

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 780 849**

51 Int. Cl.:

H04N 19/105 (2014.01)

H04N 19/503 (2014.01)

H04N 19/70 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **10.01.2013 E 18172546 (6)**

97 Fecha y número de publicación de la concesión europea: **27.11.2019 EP 3383042**

54 Título: **Método y aparato para señalización y construcción de listas de imágenes de referencia de codificación de video**

30 Prioridad:

19.01.2012 US 201261588571 P

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

27.08.2020

73 Titular/es:

**VID SCALE, INC. (100.0%)
200 Bellevue Parkway Suite 300
Wilmington, Delaware 19809, US**

72 Inventor/es:

**YE, YAN y
HE, YONG**

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 780 849 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 señalización y construcción de listas de imágenes de referencia de codificación de video

5 ANTECEDENTES

Los sistemas de codificación de video se utilizan ampliamente para comprimir señales de video digital para reducir las necesidades de almacenaje y/o el ancho de banda de transmisión de tales señales. Entre los diversos tipos de sistemas de codificación de video, tal como los sistemas basados en bloques, basados en ondícula, y basados en el objeto, son en la actualidad los sistemas de codificación de video híbridos basados en bloques los más ampliamente usados e implantados. Ejemplos de sistemas de codificación de video basados en bloques incluyen estándares internacionales de codificación de video tal como los estándares MPEG1/2/4 parte 2, H.264/MPEG-4 parte 10 AVC [1][3] y VC-1[2].

La Figura 1 es un diagrama de bloques de un sistema genérico de codificación de video híbrido basado en bloques. La señal 102 de video de entrada se procesa bloque por bloque. En todos los estándares de codificación de video existentes, la unidad de bloque de video consiste en 16x16 píxeles; tal unidad de bloque se conoce también habitualmente como macrobloque o MB. Actualmente, JCT-VC (Equipo de Colaboración Conjunta sobre Codificación de Video) de ITU-T/SG 16/Q.6/VCEG e ISO/IEC/MPEG están desarrollando el estándar de codificación de video de la próxima generación denominado Codificación de Video de Alta Eficiencia o HEVC [4]. En HEVC, los tamaños de los bloques ampliados (denominados "unidad de codificación" o CU) se usan para comprimir eficientemente señales de video de alta resolución (1080p y más). En HEVC, una CU puede ser de hasta 64x64 píxeles. Una CU puede ser además dividida en unidades de predicción o PUs, para las que se aplican métodos de predicción separados. Para cada bloque de video de entrada (MB o CU), se puede realizar predicción espacial (160) y/o predicción temporal (162). La predicción espacial (o "intra predicción") utiliza píxeles de los bloques contiguos ya codificados en la misma imagen/porción de video para predecir el bloque de video actual. La predicción espacial reduce la redundancia espacial inherente en la señal de video. La predicción temporal (también denominada "inter predicción" o "predicción compensada de movimiento") usa píxeles procedentes de imágenes de video ya codificadas (mencionadas usualmente como "imágenes de referencia") para predecir el bloque de video actual. La predicción temporal reduce la redundancia temporal inherente en la señal de video. La señal de predicción temporal para un bloque de video dado se señala normalmente mediante uno o más vectores de movimiento que indican la cantidad y la dirección de movimiento entre el bloque actual y su bloque de predicción en la imagen de referencia. También, si están soportadas múltiples imágenes de referencia (como en el caso de los recientes estándares de codificación de video tales como H.264/AVC o HEVC), entonces, para cada bloque de video, se envía adicionalmente su índice de imagen de referencia. El índice de imagen de referencia identifica qué imagen de referencia debe ser obtenida en el almacén (164) de imágenes de referencia (también conocido como "memoria intermedia de imagen descodificada" o DPB), se debe obtener la señal de predicción temporal con el fin de generar la predicción del bloque de video actual que debe ser reconstruido. Tras predicción espacial y/o temporal, el bloque (180) de decisión de modo en el codificador elige el mejor modo de predicción, por ejemplo, en base al método de optimización de velocidad-distorsión. El bloque de predicción se resta a continuación del bloque (116) de video actual; y, el residuo de predicción se transforma (104) y se cuantifica (106). Los coeficientes residuales cuantificados son cuantificados inversamente (110) y transformados inversamente (112) para formar el residuo reconstruido, el cual se añade después de nuevo al bloque (126) de predicción para formar el bloque de video reconstruido. Se puede aplicar (166) además filtraje en bucle tal como desbloqueando filtros, Desviación Adaptativa de Muestra, y Filtros de Bucle Adaptativo, sobre el bloque de video reconstruido con anterioridad a disponerlo en el almacén (164) de imágenes de referencia, y usarlo para codificar bloques de video futuros. Para formar la corriente de bits 120 de video de salida, el modo de codificación (intra o inter), la información de modo de predicción, la información de movimiento, y los coeficientes residuales cuantificados, se envían todos ellos a la unidad (108) de codificación de entropía para ser comprimidos y empaquetados adicionalmente para formar la corriente de bits.

