

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 803 426**

51 Int. Cl.:

H04N 19/31 (2014.01)

H04N 19/463 (2014.01)

H04N 19/52 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **24.03.2014 E 19164194 (3)**

97 Fecha y número de publicación de la concesión europea: **13.05.2020 EP 3534608**

54 Título: **Procedimiento y aparato para codificar o decodificar una imagen con predicción de información de movimiento de capas inter según un esquema de compresión de información de movimiento**

30 Prioridad:

05.04.2013 GB 201306221

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

26.01.2021

73 Titular/es:

**CANON KABUSHIKI KAISHA (100.0%)
30-2 Shimomaruko 3-chome, Ohta-ku
Tokyo 146-8501, JP**

72 Inventor/es:

**GISQUET, CHRISTOPHE;
ONNO, PATRICE;
LAROCHE, GUILLAUME y
FRANCOIS, EDOUARD**

74 Agente/Representante:

DURAN-CORRETJER, S.L.P

ES 2 803 426 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Procedimiento y aparato para codificar o descodificar una imagen con predicción de información de movimiento de capas inter según un esquema de compresión de información de movimiento

5 La invención se refiere, en general, al sector de la codificación y la descodificación escalables de vídeo, en particular a una codificación y una descodificación escalables de vídeo que extendería el estándar de codificación de vídeo de alta eficiencia (HEVC, High Efficiency Video Coding). Más particularmente, la invención se refiere a un procedimiento, un dispositivo y un programa informático para predicción de vectores de movimiento en un codificador y un descodificador de vídeo escalables.

10 La codificación de vídeo es una manera de transformar una serie de imágenes de vídeo en un flujo de bits digitalizado compacto, de manera que las imágenes de vídeo pueden ser transmitidas o almacenadas. Un dispositivo de codificación se utiliza para codificar las imágenes de vídeo, estando disponible un dispositivo de descodificación asociado para reconstruir el flujo de bits para presentación y visualización. Un objetivo general es formar el flujo de bits para que sea de menor tamaño que la información de vídeo original. Esto reduce ventajosamente la capacidad requerida de una red de transferencia, o de un dispositivo de almacenamiento, para transmitir o almacenar el código del flujo de bits.

15 Se ha adoptado un enfoque común estandarizado para el formato y el procedimiento del proceso de codificación, especialmente con respecto a la parte de descodificación. La gran mayoría de los estándares de vídeo del pasado dividen las imágenes de vídeo en secciones más pequeñas (denominadas macrobloques o bloques). En el nuevo estándar de compresión de vídeo denominado codificación de vídeo de alta eficiencia (HEVC) que se está ultimando, los macrobloques son sustituidos por las así denominadas unidades de codificación mayores (LCU, Largest Coding Units) (denominadas asimismo bloques de árbol de codificación (CTB, Coding Tree Blocks)) y son divididos y ajustados en bloques ahora denominados unidades de codificación (CU, Coding Units), de acuerdo con las características del segmento original de imagen considerado. Esto permite una codificación más detallada de áreas de la imagen de vídeo que contienen relativamente más información, y menos esfuerzo de codificación para las áreas con menos características. Cabe señalar que un área de imagen se conoce asimismo mediante las siguientes expresiones en la bibliografía de compresión de vídeo: bloque de píxeles, bloque, unidad de codificación (CU) y macrobloque.

20 Actualmente se está definiendo una extensión escalable de HEVC. En esta extensión se considera que las imágenes consisten en una serie de capas jerárquicas. Las capas jerárquicas incluyen una capa de base, equivalente a una colección de versiones de baja calidad de imágenes (o cuadros) de la secuencia de vídeo original, y una o varias capas de mejora (conocidas asimismo como capas de refinamiento).

25 Las imágenes de vídeo se procesaban originalmente codificando individualmente cada macrobloque, de manera semejante a la codificación digital de fotografías o imágenes estáticas. Los modelos de codificación posteriores permiten la predicción de las características en un cuadro, ya sea mediante macrobloques contiguos (predicción espacial o INTRA), o mediante asociación con un macrobloque similar en un cuadro contiguo (predicción temporal). Esto permite la utilización de información codificada ya disponible, reduciendo de ese modo la magnitud de la tasa de bits de codificación requerida global. Las diferencias entre el bloque original a codificar y el bloque utilizado para predicción se capturan en un conjunto de residuos de valores. A continuación, el bloque original es codificado en la forma de un identificador del bloque utilizado para predicción y un residuo. Son posibles muchos tipos diferentes de predicciones. La codificación efectiva elige el mejor modo de predicción que proporciona la mejor calidad para el bloque a codificar tras la descodificación, teniendo en cuenta al mismo tiempo el tamaño del flujo de bits inducido por cada modo de predicción para representar dicho bloque en el flujo de bits. El objetivo principal es un compromiso entre la calidad de la imagen descodificada y la reducción en la tasa de bits requerida, conocido asimismo como compromiso tasa/distorsión.

30 En el caso de predicción temporal, son posibles varios tipos de predicción y estos se pueden reunir en dos tipos principales: predicción mono-direccional y predicción bidireccional. En el caso de predicción mono-direccional, el bloque a predecir está asociado con un predictor. La localización del predictor se codifica como información de movimiento. Esta información de movimiento está constituida por un índice del cuadro de referencia que contiene el predictor, denominado *ref_idx* en el estándar, y un vector definido por un desplazamiento vertical y horizontal, que proporciona la localización del bloque predictor en el cuadro contemplado. En caso de predicción bidireccional, el bloque a codificar está asociado con dos predictores tomados en dos cuadros de referencia diferentes. Como consecuencia, la información de movimiento comprende dos índices de cuadros de referencia y dos vectores.

35 La propia información de movimiento se puede codificar de manera predictiva. La información de movimiento obtenida para una unidad de codificación contigua en el mismo cuadro se puede utilizar como predictores espaciales de información de movimiento. La información de movimiento obtenida a partir de una unidad de codificación situada en la misma localización en otros cuadros se puede utilizar como predictor temporal de información de movimiento. La información de movimiento que se tiene que codificar para el bloque a codificar es codificada a continuación con un índice en el predictor de información de movimiento utilizado y con información de residuo que representa la

diferencia entre el predictor de información de movimiento elegido y la información de movimiento a codificar.

La predicción de información de movimiento a partir de información de movimiento correspondiente a imágenes anteriores requiere que el codificador y el decodificador almacenen el campo de movimiento de las imágenes codificadas anteriormente. Este campo de movimiento puede representar una cantidad enorme de datos a almacenar, especialmente para vídeos que presentan una gran resolución, tales como vídeos 4k2k o 8k4k. Para limitar los requisitos de almacenamiento de los códecs HEVC, el estándar HEVC ha adoptado una estrategia que consiste en utilizar campos de movimiento comprimidos para la predicción de información de movimiento, en lugar del campo de movimiento completo.

El almacenamiento de un campo de movimiento es necesario solamente cuando se utiliza la información de movimiento de imágenes anteriores. En HEVC se puede desactivar la utilización de los predictores de información de movimiento temporales. En ese caso, se puede obtener una reducción adicional de los requisitos de almacenamiento de un códec HEVC impidiendo cualquier almacenamiento de información de movimiento.

Una de las ideas principales en un códec escalable es reutilizar información de una capa de referencia (RL, reference layer) codificada utilizando un determinado códec (por ejemplo, HEVC) para codificar información de la capa de mejora.

LASSERRE S et al. "Description of the scalable video coding technology proposal by Canon Research Centre France", Encuentro 11 JCT-VC; Encuentro 102 MPEG; Shangai; JCTVC-K0041, describe herramientas de codificación añadidas a herramientas de codificación HEVC para escalabilidad: una tecnología de codificación intra de inter-capas para tramas intra y herramientas de predicción inter-capas para tramas inter. PARK S et al. "Modifications of temporal mv compression and temporal mv predictor", Encuentro 96 MPEG; Ginebra; m19567, describe modificaciones del esquema de compresión de memoria de vectores de movimiento temporales y cambio de posición de predictor de vector de movimiento temporal.

Sería deseable utilizar la información de movimiento determinada en la capa de referencia para la codificación predictiva de información de movimiento en la capa de mejora. En particular, si la utilización de los predictores de información de movimiento temporales se desactiva en la capa de referencia, puede ocurrir que no haya disponible ninguna información de movimiento a utilizar para la predicción de información de movimiento en la capa de mejora.

La presente invención ha sido concebida para abordar una o varias de las cuestiones anteriores. Se refiere al proceso para determinar un predictor de información de movimiento en la capa de mejora de un esquema de codificación escalable, que se conoce asimismo como proceso de obtención del movimiento. Comprende una corrección de la posición en la capa de referencia utilizada para tomar la información de movimiento más relevante disponible debido al esquema de compresión.

De acuerdo con un aspecto de la invención, se da a conocer un procedimiento de codificación de una imagen acorde con la reivindicación 1 o la reivindicación 3 cuando se lee como anexo a la reivindicación 1.

De acuerdo con otro aspecto de la invención, se da a conocer un procedimiento de decodificación de una imagen acorde con la reivindicación 2 o la 3 cuando se lee como anexo a la reivindicación 2.

De acuerdo con otro aspecto de la invención, se da a conocer un dispositivo para codificar una imagen acorde con la reivindicación 4 o la reivindicación 6 cuando se lee como anexo a la reivindicación 4.

De acuerdo con otro aspecto de la invención, se da a conocer un dispositivo para decodificar una imagen acorde con la reivindicación 5 o la reivindicación 6 cuando se lee como anexo a la reivindicación 5.

De acuerdo con otro aspecto de la invención, se da a conocer un producto de programa informático para un aparato programable, comprendiendo el producto de programa informático una secuencia de instrucciones para implementar un procedimiento acorde con la invención, cuando son cargadas y ejecutadas por el aparato programable.

De acuerdo con otro aspecto de la invención, se da a conocer un medio de almacenamiento legible por ordenador que almacena instrucciones de un programa informático para implementar un procedimiento según la invención.

Por lo menos parte de los procedimientos acordes con la invención se puede implementar por ordenador. Por consiguiente, la presente invención puede adoptar la forma de una realización íntegramente de hardware, una realización íntegramente de software (incluyendo software inalterable, software residente, microcódigo, etc.) o una realización que combina aspectos de software y de hardware, pudiéndose hacer referencia en general a todas ellas como un "circuito", "módulo" o "sistema". Además, la presente invención puede adoptar la forma de un producto de programa informático incorporado en cualquier medio tangible de expresión que tenga código de programa utilizable por ordenador incorporado en el medio.

Dado que la presente invención se puede implementar en software, la presente invención se puede realizar como

código legible por ordenador para proporcionar a un aparato programable en cualquier medio de soporte adecuado. Un medio de soporte tangible puede comprender un medio de almacenamiento, tal como un disco flexible, un CD-ROM, una unidad de disco duro, un dispositivo de cinta magnética o un dispositivo de memoria de estado sólido, y similares. Un medio de soporte transitorio puede incluir una señal, tal como una señal eléctrica, una señal electrónica, una señal óptica, una señal acústica, una señal magnética o una señal electromagnética, por ejemplo una señal de microondas o de RF.

A continuación se describirán realizaciones de la invención, tan sólo a modo de ejemplo, y haciendo referencia a los siguientes dibujos, en los cuales:

la **figura 1** muestra un diagrama de bloques de un codificador de vídeo escalable clásico;
 la **figura 2** muestra posiciones espaciales y temporales para predictores de información de movimiento, en una realización de la invención;
 la **figura 3** muestra la predicción de información de movimiento en la capa de mejora utilizando información de movimiento de la capa de referencia, en una realización de la invención;
 la **figura 4** muestra el diagrama de flujo para el proceso de descodificación, en una realización de la invención;
 la **figura 5** muestra la granularidad de la información de movimiento, en una realización de la invención;
 la **figura 6** muestra esquemáticamente los principios del enfoque *TextureRL*, en una realización de la invención;
 la **figura 7** muestra el proceso adaptado de obtención de un predictor de información de movimiento, en el contexto del enfoque *TextureRL* en modos tanto AMVP como Merge en la capa de mejora, en una realización de la invención;
 la **figura 8** muestra el proceso adaptado en el contexto del enfoque de *índice de cuadro de referencia*, en una realización de la invención;
 la **figura 9** es el diagrama de flujo del proceso de obtención de información de movimiento de los modos Merge, en una realización de la invención;
 la **figura 10** presenta un diagrama de bloques de un descodificador escalable, en una realización de la invención;
 la **figura 11** es un diagrama de bloques esquemático de un dispositivo informático para la implementación de una o varias realizaciones de la invención;
 la **figura 12** muestra la obtención del conjunto de predictores de información de movimiento AMVP, en una realización de la invención;
 la **figura 13** muestra detalles de un área de memoria, en una realización de la invención.

La **figura 1** muestra un diagrama de bloques de un codificador de vídeo escalable clásico, que puede comprender una serie de subsecciones o etapas y que es representativo de una extensión escalable de HEVC. Se muestran en este caso dos subsecciones o etapas **A10** y **B10** que producen datos correspondientes a una capa **1.13** de base y datos correspondientes a una capa **1.14** de mejora. Cada una de las partes secundarias **A10** y **B10** sigue los principios de un codificador de vídeo estándar, aplicándose las etapas de transformación, cuantificación y codificación entrópica en dos caminos diferentes, uno correspondiente a cada capa.

La primera etapa **B10** está dirigida a la codificación de la capa de base compatible con H.264/AVC o HEVC, del flujo escalable de salida. La entrada a este codificador no escalable consiste en la secuencia original de imágenes de cuadro obtenidas aplicando un submuestreo **1.17** a imágenes **1.1**. Este codificador lleva a cabo sucesivamente las siguientes etapas para codificar un flujo de bits de vídeo estándar. Una imagen o cuadro a codificar (a comprimir) se divide en bloques de píxel en la etapa **1.2**, denominada unidad de codificación (CU) en el estándar HEVC. Cada primer bloque sufre una operación **1.3** de estimación de movimiento, que comprende una búsqueda, entre las imágenes de referencia almacenadas en un almacenamiento intermedio de memoria dedicada **1.4**, de bloques de referencia que proporcionen una buena predicción del bloque. Esta etapa de estimación de movimiento proporciona uno o varios índices de imagen de referencia que contienen los bloques de referencia encontrados, así como los correspondientes vectores de movimiento. Una etapa **1.5** de compensación de movimiento aplica a continuación los vectores de movimiento estimados a los bloques de referencia encontrados para obtener un bloque de residuo temporal, que es la diferencia entre el bloque de predicción de movimiento compensado y el bloque original a predecir. Además, una etapa **1.6** de predicción intra determina el modo de predicción espacial que proporcionaría el mejor rendimiento para predecir el bloque actual. De nuevo, se calcula un bloque de residuo espacial pero, en este caso, como la diferencia entre el predictor espacial y el bloque original a predecir.

Después, un mecanismo **1.7** de selección del modo de codificación elige el modo de codificación, entre las predicciones espacial y temporal, que proporciona el mejor compromiso de tasa y distorsión, en la codificación del bloque actual. Dependiendo del modo de predicción seleccionado, el bloque de predicción de residuo sufre a continuación una transformada (DCT) y una cuantificación **1.8**. Se realiza una codificación entrópica **1.10** de los coeficientes QTC (y datos de movimiento MD asociados) así cuantificados. Los datos de textura comprimidos **1.13** asociados al bloque actual codificado **1.2** son enviados para entrega.

