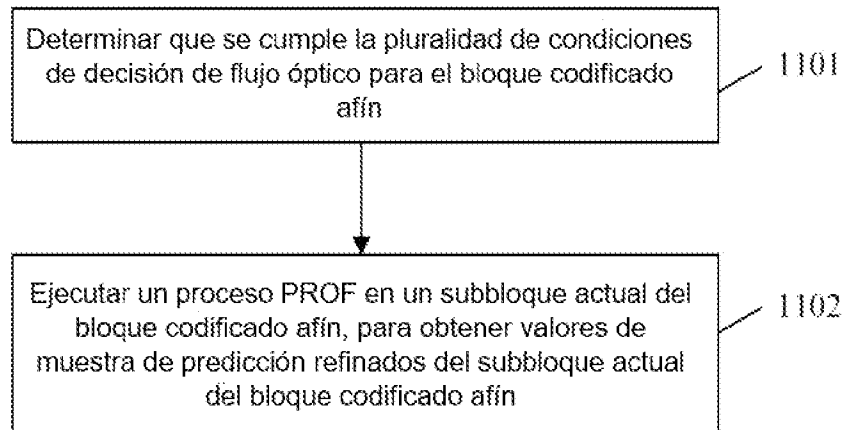
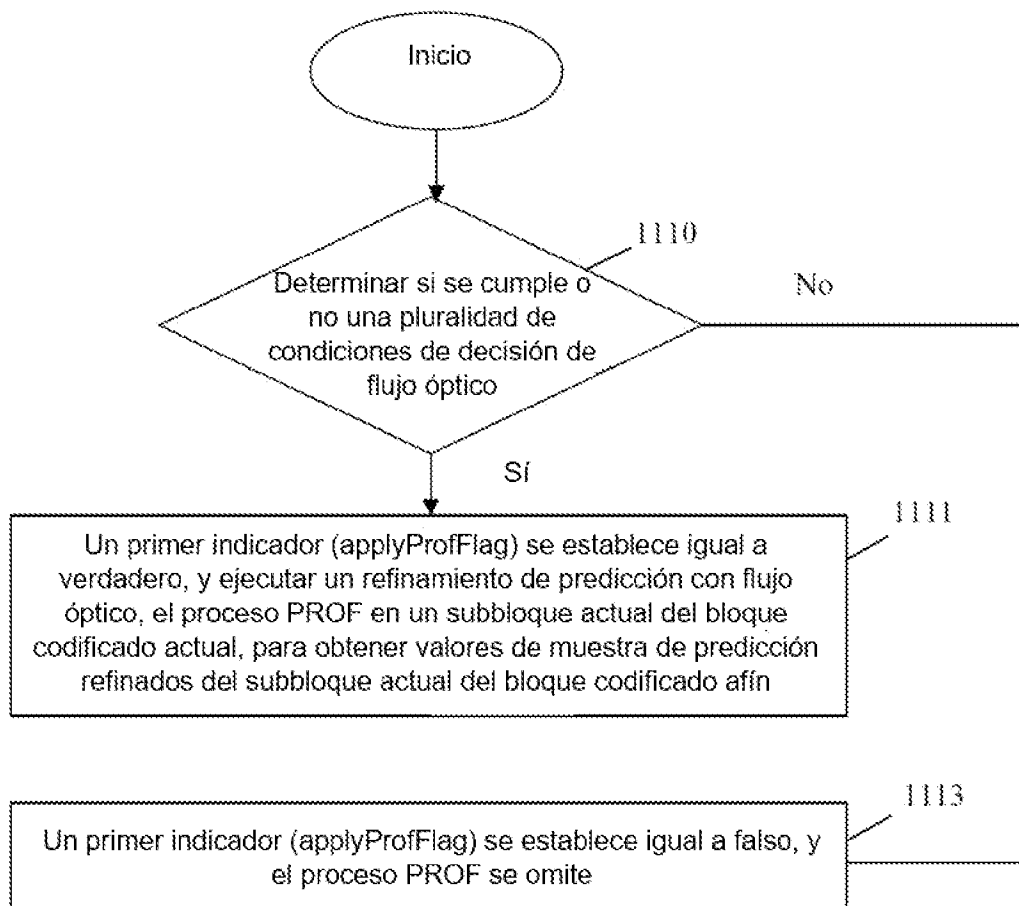