La Figura 2 proporciona un diagrama general de bloques de un descodificador de video basado en bloques. La corriente de bits 202 de video se desempaqueta en primer lugar y se descodifica en cuanto a entropía en la unidad 208 de descodificación de entropía. El modo de codificación y la información de predicción se envían ya sea a la unidad 260 de predicción espacial (si están intra codificadas) o ya sea a la unidad 262 de predicción temporal (si están inter codificadas) para formar el bloque de predicción. Si está inter codificada, la información de predicción incluye tamaños de bloques de predicción, uno o más vectores de movimiento (que indican la dirección y la cantidad de movimiento) y uno o más índices de referencia (que indican a partir de qué imagen de referencia ha de obtenerse la señal de predicción). A continuación se aplica predicción compensada de movimiento mediante la unidad 262 de predicción temporal, para formar el bloque de predicción temporal. Los coeficientes de transformación residual se envían a la unidad 210 de cuantificación inversa y a la unidad 212 de transformación inversa para reconstruir el bloque residual. El bloque de predicción y el bloque residual se suman a continuación entre sí en 226. El bloque reconstruido puede ir más allá mediante filtrado en bucle con anterioridad a ser almacenado en el almacén 264 de imágenes de referencia. El video reconstruido en el almacén de imágenes de referencia se envía a continuación para activar un dispositivo de visualización, y también se usa para predecir los futuros bloques de video.

COMPENDIO

La invención se define en el juego de reivindicaciones anexas; los ejemplos adicionales denominados realizaciones en la descripción, son ejemplos ilustrativos, no realizaciones reivindicadas en la presente solicitud.
 5 Para el propósito mencionado con anterioridad descrito en la presente memoria son los métodos y sistemas que proporcionan flexibilidad para mejorar la señalización de imágenes de referencia usadas para predicción temporal (véase el bloque 162 en la Figura 1 y el bloque 262 en la Figura 2). En particular, se mejoran los esquemas de señalización y el proceso de construcción para diferentes listas de imágenes de referencia en HEVC Working Draft 5 (WD5) [4][5].

10 Conforme a una realización, un método de generación de listas L0 y L1 de imágenes de referencia para descodificación de una imagen pronosticada dentro de los datos de video comprende: generar una primera lista ordenada de imágenes de referencia a partir de una memoria intermedia de imágenes descodificadas (DPB) en donde la lista está ordenada con las imágenes de referencia en la DBP que son temporalmente anteriores a la
 15 imagen actual, si las hay, relacionadas por orden mediante distancia temporal desde la imagen actual, seguido de las imágenes de referencia de la DPB que son temporalmente posteriores a la imagen actual, si las hay, relacionadas por orden mediante la distancia temporal a partir de la imagen actual, seguido de imágenes de referencia de largo plazo en la DBP, si las hay, en el orden en que éstas han sido almacenadas en la DPB; generar una segunda lista ordenada de imágenes de referencia a partir de una memoria intermedia de imágenes descodificadas (DPB) en donde la lista está ordenada con las imágenes de referencia de la DPB que son temporalmente posteriores a la imagen actual, si las hay, relacionadas primero en orden a la distancia temporal desde la imagen actual, seguido de las imágenes de referencia en la DPB que son temporalmente anteriores a la
 20 imagen actual, si las hay, relacionadas por orden mediante distancia temporal desde la imagen actual, seguido de imágenes de referencia de largo plazo de la DPB, si las hay, en el orden en que éstas han sido almacenadas en la DPB; generar al menos una de las listas L0 y L1 seleccionando imágenes de referencia a partir de la primera lista ordenada y de la segunda lista ordenada, respectivamente.