Para mejorar más la eficiencia de codificación, la información de movimiento asociada a los bloques INTER, que están realizando una etapa de compensación de movimiento, se codifica de manera predictiva utilizando información de movimiento de bloques contiguos. En tal caso, los bloques contiguos comprenden bloques contiguos espacialmente y, opcionalmente, temporalmente. Como consecuencia, si se utilizan los bloques contiguos temporalmente, es necesario almacenar la información de movimiento de imágenes codificadas anteriormente para

5 permitir una predicción. En la versión actual del estándar, esta información es almacenada de forma comprimida por el codificador y el decodificador para limitar la utilización de memoria del proceso de codificación y decodificación. Sin embargo, tal como se ha mencionado anteriormente, cuando no se utiliza el predictor temporal en la predicción de información de movimiento, no es necesario el almacenamiento del campo de movimiento de imágenes anteriores.

10 El bloque actual se reconstruye a continuación mediante cuantificación inversa (denominada asimismo escalamiento) y transformada inversa **1.11**. Esta etapa es seguida, si es necesario, por una suma del residuo transformado inverso y el bloque de predicción del bloque actual, para formar el bloque reconstruido. Los bloques reconstruidos se añaden al almacenamiento intermedio para formar el cuadro reconstruido. A continuación, este cuadro reconstruido es post-filtrado en **1.12**. El cuadro reconstruido después de este post-filtro se almacena en un almacenamiento intermedio **1.4** de memoria denominado DPB (que significa almacenamiento intermedio de imágenes descodificadas (Decoded Picture Buffer)) de manera que está disponible para su utilización como imagen de referencia para predecir cualesquiera imágenes posteriores a codificar.

15 Finalmente, una última etapa de codificación entrópica recibe el modo de codificación y, en caso de un bloque inter, los datos de movimiento, así como los coeficientes DCT cuantificados calculados anteriormente. Este codificador entrópico codifica cada uno de estos elementos de datos en su forma binaria y encapsula los bloques así codificados en un contenedor denominado unidad NAL (que significa capa abstracta de red (Network Abstract Layer)). Una unidad NAL contiene todas las unidades de codificación codificadas de un determinado segmento. Un flujo de bits HEVC codificado consiste en una serie de unidades NAL.

20 A continuación, la segunda etapa **A10** muestra la codificación de una capa de mejora utilizando la capa de base como capa de referencia. En este caso, esta capa de mejora proporciona un refinamiento de la resolución espacial para la capa de base sobremuestreada. Tal como se muestra en la **figura 1**, el esquema de codificación de esta capa de mejora es similar al de la capa de base, excepto en que para cada unidad de codificación de una imagen actual del flujo **1.10** que se está comprimiendo, se consideran modos adicionales en base a la predicción de capas inter. Habitualmente se incluyen las siguientes modificaciones.

25 Un modo adicional denominado IntraBL **1.90** que consiste en predecir un bloque de la capa de mejora utilizando el bloque sobremuestreado situado en la misma localización de la capa de referencia, se añade a la lista de modos considerados para un bloque de la capa de mejora.

30 La predicción de información de movimiento utilizada en los modos INTER se puede modificar para tener en cuenta información de movimiento procedente de una capa de referencia. Además, se puede aplicar codificación entrópica específica de la información de movimiento durante la etapa **1.20** de codificación entrópica.

35 Para estas nuevas herramientas, un módulo inter-capa intermedio **1.18** puede proporcionar información (información de movimiento, muestras), posiblemente sobremuestreada de acuerdo con el cambio de resolución espacial, desde la capa de referencia a los diferentes módulos de la codificación de la capa de mejora, tales como el módulo **1.23** de estimación de movimiento, el modo IntraBL **1.90** o el módulo **1.26** de predicción intra. En particular, en el enfoque de cuadros de referencia, el módulo **1.18** sobremuestreará tanto los datos de muestra como los datos de movimiento del cuadro resultante en DPB **1.4** para adaptarse a las dimensiones de la capa de mejora, e introducirá los datos resultantes (imagen y su movimiento) en el DPB **1.24** lo que, a su vez, afecta a las operaciones de **1.25** y **1.23**.

40 La **figura 10** presenta un diagrama de bloques de un decodificador escalable que aplicaría a un flujo de bits escalable compuesto de dos capas de escalabilidad, por ejemplo, que comprenden una capa de base y una capa de mejora. Este proceso de descodificación es por lo tanto el proceso inverso al proceso de codificación escalable de la **figura 1**. El flujo escalable que se está descodificando **10.10** se compone de una capa de base y una capa de mejora espacial sobre la capa de base, las cuales son desmultiplexadas en la etapa **10.20** en sus respectivas capas.

45 La primera etapa de la **figura 10** trata del proceso **B12** de descodificación de la capa de base. Este proceso de descodificación comienza mediante la descodificación entrópica **10.30** de cada unidad de codificación o bloque de cada imagen codificada en la capa de base. Esta descodificación entrópica **10.30** proporciona el modo de codificación, la información de movimiento (índices de imágenes de referencia y vectores de movimiento de bloques codificados INTER), dirección de predicción para predicción intra y datos de residuos. Estos datos de residuos consisten en coeficientes DCT cuantificados y transformados. A continuación, estos coeficientes DCT cuantificados sufren operaciones **10.31** de cuantificación y transformada inversa. Se pueden añadir datos de compensación **10.32** de movimiento o de predicción Intra **10.33** a estos residuos por medio de la operación **10.34**.

50 A continuación se aplica la etapa **10.35** de filtro anti-efecto bloque. La imagen así reconstruida se almacena a continuación en el almacenamiento intermedio **10.40** de cuadros.

55 A continuación, la información de movimiento descodificada para bloques INTER, y los bloques reconstruidos, se almacenan en un almacenamiento intermedio de cuadros en el primero de los decodificadores escalables de la **figura 10** (**B12**). Dichos cuadros contienen los datos que pueden ser utilizados como datos de referencia para

predecir una capa de escalabilidad superior.

A continuación, la segunda etapa de la **figura 10** lleva a cabo la descodificación de una capa de mejora espacial **A12** sobre la capa de base descodificada por la primera etapa. Esta descodificación de la capa de mejora espacial implica la descodificación entrópica de la segunda capa, que proporciona los modos de codificación, la información de movimiento e información de predicción intra, así como la información de residuos transformados y cuantificados de bloques de la segunda capa.

La etapa siguiente consiste en predicción de bloques en la imagen mejorada. La elección **10.51** entre diferentes tipos de predicción de bloque (INTRA, INTER o, en el caso del enfoque *TextureRL*, inter-capas) depende del modo de predicción obtenido a partir de la etapa **10.50** de descodificación entrópica.

En relación con los bloques INTRA, su tratamiento depende del tipo de unidad de codificación INTRA. En el caso de un bloque INTRA predicho inter-capas (modo de codificación IntraBL) **10.57**, si han sido codificados datos de residuo para el bloque, el resultado de la descodificación entrópica **10.50** se somete a cuantificación inversa y transformada inversa **10.52**, y a continuación se suma mediante la operación **10.53** al bloque situado en la misma localización del bloque actual de la imagen de base, en su versión descodificada, post-filtrada y sobremuestreada (en caso de escalabilidad espacial). En el caso de un bloque INTRA, dicho bloque es totalmente reconstruido, mediante cuantificación inversa, transformada inversa para obtener los datos de residuo en el dominio espacial, y a continuación predicción INTRA **10.54** para obtener el bloque totalmente reconstruido.

En relación con los bloques INTER, su reconstrucción implica su compensación **10.55** de movimiento calculada a partir de la memoria **10.60** de cuadros, la descodificación de los datos de residuo y a continuación la adición de su información de residuo descodificada a su bloque predictor temporal. En este proceso de descodificación de bloques INTER, la información de movimiento asociada al bloque considerado se puede descodificar de manera predictiva, como un refinamiento de la información de movimiento del bloque situado en la misma localización en la imagen de base. Este aspecto se detallará a continuación.

Como en la capa de base, la etapa **10.56** del filtro anti-efecto bloque se puede aplicar a las imágenes entregadas desde **10.53**, y almacenadas en la memoria **10.60** de cuadros antes de ser devueltas por el descodificador como cuadros completamente descodificados **10.70**. Se debe observar que en la realización de la invención, la compensación **10.55** de movimiento utiliza realmente datos del almacenamiento intermedio **10.60** de imágenes de la capa de mejora y del almacenamiento intermedio **10.40** de imágenes de la capa de base. Junto con los datos muestreados, el módulo **10.80** puede estar a cargo de proporcionar dichos datos desde la capa de referencia sobremuestreándolos.

Se pueden contemplar dos enfoques para predicción de imágenes. Más particularmente, la predicción de imágenes comprende la predicción de información de movimiento. En ambos enfoques, se ve afectada la predicción de información de movimiento, pero de maneras diferentes. Los dos enfoques de la predicción de imágenes se describen a continuación. Una característica de estos enfoques es permitir utilizar información de movimiento de la capa de referencia para la predicción de información de movimiento en la capa de mejora. Esta característica se describirá en mayor detalle en relación con la **figura 3**, y afecta a cómo se accede a la memoria **1.4** de cuadros de la **figura 1**, de la capa de referencia.

A continuación se explica en detalle para ambos enfoques el caso de la predicción de información de movimiento, que es el punto clave de esta invención.

El primer enfoque se denomina normalmente *TextureRL*, dado que se permite la utilización del modo IntraBL. Este enfoque utiliza sintaxis de bajo nivel a nivel de bloque para señalar la utilización del modo IntraBL. Este enfoque es denominado en ocasiones, por algunos expertos, "enfoque IntraBL".

El segundo enfoque denominado introducción de cuadro de referencia es para utilizar principalmente cambios de alto nivel. En particular, no se realiza ningún cambio de sintaxis a nivel de bloque. La característica principal del enfoque de *índice de cuadro de referencia* es introducir imágenes (posiblemente escaladas ascendentemente cuando la resolución es diferente) de la capa de referencia, denominadas imágenes ILR (que significa imágenes de referencia de capa inter (Inter Layer Reference)) en el almacenamiento intermedio de imágenes descodificadas de la capa de mejora. Estas imágenes son introducidas a continuación al final de las listas de imagen de referencia específicas (lista L0 y L1) utilizadas como imágenes de referencia en el DPB (Decoded Picture Buffer, almacenamiento intermedio de imágenes descodificadas). La introducción depende del tipo de segmento actual de la capa de mejora. En un segmento P, la imagen ILR se introduce al final de la lista L0. En un segmento B, la imagen ILR se introduce al final de ambas listas L0 y L1. Este enfoque es denominado en ocasiones, por algunos expertos, "ref_idx approach". Con este enfoque, la información de movimiento de un determinado bloque se puede codificar predictivamente utilizando un predictor temporal de información de movimiento de la capa de referencia situado en la misma localización en la capa de referencia.

Cabe señalar que IntraBL en el enfoque *TextureRL* y la utilización del cuadro de referencia introducido en el enfoque

de *índice de cuadro de referencia* son dos modos de utilizar la información de capa de base para predecir la capa de mejora. En la descripción de esta invención y para mayor simplicidad, se considera uno de estos enfoques cada vez, pero no ambos al mismo tiempo.

5 Se describirá a continuación información general en relación con información de movimiento, que es válida para ambos enfoques.

10 Un típico códec de vídeo explota las correlaciones tanto espacial como temporal entre píxeles en los respectivos modos Intra e Inter. La presente memoria se centra en los modos Inter, que explotan la correlación temporal entre píxeles del cuadro actual y de cuadros codificados/descodificados anteriormente.

15 En HEVC (y SHVC por extensión), el modo "Inter" es un modo de predicción que define una dirección de predicción temporal. Se definen ajustes de 0 a 2 de la información de movimiento, dependiendo de esta dirección temporal. Si la dirección de predicción "Inter" es igual a 0, el bloque está codificado con el modo Intra, y no contiene ninguna información de movimiento. Si la dirección de predicción "Inter" es igual a 1, el bloque contiene información de movimiento de una lista de cuadros de referencia denominada L0. Si la dirección de predicción "Inter" es igual a 2, el bloque contiene información de movimiento de otra lista de cuadros de referencia denominada L1. Si la dirección de predicción "Inter" es igual a 3, el bloque contiene información de movimiento de ambas listas L0 y L1.

20 La información de movimiento consiste en la siguiente información, un índice (ref_idx) en una lista de cuadros de referencia y un vector de movimiento que tiene dos componentes: valores de movimiento horizontal y vertical. Estos valores corresponden a un desplazamiento espacial en términos de píxeles entre la posición del bloque actual y el bloque predictor temporal en el cuadro de referencia. Este desplazamiento puede tener precisión sub-píxel (0, 1, 2 o 3 cuartos de píxel).

25 Las direcciones de predicción "Inter" 1 y 2 mencionadas anteriormente corresponden a predicciones unidireccionales y no se pueden utilizar en segmentos I (= segmento codificado Intra) sino en segmentos P (segmento predicho) y B (Bi-predicho). Una imagen de un tipo particular (I, P o B) se compone de por lo menos un segmento del mismo tipo. La dirección 3 de predicción "Inter" se denomina predicción bidireccional, y se puede utilizar solamente en segmentos B. En este caso, se consideran dos predictores de bloque, uno para cada lista L0 y L1. Por consiguiente, se consideran dos índices de cuadro de referencia, así como dos vectores de movimiento. El predictor de bloques Inter para bi-predicción es el promedio, píxel a píxel, de estos dos bloques a los que apuntan estos dos vectores de movimiento. El predictor de bloques corresponde en este caso a la noción de unidad de predicción o bloque de predicción en HEVC o SHVC.

35 Tal como se ha descrito anteriormente, la información de movimiento en HEVC es codificada mediante codificación predictiva, utilizando una serie de predictores de información de movimiento, entre los que está la información de movimiento situada en la misma localización temporalmente. Por lo tanto es necesario, para cada cuadro que se utiliza como cuadro de referencia, almacenar en los lados del codificador y del decodificador su información de movimiento relacionada. Esta información de movimiento se comprime para reducir su tamaño en la memoria dedicada de información de movimiento.

40 HEVC utiliza por lo tanto una granularidad particular para representar el movimiento. Esto se representa en la **figura 5**. Para cada bloque **5.0** de 16x16 píxeles, la granularidad mínima utilizada por HEVC es de 4x4 píxeles, con el resultado de 16 elementos potenciales de información de movimiento, uno para cada bloque 4x4. La compresión de información de movimiento consiste en mantener solamente la información de movimiento correspondiente al bloque 4x4 izquierdo superior **5.1** para un determinado bloque **5.0**.

50 El proceso de compresión de información de movimiento se puede producir en cuanto se ha realizado la elección final para un bloque de 16x16 y este se ha codificado, pero es más sencillo visualizarlo como realizándose una vez que ha sido codificada toda la imagen. Para mayor simplicidad, se puede considerar que este se lleva a cabo después del proceso de filtro de bucle adaptativo, y antes de que la imagen descodificada se ponga en el almacenamiento intermedio de imágenes descodificadas (DPB, decoded picture buffer). Este proceso de compresión se puede describir como una búsqueda particular: para determinadas coordenadas de píxel X e Y, la información de movimiento se obtiene de la posición $X' = (X \gg 4) \ll 4$ e $Y' = (Y \gg 4) \ll 4$, donde los operadores '>>' y '<<' se describen como sigue.