FIGURA 10

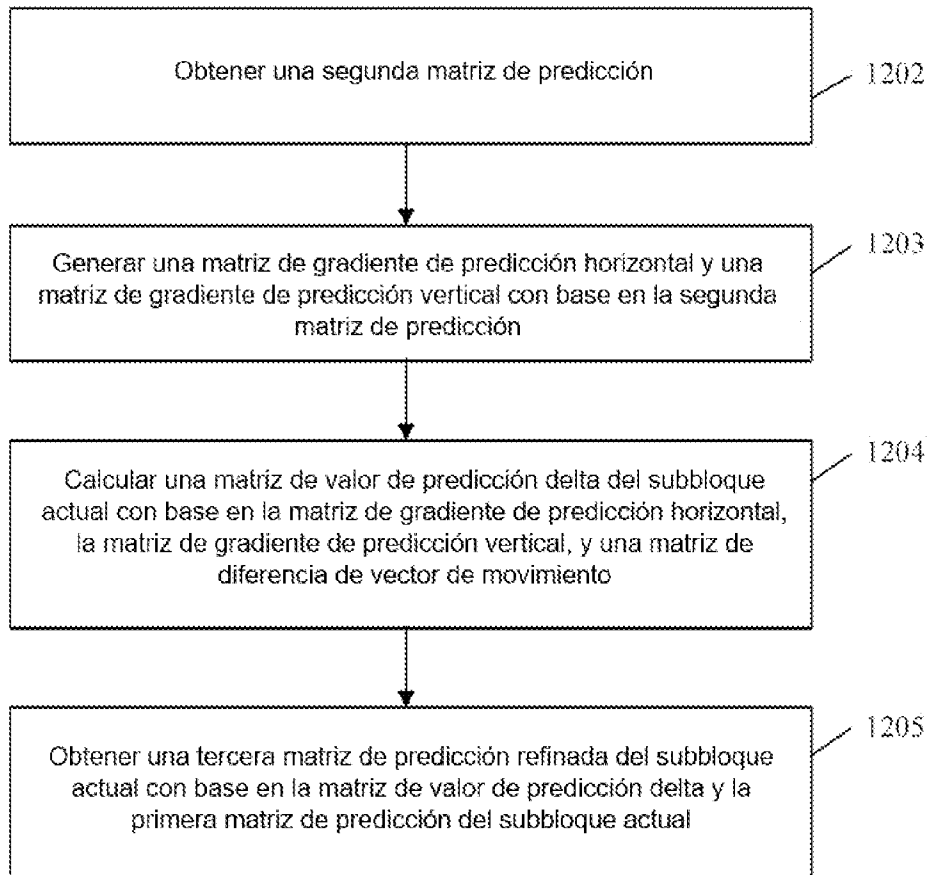




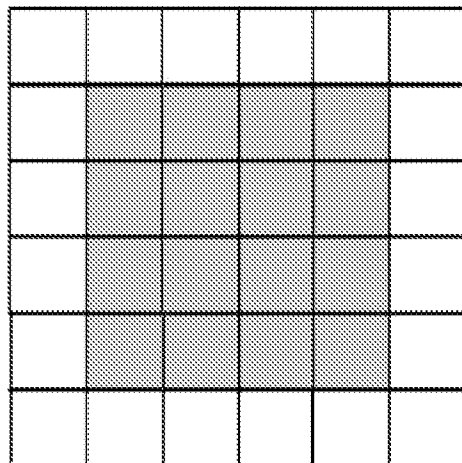
**FIGURA 11A**