Según otra realización, un método para inicializar un descodificador para listas de imágenes de referencia para descodificar una cabecera de la porción P o B comprende: construir una primera lista temporal, RefPicSetCurrTempList0, mediante:

```

    cldx = 0
    NumRpsCurrTempList = NumRpStCurr0 + NumRpsStCurr1 + NumRpsLtCurr
    para {i = 0; i < NumRpsStCurr0; cldx++, i++}
    35     RefPicSetCurrTempList0 [cldx] = RefPicSetStCurr0 [i]
    para (i = 0; i < NumRpsStCurr1; cldx++, i++)
        RefPicSetCurrTempList0 [cldx] = RefPicSetStCurr1 [i]
    para (i = 0; i < NumRpsLtCurr; cldx++, i++)
        RefPicSetCurrTempList0 [cldx] = RefPicSetLtCurr [i]
    
```

40 Conforme a otra realización más, un método de señalización de modificaciones para una pluralidad de listas de imágenes de referencia comprende señalar las modificaciones a la pluralidad de listas de imágenes de referencia usando una sintaxis de señalización unificada.

45 Conforme a una realización adicional, un método comprende: determinar un número de entradas en una lista de imágenes de referencia; generar un mensaje que incluya un valor identificativo de una entrada en la lista de imágenes de referencia, en donde el valor está representado por un solo bit si el número de entradas en la lista de imágenes de referencia es dos, el valor está representado por múltiples bits si el número de entradas en la lista de imágenes de referencia es tres o más, y el mensaje omite el valor si el número de entradas en la lista de imágenes de referencia es uno.

Conforme a otra realización más, un método de creación de una lista combinada, LC, de imágenes de referencia para ser usada para descodificación de una imagen P o B a partir de una primera lista de imágenes de referencia, L0, y una segunda lista de imágenes de referencia, L1, comprende: determinar si L0 contiene más de una
 55 entrada; determinar si L1 contiene más de una entrada; si alguna de L0 o L1 contiene más de una entrada, usar un elemento de sintaxis ref_idx_list_curr para indicar entradas en al menos una de L0 y L1 para ser añadidas a LC; si L0 contiene solamente una entrada, establecer ref_idx_list_curr en 0; si L1 contiene solamente una entrada, establecer ref_idx_list_curr_ en 0; y crear LC usando el valor de ref_idx_list_curr.

60 BREVE DESCRIPCIÓN DE LOS DIBUJOS

Se puede lograr una comprensión más detallada a partir de la descripción que sigue, dada a título de ejemplo, junto con los dibujos que se acompañan, en donde:

65 La Figura 1 es un diagrama de bloques de un esquema de codificación de video híbrido basado en bloques, en el que se puede incorporar una realización de la invención;

La Figura 2 es un diagrama de bloques de un esquema de descodificación de video basado en bloques, en el que se puede incorporar una realización de la invención;

La Figura 3 representa uni-predicción temporal de un almacén de imágenes de referencia que contiene múltiples imágenes de referencia conforme a la técnica anterior;

5 La Figura 4 representa bi-predicción temporal usando múltiples imágenes de referencia conforme a la técnica anterior;

La Figura 5 es un diagrama de flujo de un proceso para construir una lista combinada de imágenes de referencia conforme a la técnica anterior;

10 La Figura 6 muestra un ejemplo de un proceso para construir una lista combinada de imágenes de referencia conforme al proceso descrito en relación con la Figura 5;

La Figura 7 muestra ejemplos de un proceso modificado para construir una lista combinada de imágenes de referencia conforme a la técnica anterior;

La Figura 8 es un diagrama de flujo de una modificación de lista de imágenes de referencia para L0 y L1 conforme a la técnica anterior, usando L0 como ejemplo;

15 La Figura 9 es un ejemplo de proceso de ref_pic_list_modification para L0 conforme al proceso descrito en relación con la Figura 8;

La Figura 10 es un diagrama de flujo de una modificación de lista de imágenes de referencia, usando L0 como ejemplo, conforme a una realización de la presente invención;

20 La Figura 11 es un proceso de ref_pic_list_modification para el mismo ejemplo de la Figura 9 conforme a los principios de una realización de la presente invención;

La Figura 12A es un diagrama de sistema de un ejemplo de sistema de comunicaciones en el que se pueden implementar una o más de las realizaciones descritas;

La Figura 12B es un diagrama de sistema de un ejemplo de unidad de transmisión/recepción inalámbrica (WTRU) que puede ser usada dentro del sistema de comunicaciones ilustrado en la Figura 12A, y

25 Las Figuras 12C, 12D y 12E son diagramas de sistema de ejemplos de redes de acceso de radio y de ejemplos de redes centrales que pueden ser usadas dentro del sistema de comunicaciones ilustrado en la Figura 12A.