60 $x \gg y$ y representa el desplazamiento aritmético hacia la derecha de una representación de enteros de complemento a dos de x por y dígitos binarios. Esta función está definida solamente para valores enteros no negativos de y. Los bits desplazados en el MSB como resultado del desplazamiento hacia la derecha tienen un valor igual al MSB de x anterior a la operación desplazamiento.

65 $x \ll y$ y representa desplazamiento aritmético hacia la izquierda de una representación de enteros de complemento a dos, de x por y dígitos binarios. Esta función está definida solamente para valores enteros no negativos de y. Los bits desplazados en el LSB como resultado del desplazamiento hacia la izquierda tienen un valor igual a 0.

Cabe señalar que algunas implementaciones pueden utilizar un almacenamiento intermedio para almacenar el correspondiente movimiento comprimido.

En el diseño de HEVC, los vectores de movimiento son codificados mediante codificación predictiva, utilizando una serie de predictores de información de movimiento. Para un bloque INTER-codificado, existen 3 submodos, denominados submodos de codificación de bloques Skip, Inter y Merge. El submodo Inter utiliza un procedimiento particular de predicción del movimiento denominado AMVP, y utiliza datos de textura de residuos. Los submodos Skip y Merge utilizan el mismo procedimiento de predicción del movimiento (pero el primero no utiliza datos de residuo). Este procedimiento de predicción permite la selección del mejor predictor de información de movimiento a partir de un conjunto determinado, donde el conjunto se compone de información de movimiento espacial y temporal.

Se describirá el modo de predicción de información de movimiento denominado modo Merge, y cómo aplica a ambos enfoques mencionados anteriormente: *TextureRL* e *índice de cuadro de referencia*. Esto se utiliza para 2 submodos de codificación INTER: los submodos Skip y Merge. A continuación se detalla un esquema equivalente, que puede ser utilizado en modo AMVP.

La **figura 3** muestra un diagrama de flujo genérico para el esquema de predictor de información de movimiento Merge para los submodos Merge y Skip en el lado del codificador, denominado abreviadamente "el modo Merge". El principio del modo Merge es utilizar la predicción de vectores de movimiento para la compensación de movimiento sin codificar ningún refinamiento del movimiento. El módulo **3.3** de generación de predictores de información de movimiento genera el conjunto **3.4** de predictores de información de movimiento, en base al campo **3.1** de información de movimiento, como se describe en detalle a continuación. La selección **3.7** de tasa/distorsión del mejor predictor de información de movimiento se aplica entre el conjunto **3.4** de predictores de información de movimiento. Esto genera un índice **3.8** de predictor de vector de movimiento, que ha de codificarse.

El módulo **3.14** de conversión convierte dicho índice en un código unario truncado **3.15**: para un valor N a codificar, se genera una palabra de código de longitud $N+1$, excepto para el valor máximo de N , que requiere en cambio N bits. Este código consiste en N bits ajustados a 1, y un bit final ajustado a 0. Si el valor N es igual al número máximo de candidatos, entonces no es necesario el bit terminal, y la longitud de la palabra de código es por lo tanto N . Debido a dicho valor máximo, el número de candidatos Merge (normalmente 5 por HEVC), se puede seleccionar a nivel del segmento (elemento de sintaxis *five_minus_max_num_merge_cand* en HEVC), la etapa **3.14** tiene en cuenta el número máximo de predictor **3.16**.

La palabra **3.15** de código generada es a continuación codificada por entropía mediante la etapa **3.12** de codificación entrópica:

- el primer bit utiliza codificación aritmética con un contexto específico;
- los bits restantes utilizan codificación por derivación, es decir, se genera un bit real.

La **figura 4** muestra el diagrama de flujo para el correspondiente proceso de descodificación. En la primera etapa, el módulo **4.2** genera el conjunto **4.8** de predictores de información de movimiento en base al campo **4.1** de información de movimiento del cuadro actual y de los cuadros anteriores. El número máximo **4.16** de predictores de movimiento ha sido descodificado a partir del elemento de sintaxis *five_minus_max_num_merge_cand* situado en la cabecera del segmento. Este se utiliza a continuación en **4.6** para extraer la palabra de código **4.14** de predictor de información de movimiento. Esta palabra de código es convertida por la etapa **4.15** en un índice **4.9** de predictor. El predictor de información de movimiento **4.10** a utilizar se extrae a continuación del conjunto **4.8** de acuerdo con este valor **4.9** de índice de predictor. Este predictor se utiliza a continuación como la información de movimiento real durante compensación de movimiento.

Un predictor de información de movimiento o candidato contiene toda la información de movimiento: dirección (es decir, disponibilidad de un vector de movimiento e índice de referencia dentro de una lista), índice de cuadro de referencia y vectores de movimiento. Se generan varios candidatos mediante el proceso de obtención Merge descrito a continuación, teniendo cada uno un índice. En HEVC, el número máximo de candidatos Max_Cand es por defecto igual a 5, pero se puede reducir a 1. Se está describiendo en este caso la determinación de predictor de información de movimiento Merge, con partes específicas para los enfoques de *TextureRL* y de *índice de cuadro de referencia*.

La **figura 9** es el diagrama de flujo del proceso de obtención de información de movimiento de los modos Merge. En la primera etapa de la obtención, en los enfoques tanto de núcleo HEVC, de *TextureRL* como de *índice de cuadro de referencia*, se consideran 7 posiciones de bloque, **9.1** a **9.7**.

Además, en el caso del enfoque *TextureRL*, se considera otro candidato, el SMVP **9.0** (donde SMVP se refiere a predictor de vector de movimiento espacial (spatial motion vector predictor)), tal como se ha descrito en lo anterior. Estas posiciones son las posiciones espaciales y temporales representadas en la **figura 2**. Cada posición tiene el mismo nombre en ambas figuras. Este SMVP no existe en el enfoque de *índice de cuadro de referencia*.

El módulo **9.8** comprueba la disponibilidad de la información de movimiento espacial, y en el enfoque *TextureRL*, el SMVP asimismo para una capa de mejora. Este selecciona como mucho 4 predictores de información de movimiento. En este módulo, un predictor está disponible si existe en la capa de referencia y si este bloque no está codificado INTRA. Además, en lo que sigue, dentro del enfoque *TextureRL*, cualquier candidato a añadir es comparado asimismo con el SMVP además de cualquier otra información de movimiento, y es añadido de hecho solamente si es diferente. Por ejemplo, el candidato "izquierdo", designado A1 o 9.1, es comparado asimismo con el SMVP y añadido como segundo si existe movimiento en la posición X2, o como primero en caso contrario. Esta comparación, también en lo que sigue, se realiza comprobando que:

- 5
10
- la información de movimiento de dos candidatos tiene la misma dirección de predicción;
 - si es el caso, para cada elemento de información de movimiento asociado con la dirección de predicción:

o que se hace referencia al mismo cuadro (es decir, el mismo valor del índice ref_idx);
o que los vectores de movimiento son idénticos en sus coordenadas tanto vertical como horizontal.

15 La selección y la comprobación de estos 5 vectores de movimiento se describen en las condiciones siguientes:

• En el enfoque *TextureRL*, si la información de movimiento **9.0** de la posición X2 situada en la misma localización de la posición central X1 de la PU está disponible **9.8**, esta es escalada y utilizada como el primer candidato en la lista **9.10**.

• Si la información de movimiento "izquierdo" A1 **9.1** está disponible **9.8**, lo que significa que existe, y si este bloque no está codificado Intra, la información de movimiento del bloque "izquierdo" es seleccionada y utilizada como el primer candidato en la lista **9.10**.

• Si la información de movimiento "superior" B1 **9.2** está disponible **9.8**, el bloque "superior" candidato se compara **9.9** con A1 (si existe). Si B1 es igual a A1, B1 no se añade a la lista de candidatos espaciales **9.10**, de lo contrario se añade.

• Si la información de movimiento "superior derecho" B0 **9.3** está disponible **9.8**, el vector de movimiento de "superior derecho" se compara con **9.9** de B1. Si B0 es igual a B1, B0 no se añade a la lista de candidatos espaciales (**9.10**), de lo contrario se añade.

• Si el vector de movimiento "inferior izquierdo" A0 **9.4** está disponible **9.8**, la información de movimiento de "inferior izquierdo" se compara **9.9** con A1. Si A0 es igual a A1, A0 no se añade a la lista de candidatos espaciales **9.10**, de lo contrario se añade.

• Si la lista de candidatos espaciales no contiene 4 candidatos, se comprueba **9.8** la disponibilidad de la información de movimiento "superior izquierdo" B2 **9.5**, si está disponible; el vector de movimiento de "superior izquierdo" B2 se compara **9.9** con A1 y B1. Si B2 es igual a A1 o B1, B2 no se añade a la lista de candidatos espaciales **9.10**, de lo contrario se añade.

Al término de esta etapa, la lista **9.10** contiene de 0 a 4 candidatos.

45 Para el candidato temporal, se pueden utilizar dos posiciones: H **9.6** correspondiente a la posición inferior derecha BR1 del bloque situado en la misma localización, o el centro C **9.7** del bloque situado en la misma localización (situado en la misma localización significa el bloque en la misma posición en un cuadro temporalmente diferente) correspondiente a la posición central X1 del bloque actual. Estas posiciones están representadas en la **figura 2**.

50 En primer lugar, se verifica **9.11** la disponibilidad del bloque en la posición H **9.6**. Si no está disponible, entonces se verifica a continuación **9.11** el bloque en la posición central **9.7**. Si por lo menos está disponible una información de movimiento en estas posiciones, esta información de movimiento temporal se puede escalar si es necesario **9.12** para que sea homogénea con la información de movimiento procedente del cuadro de referencia con índice 0, para ambas listas L0 y L1 si es necesario, con el fin de crear el candidato temporal **9.13**; el candidato temporal es introducido a continuación en la lista de candidatos Merge inmediatamente después de los candidatos espaciales.

Además, la posición final para el candidato temporal, H o central dependiendo de la disponibilidad, está obligada a permanecer dentro del mismo CTB (que significa bloque de árbol de codificación (Coding Tree Block)) o su contiguo derecho con el fin de reducir los accesos a memoria.

60 Es importante observar que, para todas las capas y todos los enfoques, pero con la máxima importancia en la capa de referencia, este predictor de información de movimiento es determinado y añadido condicionalmente en función de:

- 65 - si dicho predictor temporal de información de movimiento (TMVP, temporal motion information predictor) está

deshabilitado a nivel de secuencia, por ejemplo utilizando el indicador `sps_temporal_mvp_enable_flag` localizado en el SPS (que significa conjunto de parámetros de secuencia (Sequence Parameter Set)) -esto es particularmente relevante para una realización de la invención;

- en caso de que esté habilitado a nivel de secuencia, si está inhabilitado a nivel de segmento, por ejemplo utilizando el indicador `enable_temporal_mvp_flag` localizado en la cabecera de segmento.

El hecho de que este predictor de información de movimiento se pueda inhabilitar, junto con cómo se ve afectado por la compresión de memoria de vectores de movimiento, desempeña un papel importante en el proceso descrito y en cómo se obtiene el predictor SMVP 9.0.

En segundo lugar, en el enfoque de *índice de cuadro de referencia*, este predictor temporal de información de movimiento puede proceder del cuadro insertado. Tal como se describirá a continuación, dicha información de movimiento se obtiene de hecho del campo de movimiento comprimido del cuadro de la capa de referencia.

Si la cantidad de candidatos (`Nb_Cand`) **9.14** es estrictamente inferior al número máximo de candidatos `Max_Cand`, 5 por defecto y como mucho, los candidatos combinados se generan en una etapa **9.15**, de lo contrario, la lista final de candidatos Merge se construye en una etapa **9.18**. El módulo **9.15** se utiliza solamente cuando el cuadro actual es un cuadro B, y genera varios candidatos en base a los candidatos disponibles en dos listas Merge en una etapa **9.15**. Esta generación consiste en la combinación de una información de movimiento de un candidato de la lista L0 con otra información de movimiento de un candidato diferente de la lista L1.

Si el número de candidatos (`Nb_Cand`) **9.16** es estrictamente inferior al número máximo de candidatos `Max_Cand`, se añaden candidatos con información de movimiento nula sin desplazamiento (0, 0) (es decir, los valores de los vectores de movimiento son todos iguales a cero), etapa **9.17**, incrementando `Nb_Cand`, hasta que `Nb_Cand` es igual a `Max_Cand`.

Al término de este proceso, la lista final de candidatos Merge está construida en una etapa **9.18**.

La especificación actual para SHVC (extensión escalable de HEVC) no utiliza un predictor de información de movimiento obtenido a partir de la capa de referencia en el modo AMVP, pero este se puede introducir como sigue.

La **figura 12** muestra la obtención del conjunto de predictores de información de movimiento AMVP. Este proceso se utiliza para codificar predictivamente la información de movimiento. En comparación con el modo Merge, es necesario transmitir información adicional: una dirección de predicción, y para cada elemento de información de movimiento a transmitir, se transmite asimismo un índice de cuadro de referencia, un índice de predictor y un residuo de información de movimiento.

El índice de predictor depende del número de candidatos: HEVC tiene que generar como máximo 2 predictores de información de movimiento. En ese caso, `Max_Cand` en esta figura se iguala a 2, pero se puede concebir utilizar 3 para una capa de mejora. El primer candidato espacial se selecciona entre los bloques izquierdos A0 **12.1** y A1 **12.2**, para posiciones similares a las del modo Merge.

Los dos predictores espaciales de información de movimiento del modo AMVP se escogen entre los anteriores y entre los bloques izquierdos incluyendo los bloques de esquina superior y el bloque de la esquina izquierda, de nuevo tal como para el modo Merge. Este predictor izquierdo Cand1 **12.9** es seleccionado **12.8** entre los bloques "inferior izquierdo" A0 e "izquierdo" A1. En este orden específico, se evalúan las siguientes condiciones hasta que se encuentra un valor de información de movimiento para la información de movimiento a partir de la misma lista de referencia y la misma imagen de referencia, o de la información de movimiento de la otra lista de referencia y de la misma imagen de referencia.

El anterior predictor de información de movimiento Cand2 **12.11** se selecciona en una etapa **12.10** entre "superior derecho" B0 **12.3**, "superior" B1 **12.4** y "superior izquierdo" B2 **12.5** en este orden específico, con las mismas condiciones que se describen a continuación. A continuación se compara Cand1 y Cand2 para eliminar uno de estos predictores de información de movimiento si son iguales **12.15**. Después de este recorte, el número de candidatos se compara con `Max_Cand` en una etapa **12.16**: si son iguales, entonces la lista de candidatos de información de movimiento AMVP está completamente determinada, y el proceso finaliza en la etapa **12.23**.

De lo contrario, si el número de candidatos está por debajo de `Max_Cand`, el predictor de movimiento temporal Cand3 **12.14** se obtiene tal como en el modo Merge y se añade, si existe, en una etapa **12.17**. Para ello, la posición inferior derecha (H) **12.6** se considera en primer lugar en el módulo **12.12** de comprobación de disponibilidad. Si esta no existe, se selecciona el centro del bloque situado en la misma localización **12.7**.

A continuación, se compara de nuevo el número de candidatos añadidos con el número máximo de candidatos, en una etapa **12.18**. Si se alcanzado este número máximo, se construye la lista final de predictores AMVP en la etapa **12.23**. De lo contrario, la etapa **12.19** comprueba si se está construyendo una lista para una capa de mejora. Si no, se reanuda la construcción de lista clásica en la etapa **12.22**, donde se añaden a la lista tantos candidatos como

como sea necesario para alcanzar el máximo, y por lo tanto esta se completa y finaliza en la etapa **12.23**. De lo contrario, se produce un proceso específico, donde el candidato SMVP se obtiene de la capa de referencia siguiendo el cálculo descrito a continuación. Cuando esto se ha realizado, se reanuda el procesamiento normal en la etapa **12.22**.