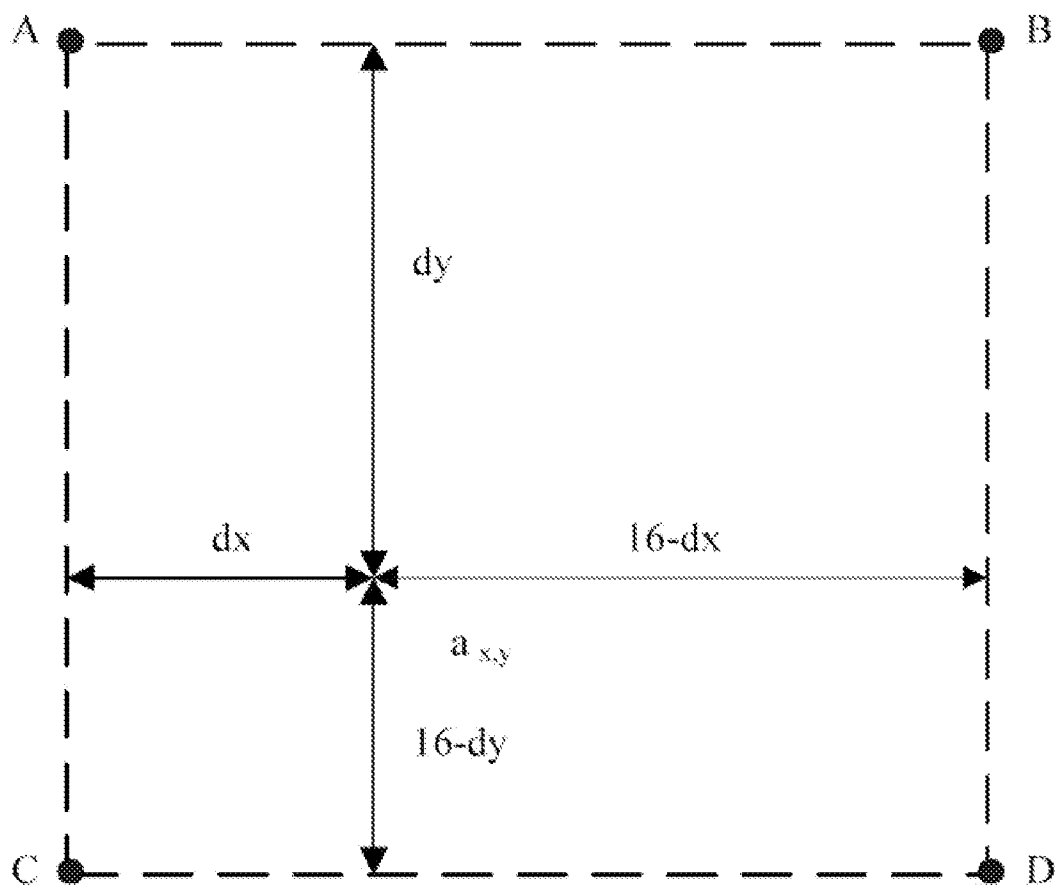
**FIGURA 11B**



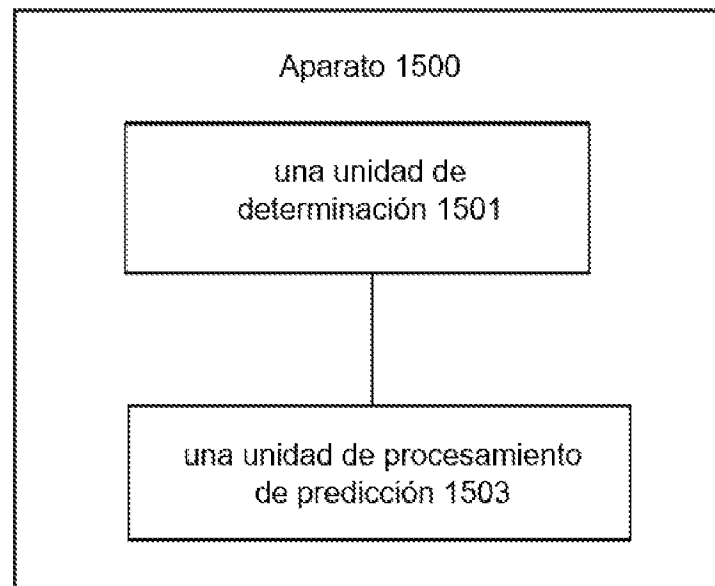
**FIGURA 12**



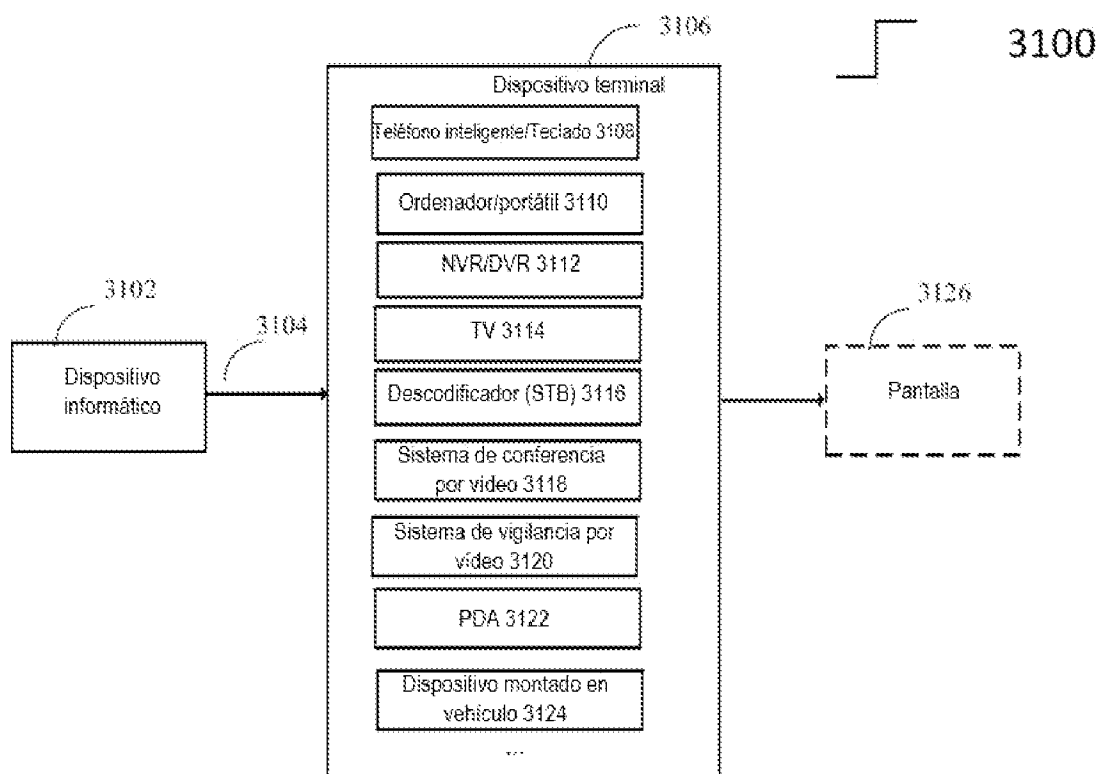