DESCRIPCIÓN DETALLADA

30 Según se utilizan en la presente memoria, los términos “predicción temporal”, “predicción de movimiento”, “predicción compensada de movimiento” e “inter predicción” se utilizan intercambiamente; y, los términos “almacén de imágenes de referencia”, “memoria intermedia de imágenes descodificadas” y “DPB” se usan intercambiamente.

35 Conforme a técnicas conocidas adoptadas en el H.264 y en HEVC WD 5, la predicción temporal de bloques de video puede realizarse usando una técnica de uni-predicción o una técnica de bi-predicción. Con el fin de llevar a cabo predicción conforme a tales técnicas, se señalizan y se construyen listas de imágenes de referencia. Para uni-predicción, puede existir una única lista de imágenes de referencia a partir de la cual se predicen los bloques en la imagen actual. Para bi-predicción, existen dos listas, L0 y L1, siendo seleccionada una imagen de referencia en cada lista para formar predicciones de los bloques en la imagen actual. Más aún, se han realizado propuestas (aunque ya no siguen incorporadas en el último HEVC WD9 [9] en el momento de este escrito) para técnicas de bi-predicción que incluyen el uso de una tercera lista que es una combinación de las dos primeras listas, L0 y L1, denominada en la presente memoria lista LC. En la presente memoria se describen métodos y sistemas para una técnica eficiente unificada para señalización de una sintaxis de modificación para todas las listas de imágenes de referencia L0, L1 y/o LC, así como una técnica para señalización de la lista combinada de imágenes de referencia, LC.

40 La Figura 3 ayuda a ilustrar esquemáticamente uni-predicción a partir de una única lista 301 de imágenes de referencia según puede ser llevada a cabo por un procesador de inter predicción (por ejemplo, el bloque 162 en la Figura 1). Según técnicas de uni-predicción, la lista 301 de imágenes de referencia contiene enlaces a bloques de video, por ejemplo, el bloque 304, desde tramas de video contiguas, ya codificadas, para predecir el bloque de video actual, y de ese modo poder aprovechar la correlación temporal y eliminar la redundancia temporal inherente en la señal de video. Estas tramas de video ya codificadas se almacenan en una memoria intermedia de imágenes descodificadas (DPB, por ejemplo, el almacén 164 de imágenes de referencia en la Figura 1).

55 H.264/AVC y HEVC WD5 permiten el uso de más de una imagen de referencia. En la Figura 3, se puede usar una lista de N imágenes de referencia 303, etiquetadas como ref^n , con $n=0, \dots, N-1$, para predecir bloques de video 307 en la imagen actual 305. Supóngase que se selecciona ref^n como base a partir de la cual se va a predecir el bloque 307 actual con un vector de movimiento (mvx, mvy). La predicción temporal se realiza como:

$$60 \quad P(x,y) = ref^n(x.mvx, y-mvy) \quad (1)$$

donde $ref^n(x,y)$ es el valor de pixel en la posición (x, y) en el ref^n de la imagen de referencia, y $P(x,y)$ es el bloque predicho. Los sistemas de codificación de video existentes soportan inter predicción con precisión de fracción de pixel [1][2][4]. Cuando un vector de movimiento (mvx, mvy) tiene valor de fracción de pixel, se aplican filtros de interpolación para obtener los valores de pixel en las posiciones fraccionales de pixel.

En la ecuación (1), la predicción temporal proviene de una fuente (es decir, ref^m), que se menciona normalmente como uni-predicción. Una imagen o una porción (un grupo de bloques de video) en la que todos los bloques de la imagen o de la porción se han predicho usando uni-predicción, se menciona normalmente como una imagen P o una porción P.