5 Se ha elegido mostrar la utilización de este candidato después de Cand3. Por supuesto, es obvio imaginar utilizarlo antes de Cand3, entre Cand1 y Cand2 o entre Cand2 y Cand3. En todos los casos, se puede realizar una comparación adicional en el proceso de recorte para tener en cuenta el nuevo candidato SMVP.

10 Cuando se considera aplicar el modo Merge al enfoque *TextureRL*, el modo Merge añade un nuevo predictor de información de movimiento, el SMVP, en la capa de mejora obtenida a partir de su capa de referencia. Dicho predictor de información de movimiento procede actualmente de la información de movimiento utilizada para determinar el candidato temporal en la capa de referencia, que está comprimida. La **figura 6** muestra esquemáticamente los principios del enfoque *TextureRL*. La descripción se refiere en este caso al componente de luma de la imagen, pero el proceso aplica también a los componentes de color croma.

15 En la **figura 6** se representa una imagen **6.1** de la capa de mejora y su imagen **6.2** de la capa de referencia, con una relación espacial de R (habitualmente 1, 1,5 o 2) entre **6.1** y **6.2**. Cualquiera que sea el valor de R , se utiliza la expresión "escalamiento ascendente" como el proceso de remuestreo aplicado a una capa de referencia para que se corresponda con las dimensiones de la capa de mejora. Si R es igual a 1, el remuestreo produce una salida idéntica a la entrada. El remuestreo inverso se denomina "escalamiento descendente". La imagen de capa de mejora está subdividida en una parrilla que representa la granularidad de la imagen. Cada uno de los cuadrados menores se denomina una muestra en el siguiente texto.

25 A continuación, para una determinada unidad de predictor **6.3**, representada con un cuadrado de línea gruesa, el proceso consiste en:

A. Calcular la localización central **6.4** ($xPCtr$, $yPCtr$) del bloque de predicción de luma considerado **6.3**, que se obtiene como sigue

30 o

$$xPCtr = xP + nPbW / 2$$

35 o

$$yPCtr = yP + nPbH / 2$$

40 ▪ xP , yP especifican la muestra superior izquierda **6.6** del bloque de predicción de luma actual con respecto a la muestra **6.7** de luma superior izquierda de la imagen actual.
 ■ $nPbW$ y $nPbH$ especifican la anchura y la altura del bloque de predicción de luma.

45 B. Realizar escalamiento descendente de sus coordenadas de acuerdo con el factor de escala R (1, 1,5 y 2,0) para encontrar la posición **6.5** situada en la misma localización en la imagen **6.2** de la capa de referencia;

o

$$xPCtrRL = (xPCtr * PicWRL + ScaledW / 2) / ScaledW$$

50 o

$$yPCtrRL = (yPCtr * PicHRL + ScaledH / 2) / ScaledH$$

55 ▪ Las variables $PicWRL$ y $PicHRL$ se igualan a la anchura y la altura de la imagen de la capa de referencia.

▪ El valor de $ScaledH$ adopta el valor de $R * PicHRL$ y el valor de $ScaledW$ es igual al valor de $R * PicWRL$.

60 C. Recuperar la información de movimiento en dicha localización a partir de la imagen **6.2** de la capa de referencia, identificando un bloque de predicción de luma $bIPb$ **6.8**, con el número 1, que abarca la localización modificada proporcionada por $((xPCtrRL >> 4) << 4, (yPCtrRL >> 4) << 4)$ en el interior de la imagen de la capa de referencia. Esto corresponde a la etapa de integración de movimiento de la capa de referencia.

o La localización luma ($xPRL$, $yPRL$) se iguala entonces a la muestra superior izquierda **6.8** del bloque de predicción de luma situado en la misma localización, especificado por $bIPb$ con respecto a la muestra luma superior izquierda de la imagen de la capa de referencia.

65 D. Si la información correspondiente no es intra, extraer los vectores de movimiento MV_{RL} y someterlos a

escalamiento ascendente de acuerdo con la relación R. La operación genera básicamente el vector de movimiento espacial cuyo valor SMVP es igual a: $SMVP = \text{rnd}(R * MV_{RL}(\text{rnd}(xPRL/R), \text{rnd}(yPRL/R)))$, donde $\text{rnd}(\cdot)$ representa un proceso de redondeo. Junto con los índices de cuadro de referencia asociados a este vector de movimiento espacial, esto constituye el predictor de vector de movimiento espacial introducido al inicio del conjunto.

El actual diseño de SHVC para la *TextureRL* exige que los vectores de movimiento para los valores de la capa de referencia MV_{RL} se obtengan del almacenamiento intermedio de compresión de movimiento de la capa de referencia, tal como se ve en la etapa C anterior. Esto es necesario debido a que la información de movimiento correspondiente al bloque 4x4 **6.8** es la única mantenida para el bloque de 16x16 íntegro por el proceso de compresión de información de movimiento.

Considerando a continuación el enfoque de *índice de cuadro de referencia*, en ese caso, la información de movimiento del nuevo cuadro, que se introduce en la lista de referencia de la capa de mejora, procede también de dicho campo de información de movimiento comprimido. Dicha información de movimiento puede ser utilizada a continuación para determinar el predictor temporal, tal como se ha descrito anteriormente.

Se detalla a continuación cómo se obtiene este movimiento. Para un determinado bloque de 16x16, se selecciona el centro de dicho bloque, y esta posición se utiliza equivalentemente a la que se ha descrito anteriormente para encontrar la correspondiente información de movimiento. Se van a detallar las etapas correspondientes para el componente luma. Cabe destacar que la mayoría de las partes son esencialmente idénticas al proceso descrito en relación con la **figura 6**, y se mantienen las definiciones para las variables idénticas. Estas son:

A. La localización central (xPCtr, yPCtr) del bloque de predicción de luma se obtiene como sigue (los nombres de variables se definen en la sección anterior):

$$xPCtr = xP + 8$$

$$yPCtr = yP + 8$$

B. Realizar el escalamiento descendente de sus coordenadas de acuerdo con el factor de escala R (1, 1,5 y 2,0) para encontrar la posición situada en la misma localización en la imagen de la capa de referencia;

$$xPCtrRL = (xPCtr * PicWRL + ScaledW / 2) / ScaledW$$

$$yPCtrRL = (yPCtr * PicHRL + ScaledH / 2) / ScaledH$$

Las variables PicWRL y PicHRL se igualan a la anchura y la altura de la imagen de la capa de referencia. El valor de ScaledH adopta el valor de $R * PicHRL$ y el valor de ScaledW es igual al valor de $R * PicWRL$.

C. La posición situada en la misma localización (xRL, yRL) se obtiene como sigue

$$xRL = (xRef \gg 4) \ll 4$$

$$yRL = (yRef \gg 4) \ll 4.$$

D. El vector de movimiento de la capa de referencia se obtiene como sigue. Básicamente, la operación genera los vectores de movimiento del valor RL_MV de la capa de referencia como sigue: $RL_MV = \text{rnd}(R * MV(\text{rnd}(xPRL/R), \text{rnd}(yPRL/R)))$.

Esta información se utiliza a continuación como si fuera la salida de la compresión de información de movimiento. Esto permite utilizar la información de movimiento de la capa de referencia para predecir información de movimiento en la capa de mejora. A diferencia del enfoque *TextureRL*, la información de movimiento tiene una granularidad más gruesa pero puede ser utilizada como un predictor temporal de información de movimiento, tanto para el proceso de determinación de listas Merge como para el de AMVP utilizado para bloques inter.

Una vez se ha presentado la estructura global de un códec escalable, los dos enfoques y cómo estos utilizan información de movimiento para predicción, se puede resumir lo siguiente.

En el enfoque *TextureRL*, un nuevo predictor de información de movimiento en la capa de mejora se obtiene a partir de su capa de referencia. Dicho predictor de información de movimiento procede habitualmente de la información de movimiento utilizada para determinar el candidato temporal en la capa de referencia, que está comprimida. Por lo tanto, la compresión afecta a la obtención de este, y por lo tanto al modo Merge. En el modo AMVP, si se presenta un candidato escalable, la compresión también se vería afectada. Los modos AMVP y Merge en el enfoque de *índice de cuadro de referencia* siempre se ven afectados, dado que también utilizan el predictor de vector de movimiento temporal y, si el cuadro al que se hace referencia es el que se ha introducido, este predictor procedería entonces del

movimiento del cuadro de la capa de referencia.

En el enfoque de *índice de cuadro de referencia*, la información de movimiento del nuevo cuadro, que se introduce en la lista de referencia de la capa de mejora, procede también de dicha información de movimiento comprimido.

Tal como se ha explicado en relación con la **figura 6**, la información de movimiento almacenada en la capa de referencia está comprimida. Esto significa que para un bloque de 16x16 completo, donde existen inicialmente hasta 16 elementos de información de movimiento, uno para cada bloque 4x4 contenido dentro del bloque de 16x16, se mantiene solamente uno, habitualmente el relativo al bloque 4x4 superior izquierdo.

En el proceso de obtención de predictores de información de movimiento cuando se requiere información de movimiento de la capa de referencia, debido a esta compresión la información de movimiento utilizada es la que está disponible, es decir la información de movimiento asociada con el bloque 4x4 superior izquierdo. Haciendo referencia de nuevo a la **figura 6**, cuando se busca la información de movimiento asociada con el punto **6.5** situado en la misma localización, correspondiente al centro **6.4** de la unidad de codificación a codificar, se utiliza la información de movimiento asociada con el bloque 4x4 superior izquierdo **6.8** que ha recibido el número 1. Se debe observar que la información de movimiento asociada con el bloque 4x4 superior izquierdo que ha recibido el número 3, correspondiente a la información de movimiento mantenida después de la compresión para el bloque de 16x16 situado debajo, está más cerca de la posición del punto **6.5** situado en la misma localización, y por lo tanto es probable que sea más relevante que la información de movimiento del bloque 4x4 **6.8**.

Se puede medir que esta elección no óptima de la información de movimiento debido al proceso de compresión aplicado a la información de movimiento en la capa de referencia provoca una pérdida de eficiencia de codificación. En una realización de la invención, el proceso de obtención del predictor de información de movimiento está adaptado para superar este problema de posición.

La **figura 7** detalla el proceso adaptado en el contexto del enfoque *TextureRL*. Este se puede aplicar en el proceso de obtención de tanto AMVP como Merge, en una capa de mejora. Este proceso modificado de obtención Merge puede estar localizado en el módulo **1.23** de estimación de movimiento en la **figura 1** del codificador, y del módulo **10.55** de estimación de movimiento en la **figura 10** del decodificador. Esencialmente, todo esto ocurre cuando se determina el candidato de SMVP **9.0** en la **figura 9**.

La etapa **7.1** inicializa el proceso calculando la posición para la que determinar el movimiento en la capa de referencia, por ejemplo ajustando la información de unidad de predictor actual (dimensiones/posición) y obteniendo el centro de dicha unidad de predictor. La adaptación principal reside en la etapa **7.3**, que corrige la posición. Esto se realiza en primer lugar por medio de las siguientes 2 posibilidades.

En una primera realización, para una determinada coordenada X obtenida para la posición en la capa de referencia, por ejemplo xPCtrRL o yPCtrRL descritas anteriormente, se calcula un nuevo valor realizando una operación de redondeo de acuerdo con dos parámetros r y M.

Por ejemplo, el nuevo X' se puede calcular como sigue:

$$X' = \lfloor ((X + r)/M) \rfloor * M;$$

Donde $\lfloor x \rfloor$ representa el truncamiento de x, que significa tomar su parte entera. M puede ser una potencia de 2, en esta realización, $M = 16$ para corresponderse a la granularidad del movimiento comprimido HEVC. En esta realización, se utiliza $r = 4$ y no la elección más natural $r = 8$, ya que proporciona una mejor eficiencia de codificación.

Lo mismo se puede aplicar a la otra coordenada. Solamente como ejemplo comparativo, es posible elegir diferentes valores de los parámetros r y M.

En un ejemplo comparativo, la corrección de la posición se puede basar en una tabla de consulta. En este caso, dadas unas coordenadas (X, Y), se puede definir una tabla de correcciones F[X, Y] para, por lo menos, una de las coordenadas. Esta tabla puede ser diferente para cada coordenada. La tabla puede asimismo indexarse solamente mediante una de las coordenadas, es decir, X o Y. La tabla puede asimismo reducirse utilizando como índice un valor relativo a la coordenada en lugar de la propia coordenada, por ejemplo, la corrección se puede obtener mediante $F[X \bmod M]$ en lugar de F[X], donde $M = 2^N$ es un valor típico. En un ejemplo, $M = 16$.

En todos los casos, el valor correctivo (ya sea un valor de r o una tabla para por lo menos un componente) se puede transmitir en información de sintaxis de alto nivel y recuperarse de la misma, por ejemplo en el conjunto de parámetros de vídeo, el conjunto de parámetros de secuencia, el conjunto de parámetros de imagen o la cabecera de segmento. En el caso de transmitirse por lo menos un valor de r:

- un indicador de bit puede indicar si el valor de r es un primer o un segundo valor, por ejemplo 0 y 4 (en cuyo caso, se puede ver como un indicador activado/desactivado para la corrección);
- un código puede indicar el valor explícito de r , por ejemplo un código unario truncado que representa el valor de r menos 4, por ejemplo secuencias binarias '0' para $r = 4$, '10' para $R = 5$, '110', '1110' y '1111' para los otros valores.

5 Es importante observar en lo anterior que solamente puede estar afectada una coordenada, en particular la abscisa, dado que modificar la ordenada conduciría a recuperar información de movimiento de un área de memoria diferente y provocaría, por lo tanto, accesos a memoria adicionales.

10 Siguiendo este requisito de reducir los accesos a memoria, por lo menos un valor corregido se puede modificar a otro valor, siendo posiblemente este otro valor el valor original, si dicho valor corregido no se corresponde con un criterio, tal como satisfacer un umbral. Dicho umbral puede ser la dimensión de la imagen a lo largo de la coordenada, de tal modo que no se pueda producir ninguna búsqueda fuera de la imagen. Alternativamente, dicho umbral puede ser el límite de un área de memoria a lo largo de dicha coordenada. El área de memoria corresponde habitualmente a un conjunto predeterminado de la mayor unidad de codificación en la capa de referencia. Esta área de memoria se mostrará con mayor detalle en la **figura 13**.

20 A continuación se reanuda la determinación normal de predictores en la etapa **7.4**. La información de movimiento se recupera del almacenamiento intermedio de movimiento comprimido utilizando la posición de salida de la etapa **7.3**. Si esta es intra (es decir, no existe movimiento), el candidato se marca como tal en la etapa **7.8**, en particular no calculando ni añadiendo ningún predictor a la lista de candidatos Merge, finalizando por lo tanto el proceso de obtención en la etapa **7.9**. De lo contrario, se realiza un escalamiento ascendente del correspondiente movimiento para corresponderse con las dimensiones de la capa de mejora.