**FIGURA 13**



**FIGURA 14**



**FIGURA 15**



**FIGURA 16**

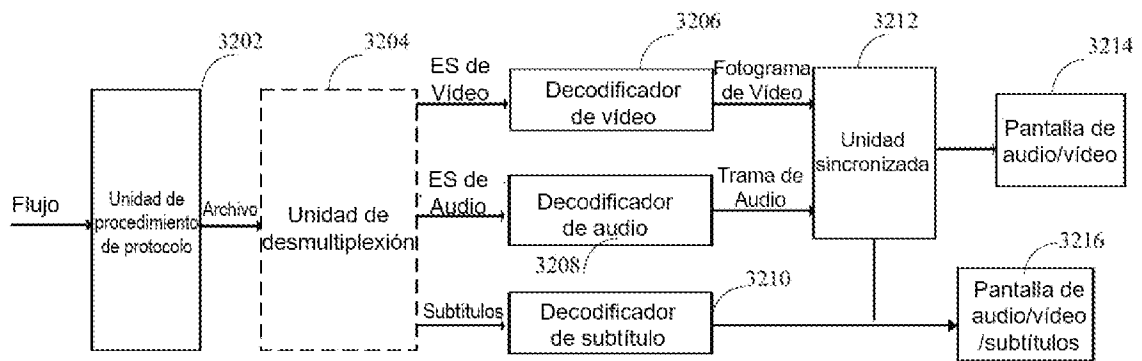


FIGURA 17