Para mejorar la precisión de la predicción temporal, los sistemas de codificación de video más recientes basados en bloques soportan también predicción multi-hipótesis, donde la señal de predicción se forma combinando una pluralidad de señales de predicción procedentes de imágenes de referencia diferentes. Una forma habitualmente utilizada de predicción multi-hipótesis se denomina bi-predicción, donde dos señales de predicción, cada una de ellas procedente de una imagen en una lista diferente de imágenes de referencia, se combinan para formar la predicción para el bloque actual. La Figura 4 ayuda a ilustrar la bi-predicción. En particular, dos listas de imágenes de referencia, la lista 0, 401, y la lista 1, 403, se usan para predecir bloques de video en la imagen actual. La lista 0 contiene un total de N_0 imágenes 404, mientras que la lista 1 contiene un total de N_1 imágenes 404. En la Figura 4, ref^{m_0} de la lista 0, 401, con vector de movimiento (mvx_0 , mvy_0) y ref^{m_1} de la lista 1, 403, con vector de movimiento (mvx_1 , mvy_1), se seleccionan para formar la bi-predicción del bloque 410 predicho de la imagen actual 412 como en la Ecuación (2):

$$P(x, y) = \frac{P_0(x, y) + P_1(x, y)}{2} = \frac{ref^{m_0}(x - mvx_0, y - mvy_0) + ref^{m_1}(x - mvx_1, y - mvy_1)}{2} \quad (2)$$

donde $P_0(x, y)$ y $P_1(x, y)$ son el primer y el segundo bloques de predicción 407 y 408, respectivamente. Una imagen o una porción se denomina normalmente como imagen B o porción B, si al menos algunos bloques de la imagen o de la porción se predicen usando bi-predicción (mientras que otros se pueden predecir usando uni-predicción). La bi-predicción está soportada en todos los estándares de codificación de video recientes, tales como MPEG2/4, VC1, H.264 y HEVC.

Tras la predicción, el bloque de predicción $P(x, y)$ se resta del bloque de video original en un primer sumando (véase 116 en la Figura 1) para formar un bloque residual de predicción. El bloque residual de predicción se transforma en la unidad 104 de transformación y se cuantifica en la unidad 106 de cuantificación. Los bloques de coeficientes de transformación residual cuantificados se envían a continuación a una unidad 108 de codificación de entropía para ser codificados en entropía para reducir más la tasa de bits. Los coeficientes residuales codificados en entropía son empaquetados a continuación para formar parte de una corriente de bits 120 de video de salida.

La estructura de la lista de imágenes de referencia para una imagen/porción P es relativamente simple dado que todos los bloques se predicen usando uni-predicción, lo que significa que solamente se necesita una lista de imágenes de referencia. En una imagen/porción B, sin embargo, algunos bloques pueden ser predichos usando bi-predicción, mientras que otros se predicen usando uni-predicción. En HEVC, las listas de imágenes de referencia para la bi-predicción, en especial la lista 0 (o L0) 401 y la lista 1 (o L1) 403, según se aprecia en la Figura 4, son las mismas que en H.264/AVC. Sin embargo, HEVC difiere de H.264/AVC en cómo se forma la lista de imágenes de referencia para uni-predicción para imágenes/porciones B. En H.264/AVC, la uni-predicción para un bloque de video en una imagen/porción B necesita indicar en primer lugar si la predicción proviene de L0 o de L1, seguido por el ref_idx para esa lista particular. En HEVC, en el 4º encuentro de JCT-VC, se presentó el concepto de lista combinada de imágenes de referencia [8]. La lista combinada, mencionada como LC en la presente descripción, se forma combinando L0 y L1 entre sí; LC sirve entonces como la única lista de imágenes de referencia para todos los bloques predichos usando uni-predicción en una imagen/porción B.