25 La **figura 8** muestra el proceso adaptado en el contexto del enfoque de *índice de cuadro de referencia*. Este se puede aplicar en el proceso de obtención de tanto AMVP como Merge en una capa de mejora. Este proceso adaptado está localizado en el almacenamiento intermedio **1.24** de cuadros o bien en el módulo **1.23** de estimación de movimiento en el codificador de la **figura 1**, y en el almacenamiento intermedio **10.60** de cuadros o en el módulo **10.55** de estimación de movimiento del descodificador en la **figura 10**. De hecho, afecta al contenido de la memoria de cuadros en relación con la información de movimiento comprimido.

30 Por lo tanto, la etapa **8.1** inicializa el proceso de obtención de predictores de información de movimiento ajustando el bloque de 16x16 actual al primero en la imagen de capa de mejora. En la etapa **8.2** se determina la posición del centro de la unidad de codificación 16x16, y en la etapa **8.3** se encuentra la correspondiente posición situada en la misma localización en la capa de referencia. La nueva etapa **8.4** donde se encuentra la posición está corregida. Se puede hacer referencia a la anterior etapa **7.1** para ver los detalles de la corrección, aplicando lo mismo en este caso.

35 En la etapa **8.5** se comprueba si el movimiento en dicha posición es intra. En caso afirmativo, el movimiento del bloque de 16x16 se ajusta a continuación como intra en la etapa **8.7**, de lo contrario, los vectores de movimiento son obtenidos y escalados ascendentemente para corresponderse con las dimensiones de la capa de mejora, y los vectores de movimiento sometidos a escalamiento ascendente, los índices de referencia y las disponibilidades se ajustan como los predictores de información de movimiento del actual bloque de 16x16 en la etapa **8.8**.

40 La etapa **8.9** prepara la siguiente iteración verificando si el bloque actual es el último en la imagen. Si este es el caso, la información de movimiento para el nuevo cuadro está entonces completamente determinada, y el proceso finaliza en la etapa **8.11**. De lo contrario, el bloque actual se ajusta al siguiente bloque de 16x16 en la etapa **8.10**, y la iteración vuelve a la etapa **8.2**.

45 La **figura 13** muestra los detalles de lo que se define como un área de memoria. Dada una imagen **13.4** de la capa de referencia y su imagen asociada **13.5** de la capa de mejora, se puede definir un área **13.6** para la que aplicar el proceso de determinación de movimiento, por ejemplo un CTB. Se puede aplicar una restricción de accesos a memoria (por ejemplo, en caso de un codificador canalizado, o de descodificación donde se procesan CTB de la capa de mejora inmediatamente después de los CTB de la capa de referencia situados en la misma localización), y por lo tanto se puede definir en un primer aspecto de la invención un área **13.1** situada en la misma localización, en el interior del cuadro **13.4** de la capa de referencia. El área de memoria mencionada en las etapas **7.1** y **8.4** corresponde en un primer aspecto al área **13.2** que contiene **13.1**, en este caso compuesta de 2 CTB de la capa de referencia: las posiciones corregidas encontradas para cualquier parte de **13.6** tienen que permanecer dentro del área **13.2**. De manera menos restrictiva, se puede imponer que el área de memoria puede comprender una columna adicional de CTB a la derecha del área **13.2**, teniendo como resultado el área **13.3**. Se puede comprender que la limitación se basa en este caso en **13.6**, pero se puede utilizar cualquier tamaño de área en la capa de mejora, o memoria aumentada en la capa de referencia.

50 La **figura 11** es un diagrama de bloques esquemático de un dispositivo informático **11.0** para la implementación de una o varias realizaciones de la invención. El dispositivo informático **11.0** puede ser un dispositivo tal como un microordenador, una estación de trabajo o un dispositivo portátil ligero. El dispositivo informático **11.0** comprende un

bus de comunicación conectado a:

- una unidad **11.1** central de proceso, tal como un microprocesador, designada CPU;
- una memoria **11.2** de acceso aleatorio, designada RAM, para almacenar código ejecutable del procedimiento de las realizaciones de la invención así como los registros adaptados para registrar variables y parámetros necesarios para implementar el procedimiento para codificar o descodificar por lo menos parte de una imagen, según las realizaciones de la invención, pudiendo expandirse la capacidad de memoria de la misma mediante una RAM opcional conectada a un puerto de expansión, por ejemplo;
- una memoria **11.3** de sólo lectura, designada ROM, para almacenar programas informáticos para implementar realizaciones de la invención;
- una interfaz **11.4** de red está conectada habitualmente a una red de comunicación sobre la que son transmitidos o recibidos datos digitales que tienen que ser procesados. La interfaz **11.4** de red puede ser una única interfaz de red, o componerse de un conjunto de diferentes interfaces de red (por ejemplo, interfaces cableadas e inalámbricas, o diferentes clases de interfaces cableadas o inalámbricas). Los paquetes de datos se escriben en la interfaz de red para transmisión o son leídos desde la interfaz de red para recepción, bajo el control de la aplicación de software que corre en la CPU **11.1**;
- una interfaz **11.5** de usuario se puede utilizar para recibir entradas de un usuario o para presentar información a un usuario;
- se puede disponer un disco duro **11.6** designado HD, como un dispositivo de almacenamiento masivo;
- se puede utilizar un módulo de E/S **11.7** para recibir/enviar datos desde/hacia dispositivos externos, tales como una fuente de vídeo o una pantalla.

El código ejecutable se puede almacenar en memoria **11.3** de sólo lectura, en el disco duro **11.6** o en un medio digital extraíble, tal como por ejemplo un disco. Según una variante, el código ejecutable de los programas puede ser recibido por medio de una red de comunicación, mediante la interfaz **11.4** de red, para ser almacenado en uno de los medios de almacenamiento del dispositivo **11.0** de comunicación, tal como el disco duro **11.6**, antes de ser ejecutado.

La unidad **11.1** central de proceso está adaptada para controlar y dirigir la ejecución de las instrucciones o de partes de código de software del programa o programas acordes con las realizaciones de la invención, instrucciones que están almacenadas en uno de los medios de almacenamiento mencionados anteriormente. Después del encendido, la CPU **11.1** puede ejecutar instrucciones de la memoria RAM principal **11.2** en relación con una aplicación de software, después de que dichas instrucciones han sido cargadas desde la ROM **11.3** de programas o desde el disco duro (HD) **11.6**, por ejemplo. Dicha aplicación de software, cuando es ejecutada por la CPU **11.1**, hace que se lleven a cabo las etapas de los diagramas de flujo mostrados en las figuras 1 a 4.

Cualquier etapa del algoritmo mostrado en la **figura 7** se puede implementar en software mediante la ejecución de un conjunto de instrucciones o de un programa mediante una máquina informática programable, tal como un PC ("Personal Computer", ordenador personal), un DSP ("Digital Signal Processor", procesador de señal digital) o un microcontrolador, o implementarse en hardware mediante una máquina o un componente dedicado, tal como una FPGA ("Field-Programmable Gate Array", matriz de puertas programables in situ) o un ASIC ("Application-Specific Integrated Circuit", circuito integrado de aplicación específica).

Aunque la presente invención se ha descrito en lo anterior haciendo referencia a realizaciones específicas, la presente invención no se limita a las realizaciones específicas, y para un experto en la materia serán evidentes modificaciones que quedan dentro del alcance de la presente invención.

Se ocurrirán muchas modificaciones y variaciones a los expertos en la materia tras hacer referencia a las anteriores realizaciones ilustrativas, que se proporcionan solamente a modo de ejemplo y que no están destinadas a limitar el alcance de la invención, que está determinado exclusivamente por las reivindicaciones adjuntas. En particular, las diferentes características de diferentes realizaciones se pueden intercambiar, cuando sea adecuado.

En las reivindicaciones, la expresión "que comprende" no excluye otros elementos o etapas, y el artículo indefinido "un" o "una" no excluye una pluralidad. El mero hecho de que se enuncien características diferentes en realizaciones dependientes diferentes entre sí no indica que no se pueda utilizar ventajosamente una combinación de estas características YY2.

REIVINDICACIONES

1. Procedimiento de codificación de una imagen según un formato de codificación escalable, comprendiendo dicho formato de codificación escalable una capa de referencia y una capa de mejora, siendo codificada predictivamente como mínimo parte de la imagen en base a información de movimiento, siendo codificada predictivamente dicha propia información de movimiento en base a un conjunto de predictores de información de movimiento, en el que el procedimiento comprende:
- 5 determinar coordenadas de una muestra de componente de color en la capa de referencia; y
 10 determinar un conjunto de predictores de información de movimiento que incluye un predictor de información de movimiento basado en información de movimiento asociada con un área de imagen perteneciente a la capa de referencia,
caracterizado por que determinar las coordenadas comprende:
- 15 obtener una posición central en un área de imagen para la capa de mejora;
 determinar, utilizando la posición central, una posición correspondiente en la capa de referencia; y
 corregir por lo menos una coordenada X de dicha posición correspondiente, a un nuevo valor X' proporcionado por $X' = \lfloor ((X + 4)/16) \rfloor * 16$, donde $\lfloor ((X + 4)/16) \rfloor$ es un truncamiento de $((X + 4)/16)$ a un valor entero,
- 20 y determinar el conjunto comprende, si está disponible, seleccionar información de movimiento asociada con la como mínimo una coordenada corregida en la capa de referencia para incluirla como un predictor de información de movimiento en dicho conjunto de predictores de información de movimiento,
 en el que la información de movimiento seleccionada es información de movimiento disponible a partir de un bloque 16 x 16 de una imagen perteneciente a la capa de referencia.
- 25 2. Procedimiento de descodificación de una imagen según un formato de codificación escalable, comprendiendo dicho formato de codificación escalable una capa de referencia y una capa de mejora, siendo codificada predictivamente como mínimo parte de la imagen en base a información de movimiento, siendo codificada predictivamente dicha propia información de movimiento en base a un conjunto de predictores de información de movimiento, en el que el procedimiento comprende:
- 30 determinar coordenadas de una muestra de componente de color en la capa de referencia; y
 determinar un conjunto de predictores de información de movimiento que incluye un predictor de información de movimiento basado en información de movimiento asociada con un área de imagen perteneciente a la capa de referencia,
 35 **caracterizado por que** la determinación de las coordenadas comprende:
- obtener una posición central en un área de imagen para la capa de mejora;
 determinar, utilizando la posición central, una posición correspondiente en una capa de referencia; y
 40 corregir como mínimo una coordenada X de dicha posición correspondiente a un nuevo valor X' proporcionado por $X' = \lfloor ((X + 4)/16) \rfloor * 16$, donde $\lfloor ((X + 4)/16) \rfloor$ es un truncamiento de $((X + 4)/16)$ a un valor entero,
- y la determinación del conjunto comprende, si está disponible, seleccionar información de movimiento asociada con la como mínimo una coordenada corregida en la capa de referencia para incluirla como un predictor de información de movimiento en dicho conjunto de predictores de información de movimiento,
 45 en el que la información de movimiento seleccionada es información de movimiento disponible a partir de un bloque 16 x 16 de una imagen perteneciente a la capa de referencia.
3. Procedimiento según la reivindicación 1 o la reivindicación 2, en el que la como mínimo una coordenada X comprende una coordenada horizontal x o una coordenada vertical y.
- 50 4. Dispositivo de codificación de una imagen según un formato de codificación escalable, comprendiendo dicho formato de codificación escalable una capa de referencia y una capa de mejora, siendo codificada predictivamente como mínimo parte de la imagen en base a información de movimiento, siendo codificada predictivamente dicha propia información de movimiento en base a un conjunto de predictores de información de movimiento, donde el dispositivo comprende:
- 55 un módulo de determinación de la posición para determinar coordenadas de una muestra de componente de color en la capa de referencia; y
 60 un módulo de determinación de predictores para determinar un conjunto de predictores de información de movimiento que incluye un predictor de información de movimiento basado en información de movimiento asociada con un área de imagen perteneciente a la capa de referencia,
caracterizado por que el módulo de determinación de la posición está configurado para:

obtener una posición central en un área de imagen para la capa de mejora;
 determinar, utilizando la posición central, una posición correspondiente en la capa de referencia; y
 corregir por lo menos una coordenada X de dicha posición correspondiente a un nuevo valor X' proporcionado por
 $X' = \lfloor ((X + 4)/16) \rfloor * 16$, donde $\lfloor ((X + 4)/16) \rfloor$ es un truncamiento de $((X + 4)/16)$ a un valor entero,

5 y el módulo de determinación de predictores está configurado para, si está disponible, seleccionar información de movimiento asociada con la como mínimo una coordenada corregida en la capa de referencia para incluirla como un predictor de información de movimiento en dicho conjunto de predictores de información de movimiento, en el que la información de movimiento seleccionada es información de movimiento disponible a partir de un bloque 10 16 x 16 de una imagen perteneciente a la capa de referencia.

5. Dispositivo de descodificación de una imagen según un formato de codificación escalable, comprendiendo dicho formato de codificación escalable una capa de referencia y una capa de mejora, siendo codificada predictivamente como mínimo parte de la imagen en base a información de movimiento, siendo codificada predictivamente dicha propia información de movimiento en base a un conjunto de predictores de información de movimiento, en el que el dispositivo comprende:

un módulo de determinación de la posición para determinar coordenadas de una muestra de componente de color en la capa de referencia; y
 20 un módulo de determinación de predictores para determinar un conjunto de predictores de información de movimiento que incluye un predictor de información de movimiento basado en información de movimiento asociada con un área de imagen perteneciente a la capa de referencia,
caracterizado por que el módulo de determinación de la posición está configurado para:

25 obtener una posición central en un área de imagen para la capa de mejora;
 determinar, utilizando la posición central, una posición correspondiente en una capa de referencia; y
 corregir como mínimo una coordenada X de dicha posición correspondiente a un nuevo valor X' proporcionado por
 $X' = \lfloor ((X + 4)/16) \rfloor * 16$, donde $\lfloor ((X + 4)/16) \rfloor$ es un truncamiento de $((X + 4)/16)$ a un valor entero,

30 y el módulo de determinación de predictores está configurado para seleccionar información de movimiento asociada con la como mínimo una coordenada corregida en la capa de referencia para incluirla como un predictor de información de movimiento en dicho conjunto de predictores de información de movimiento, en el que la información de movimiento seleccionada es información de movimiento disponible a partir de un bloque 35 16 x 16 de una imagen perteneciente a la capa de referencia.

6. Dispositivo según la reivindicación 4 o la reivindicación 5, en el que la como mínimo una coordenada X comprende una coordenada horizontal x o una coordenada vertical y.

7. Producto de programa informático para un aparato programable, comprendiendo el producto de programa informático una secuencia de instrucciones para implementar un procedimiento según cualquiera de las reivindicaciones 1 a 3, cuando se carga en un aparato programable y es ejecutado por el mismo.

8. Medio de almacenamiento legible por ordenador que almacena instrucciones de un programa informático según la reivindicación 7.

45

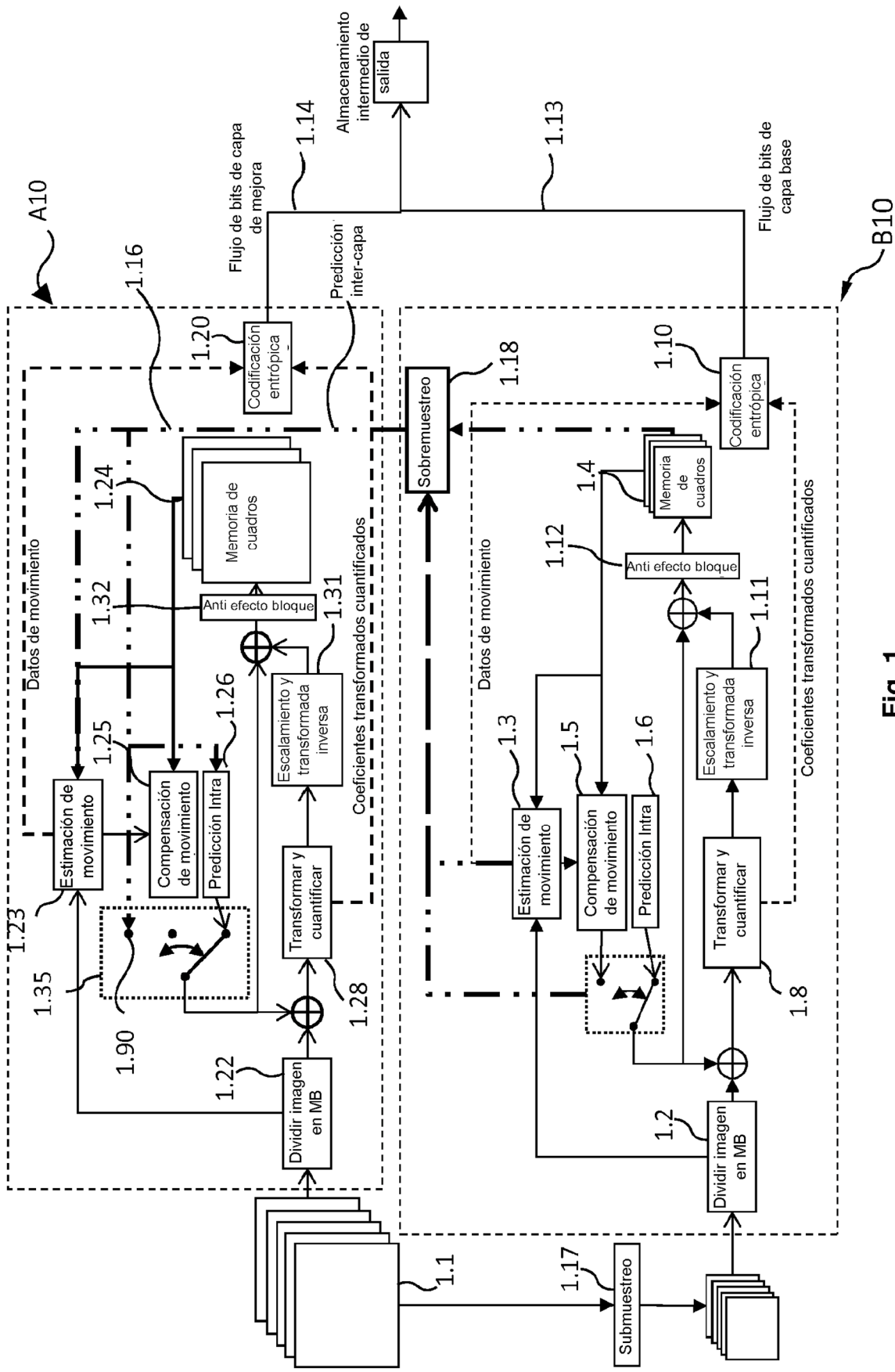


Fig. 1

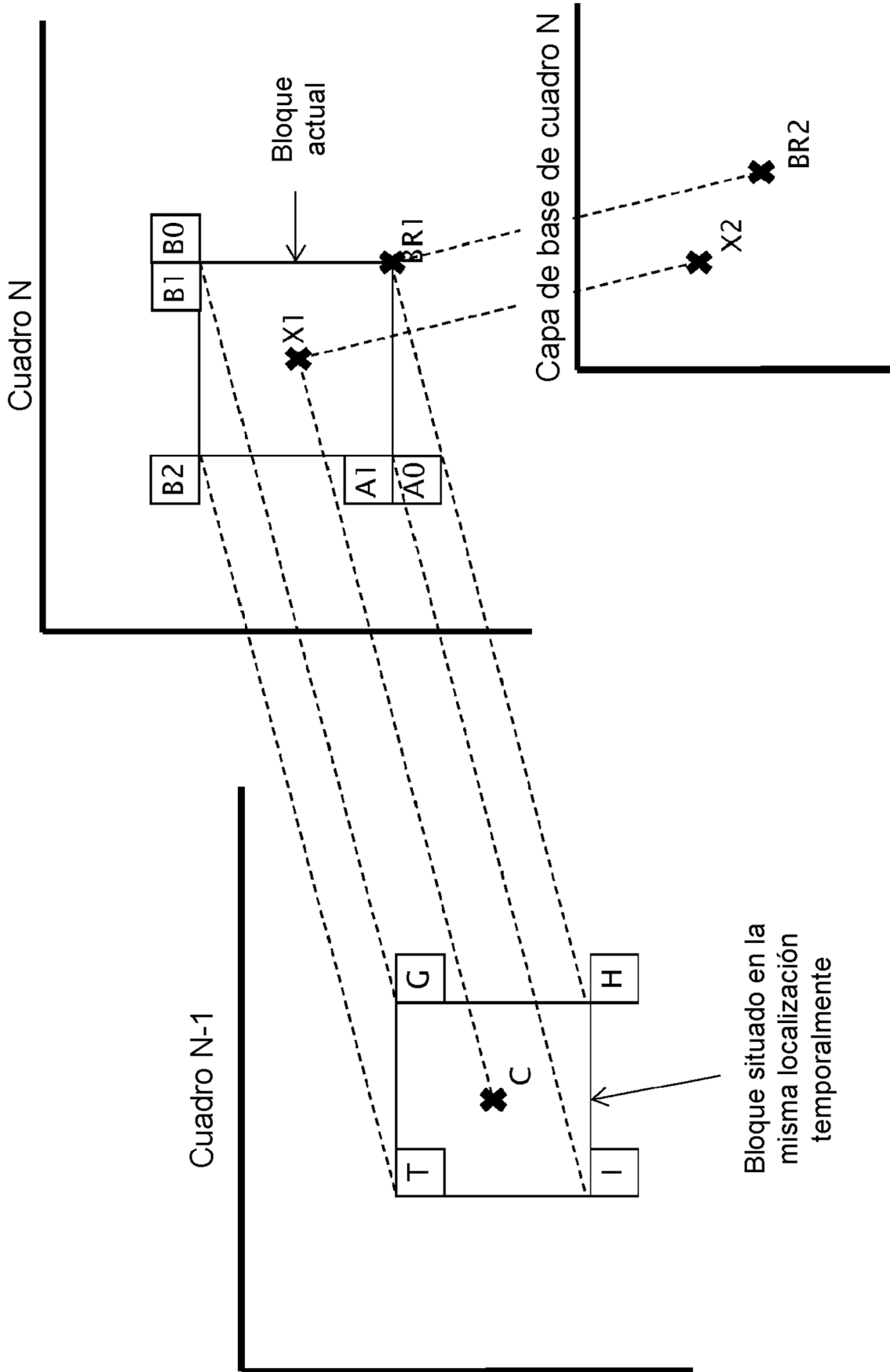


Fig. 2

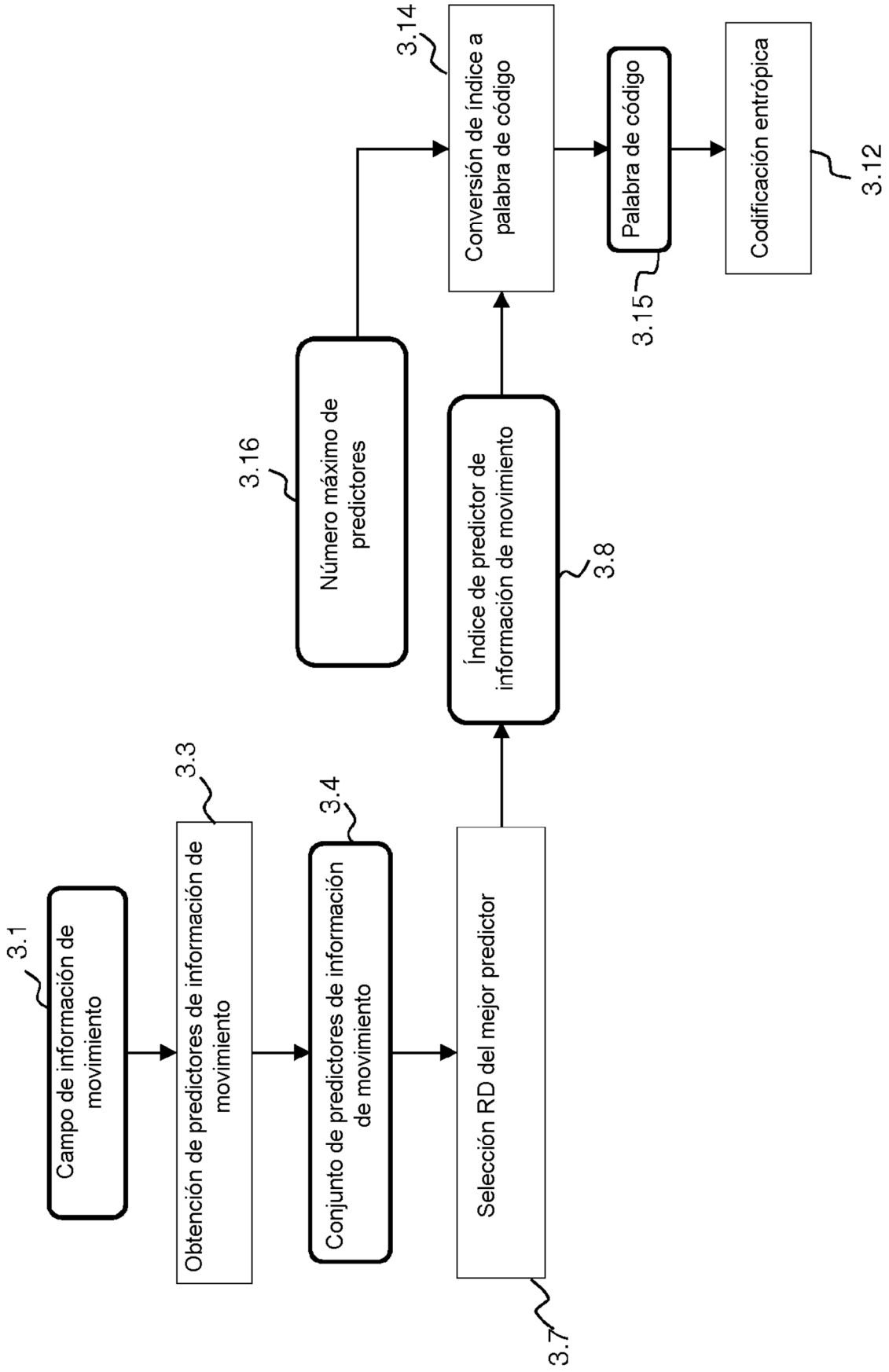


Fig. 3

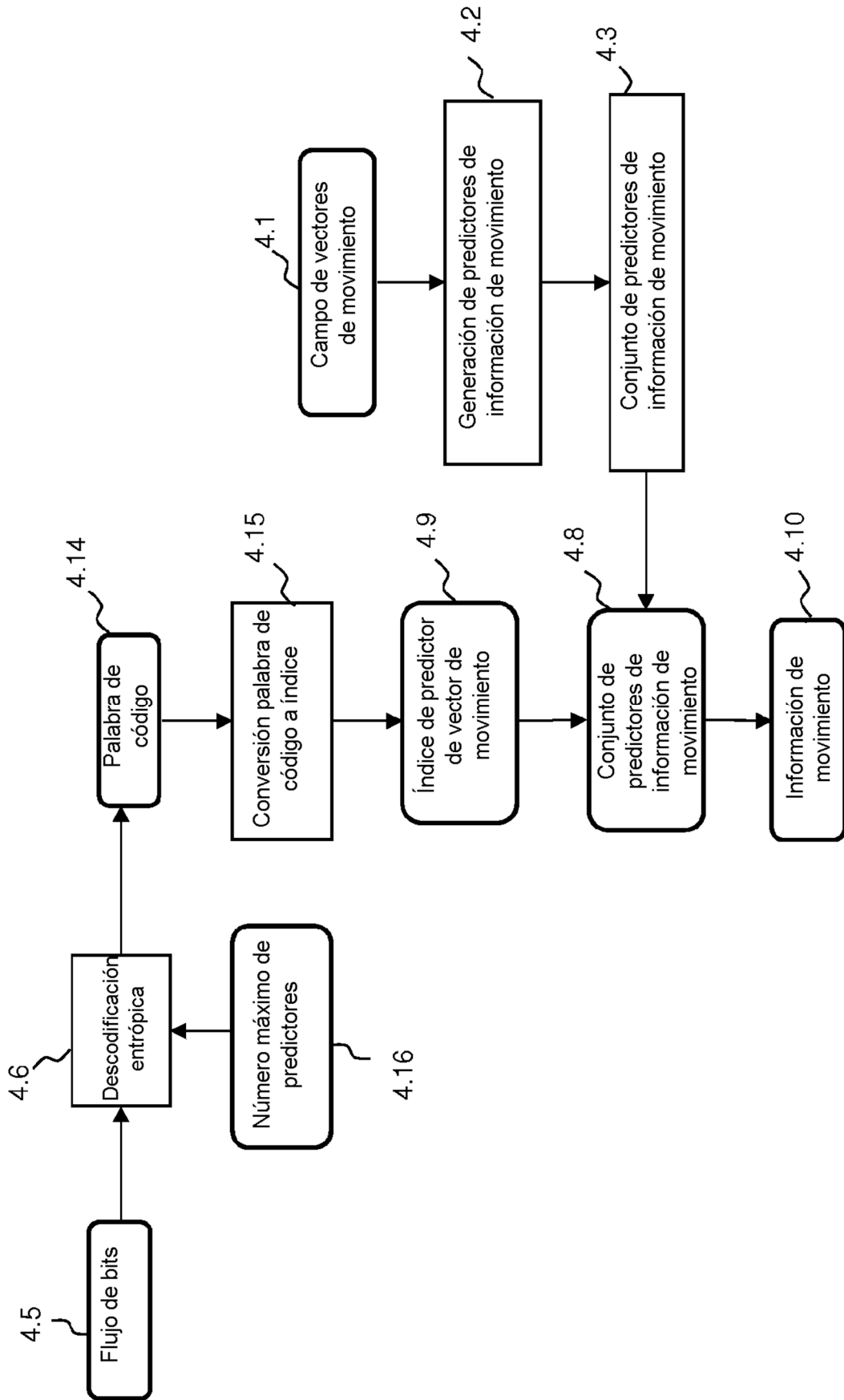


Fig. 4

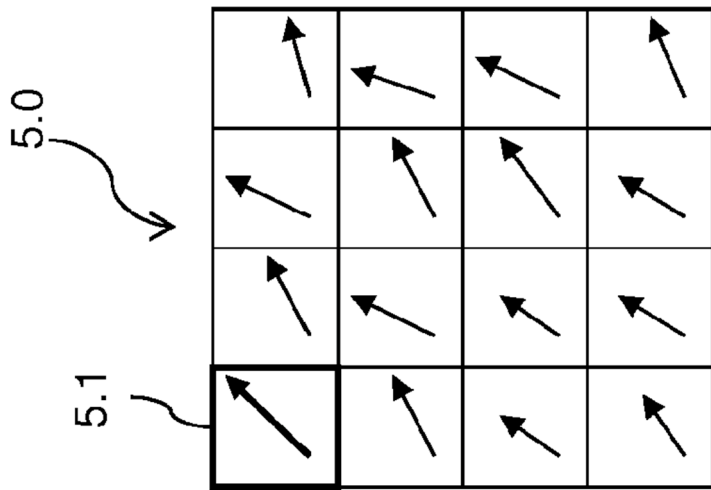


Fig. 5

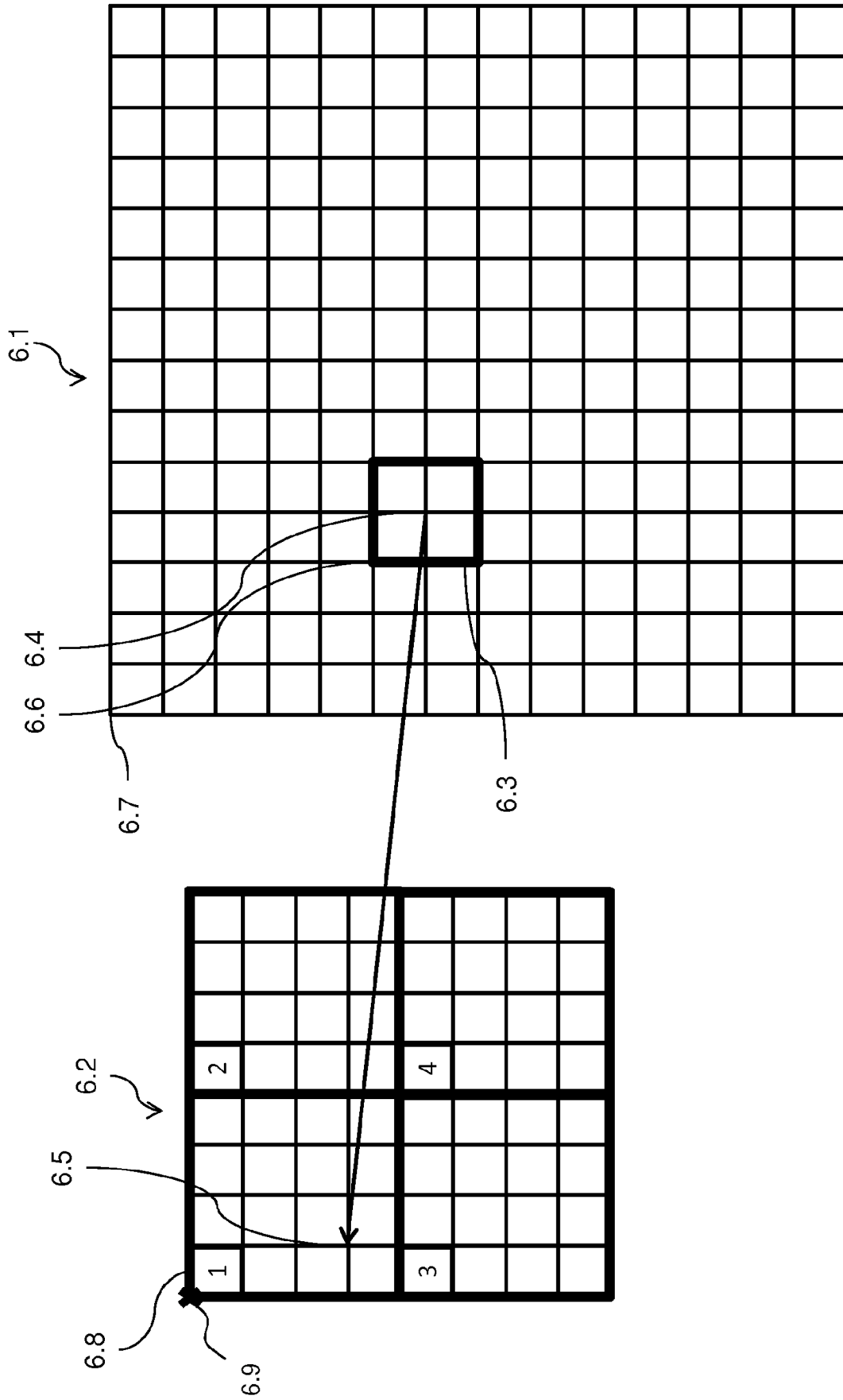


Fig. 6

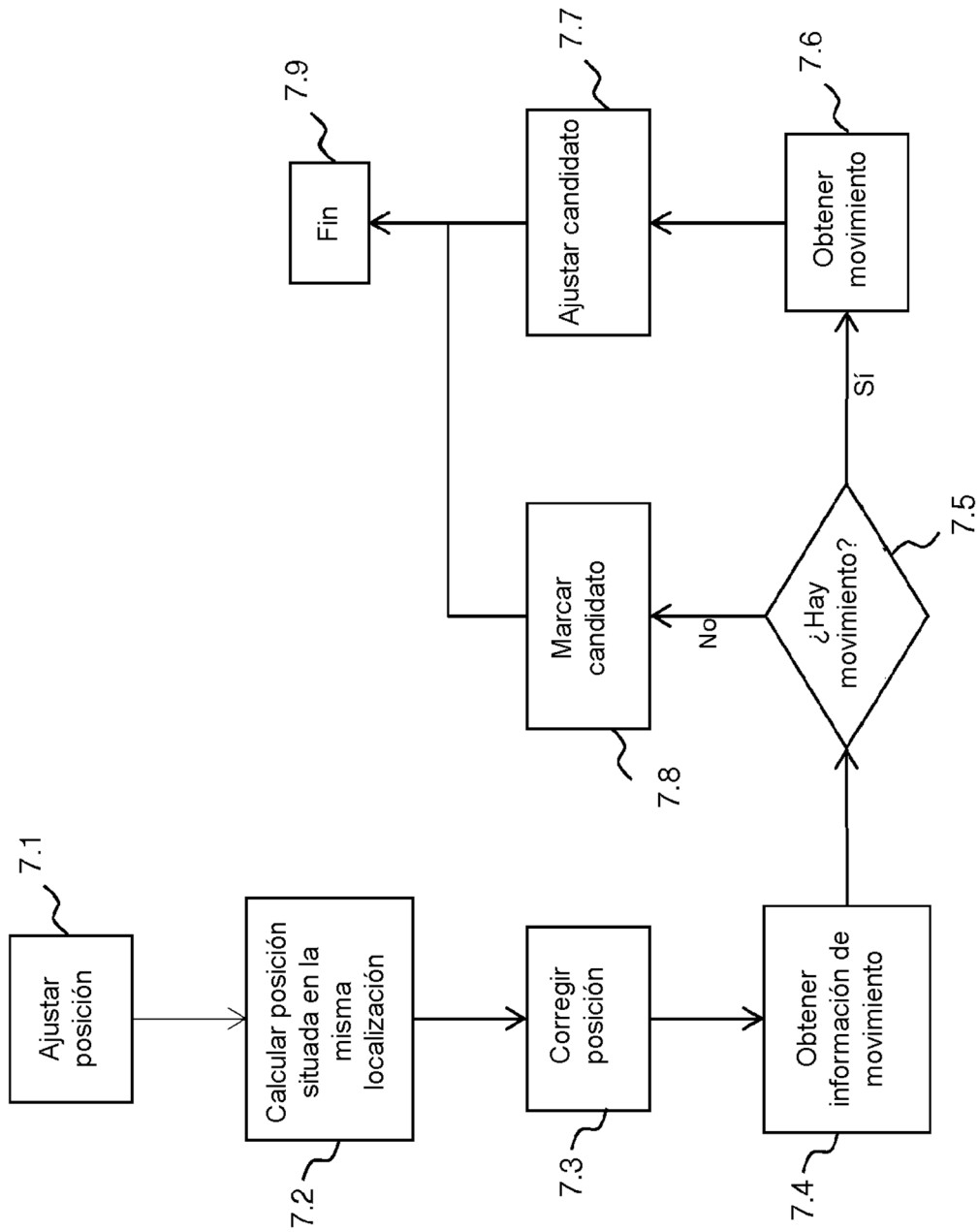


Fig. 7

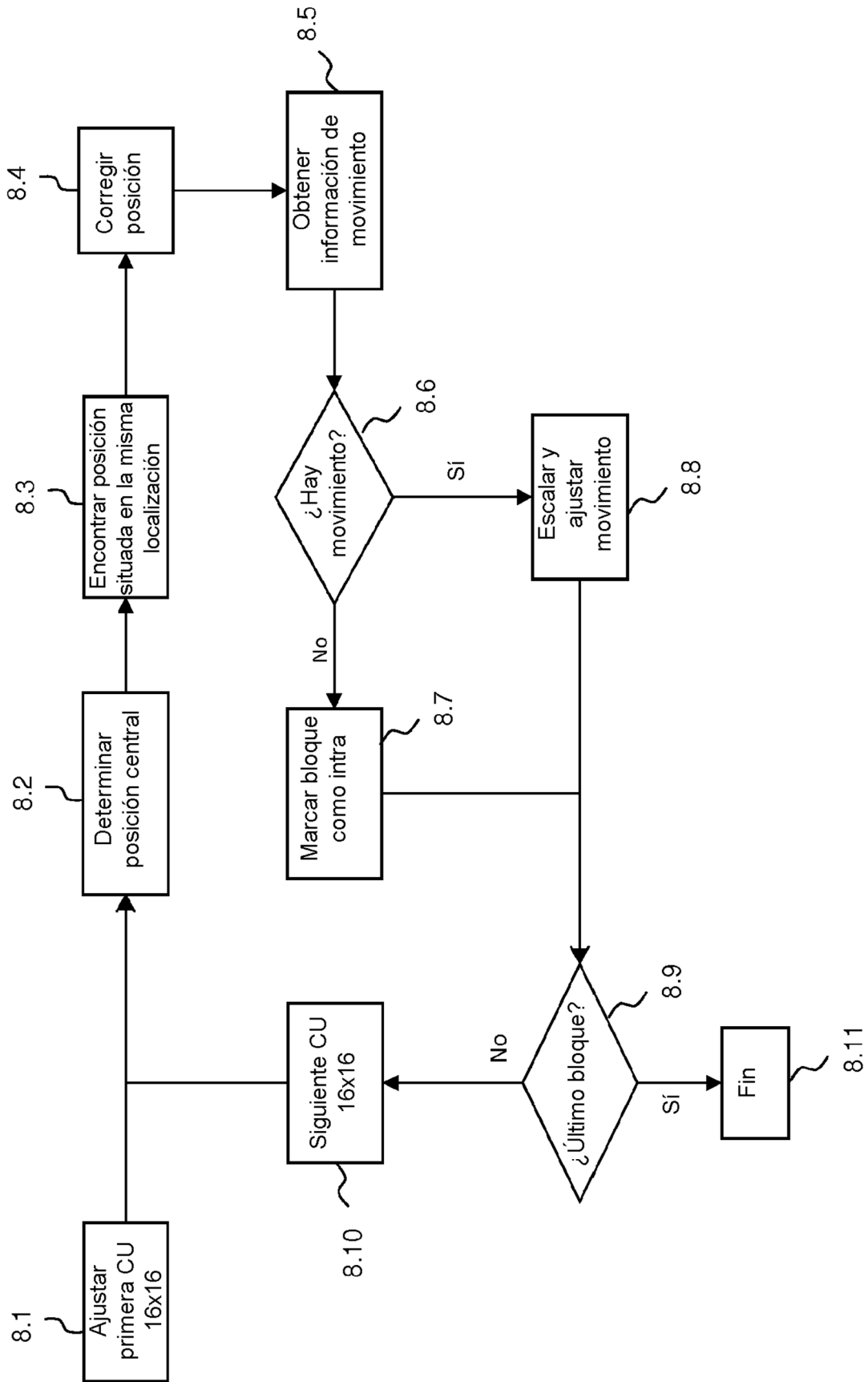


Fig. 8

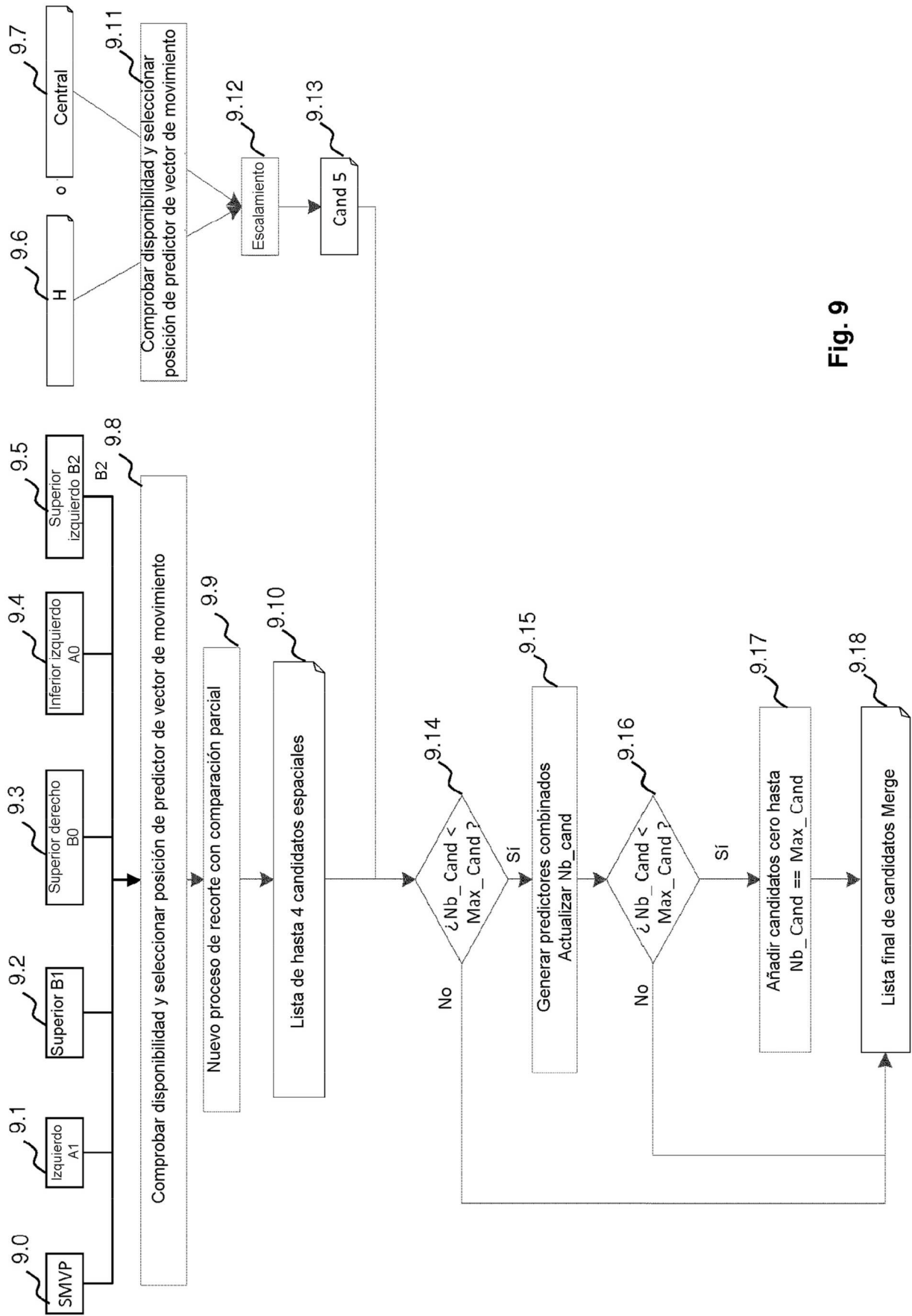


Fig. 9

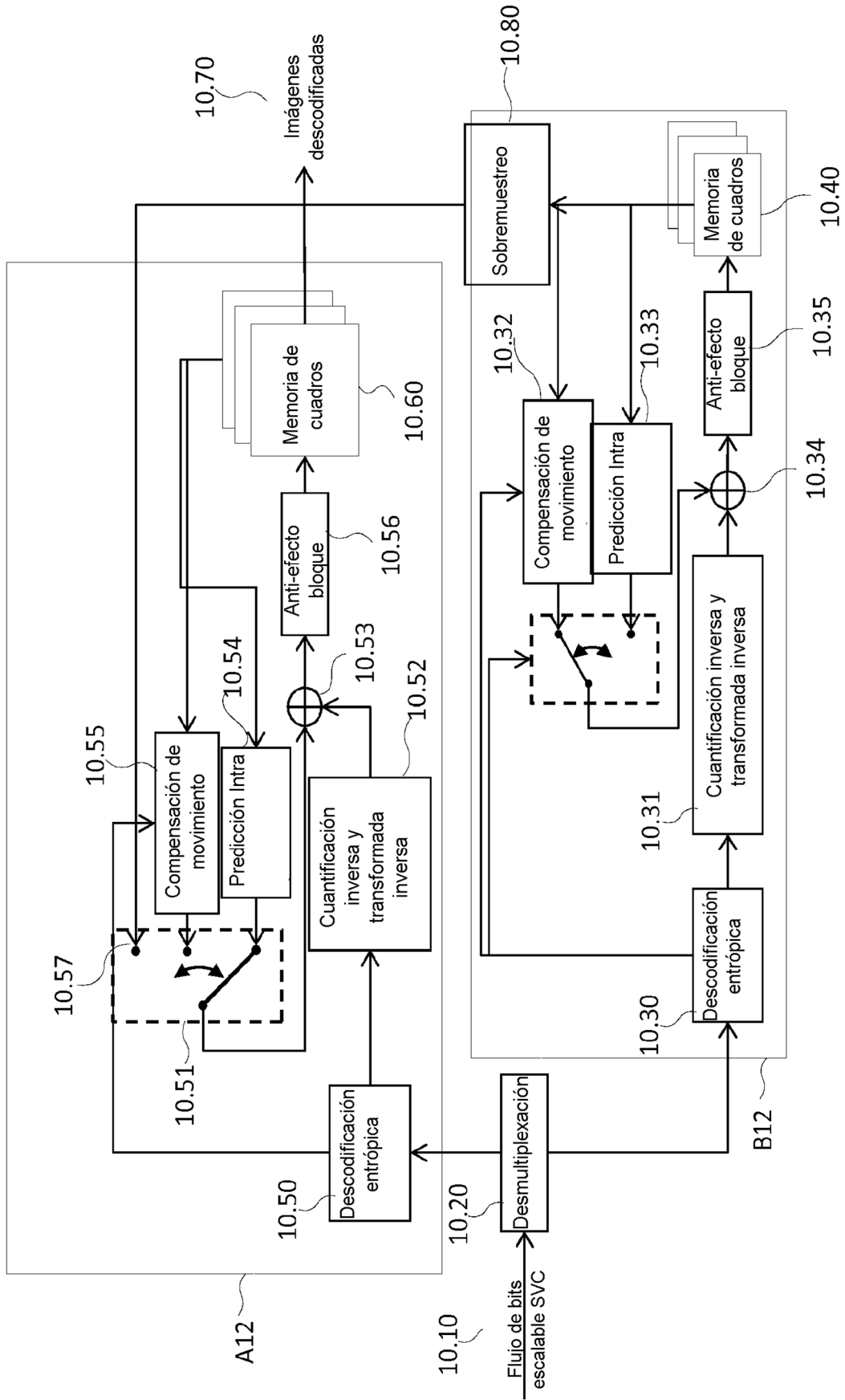


Fig. 10

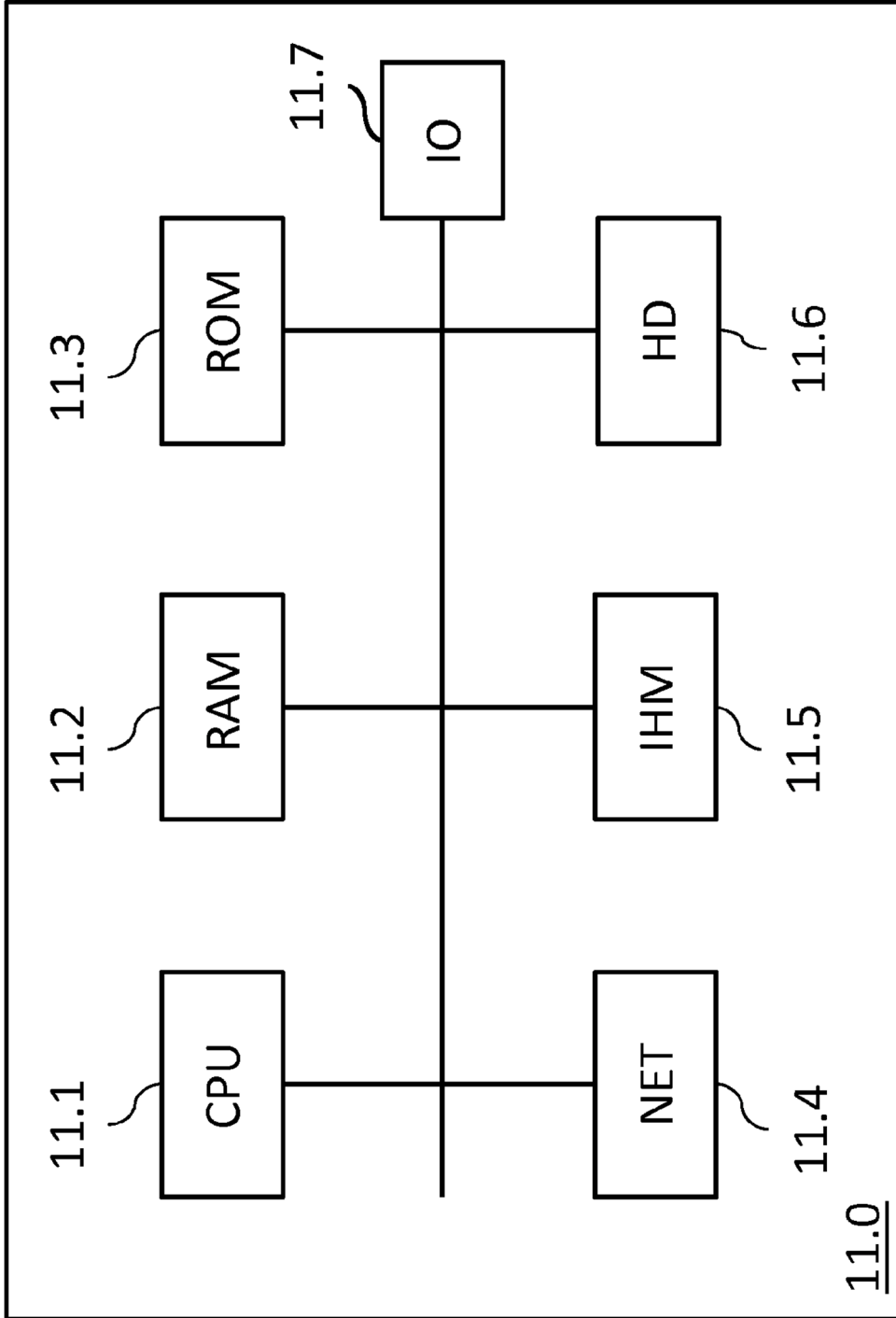


Fig. 11

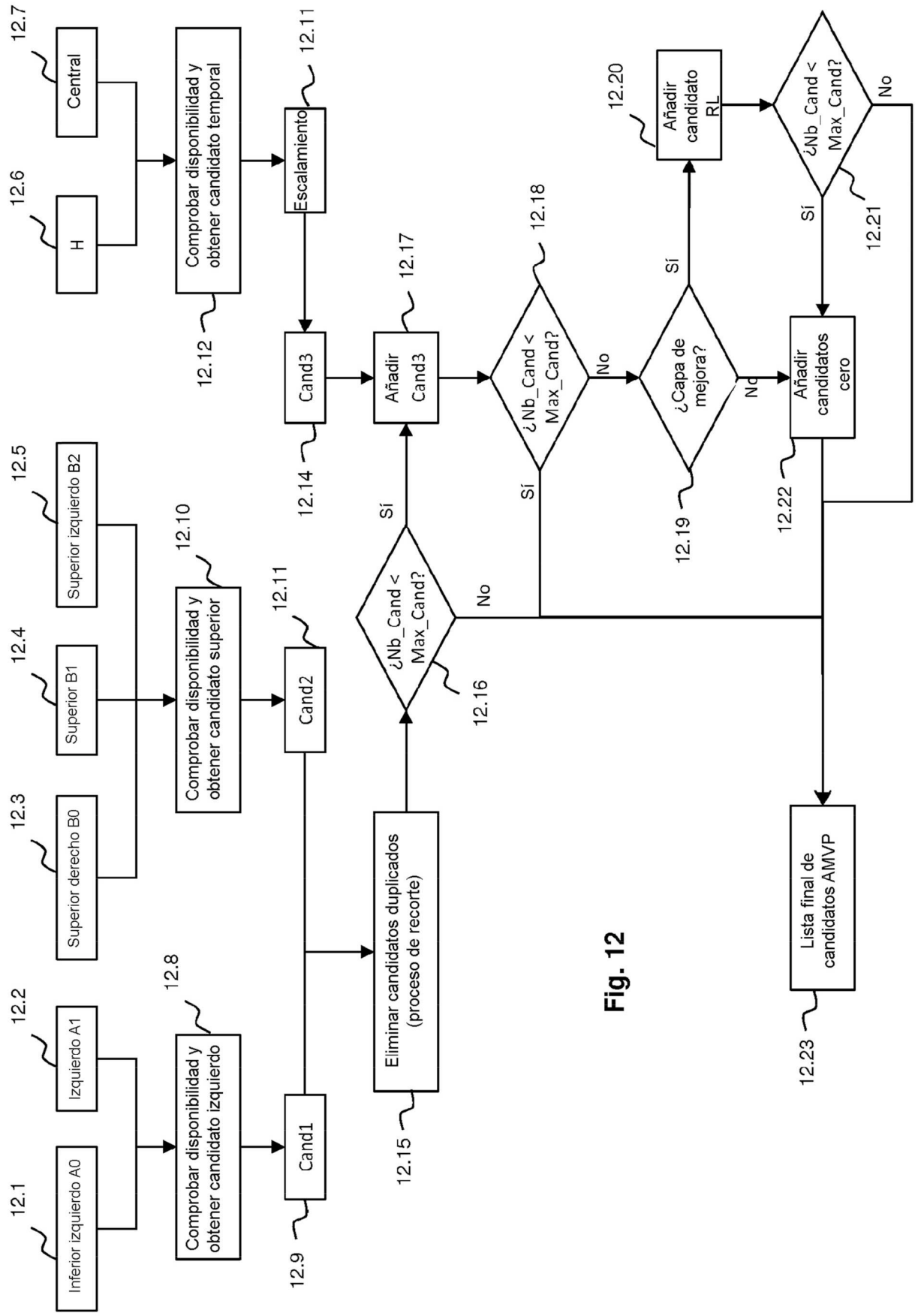


Fig. 12

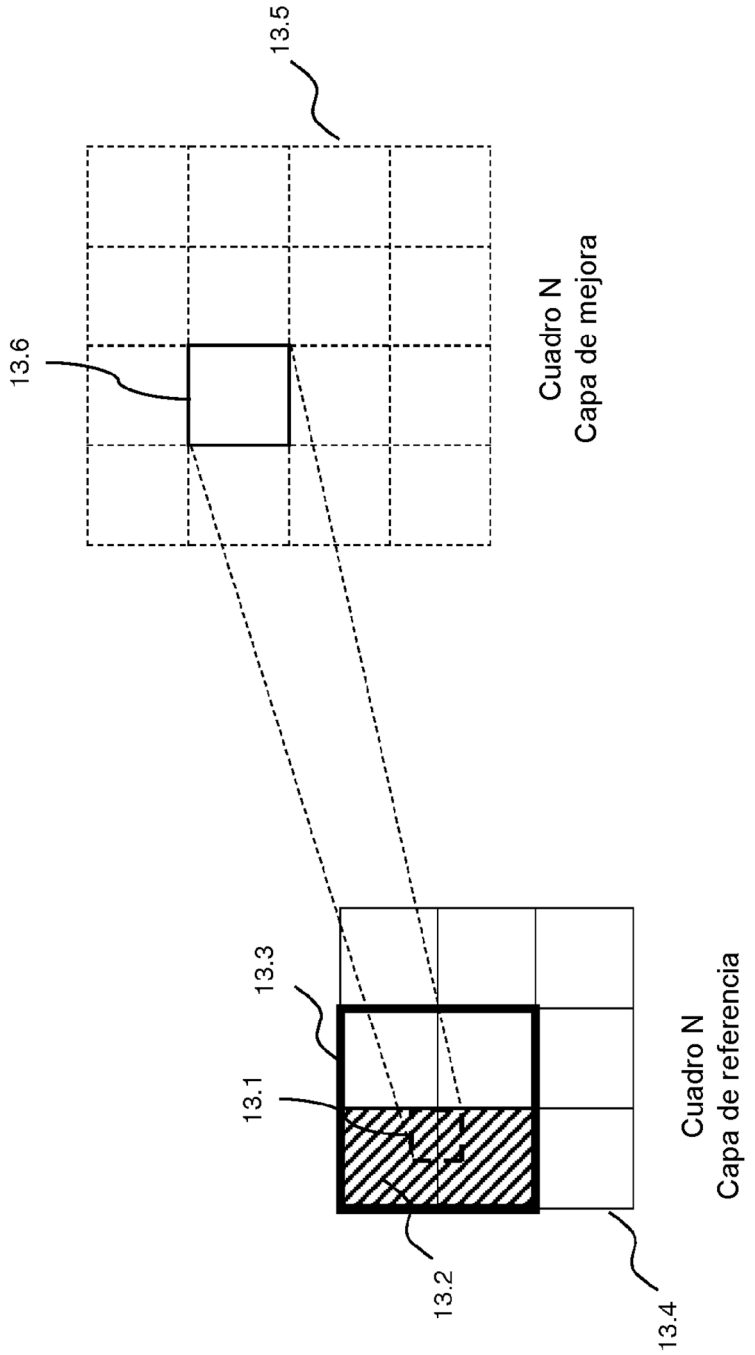


Fig. 13

REFERENCIAS CITADAS EN LA DESCRIPCIÓN

5 *Esta lista de referencias citada por el solicitante es únicamente para mayor comodidad del lector. No forman parte del documento de la Patente Europea. Incluso teniendo en cuenta que la compilación de las referencias se ha efectuado con gran cuidado, los errores u omisiones no pueden descartarse; la EPO se exime de toda responsabilidad al respecto.*

Literatura no patente citada en la descripción

- 10 • **LASSERRE S et al.** Description of the scalable video coding technology proposal by Canon Research Centre France. 11. *JCT-VC Meeting*; 102. *MPEG Meeting; Shanghai; JCTVC-K0041*
- **PARK S et al.** Modifications of temporal mv compression and temporal mv predictor. 96. *MPEG Meeting; Geneva; m19567*