Normalmente en HEVC, por defecto, la lista combinada, LC, se forma tomando imágenes únicas desde L0 y L1 de una manera alternativa para asegurar una redundancia mínima en la lista combinada. Un diagrama de flujo de la generación de lista combinada por defecto se proporciona en la Figura 5. En particular, los índices i , j y k , en las listas L0, L1 y LC, respectivamente, se inicializan en 501, puesto que son los tamaños de las dos listas L0 y L1. En la decisión 503, se determina si todas las imágenes de referencia en L0 han sido comprobadas. Si no lo han sido, el flujo avanza hasta el bloque de decisión 505, en donde se determina si la imagen de referencia con el índice i en L0 está ya en la lista combinada LC. Si no lo está, se añade y el índice en la lista combinada LC se incrementa (507). También se incrementa el índice i (509). Si, por otra parte, está ya en la lista LC, el flujo avanza en cambio directamente desde 505 a 509. A continuación, se lleva a cabo un proceso esencialmente idéntico en relación con la imagen de referencia de índice j en la lista L1. Específicamente, en la decisión 511, se determina si todas las imágenes de referencia en L1 han sido comprobadas. Si no lo han sido, el flujo avanza hasta el bloque de decisión 513, en donde se determina si la imagen de referencia con índice j en L1 está ya en la lista combinada LC. Si no lo está, ésta se añade y el índice en LC se incrementa (515). El índice j en L1 también se incrementa (517). Si, por otra parte, la imagen de referencia con índice j en L1 está ya en LC, el flujo avanza en cambio directamente desde 513 a 517. Según se aprecia en el bloque de decisión 519, el proceso se repite comprobando alternativamente la siguiente imagen de referencia en cada una de las listas L0 y L1 hasta que se

alcanza el final de las dos listas.

Un ejemplo de lista combinada LC creada por el proceso ilustrado mediante el diagrama de flujo de la Figura 5, ha sido proporcionado en la Figura 6. En este ejemplo, la imagen actual que está siendo codificada está temporalmente entre las imágenes de referencia 2 y 4. Adicionalmente, L0 contiene las imágenes de referencia Ref 2, Ref 1 y Ref 4, por ese orden, y L1 contiene las imágenes de referencia Ref 4, Ref 5 y Ref 2, por ese orden. Siguiendo el flujo en la Figura 5, el ejemplo de la Figura 6 forma la lista combinada LC examinando, de una forma alternativa, si cada una de las tres imágenes de referencia en L0 y en L1, respectivamente, existen ya en la LC, y añadiendo todas las imágenes de referencia que no existan previamente en LC. Como resultado, el ejemplo de la Figura 6 forma una lista combinada LC con cuatro imágenes de referencia, añadiendo a LC, por orden, la primera imagen de referencia de L0 (Ref 2), la primera imagen de referencia de L1 (Ref 4), la segunda imagen de referencia de L0 (Ref 1), la segunda imagen de referencia de L1 (Ref 5), cortando la tercera imagen de referencia en L0 (Ref 4) debido a que ésta ya ha sido añadida a LC en virtud de ser la misma imagen que la primera imagen de referencia en L1, y cortando la tercera imagen de referencia en L1 (Ref 2) debido a que ya ha sido añadida a LC en virtud de ser la misma imagen que la primera imagen de referencia en L0.

Obsérvese que el orden de codificación de las imágenes de referencia en cada una de las listas L0, L1 y LC en la Figura 6, es diferente del orden de visualización, dado que las imágenes de referencia Ref 4 y Ref 5 (últimas en el orden de visualización) han sido codificadas antes que la imagen actual. Este proceso por defecto para la construcción de LC yendo y viniendo entre las listas L0 y L1, asegura que cada entrada en LC representa una imagen única en la secuencia de video codificada, asegurando con ello una redundancia mínima.

Puesto que el proceso por defecto no soporta reordenación de imágenes de referencia (es decir, tener un tamaño de lista diferente del tamaño de la lista por defecto, estando las entradas en la lista ordenadas de forma diferente a las del proceso por defecto, repitiendo algunas entradas en la lista, y/o eliminando algunas entradas de la lista, y así sucesivamente), se usan elementos de sintaxis adicionales en HEVC WD5 (véase la Tabla 1 que sigue) para soportar el proceso de modificación de la lista combinada LC. La Figura 7 proporciona dos ejemplos de modificación de lista combinada, donde el primer ejemplo muestra una LC reordenada, y el segundo ejemplo muestra una LC con entradas repetidas y un tamaño de LC modificado (3 entradas) diferente del tamaño de la LC por defecto (4 entradas). En HEVC WD5, la lista combinada LC de imágenes de referencia se señala usando la tabla de sintaxis de la Tabla 1.

Tabla 1. Sintaxis de combinación de lista de imágenes de referencia en WDS (4)

ref_pic_list_combnation() {	Descriptor
si(slice_type% 5 = = 1) {/b slice	
ref_pic_list_combination_flag	u(1)
si(ref_pic_list_combination_flag) {	
num_ref_idx_lc_active_minus1	ue(v)
ref_pic_list_modification_flag_lc	u(1)
si(ref_pic_list_modification_flag_lc)	
para (i=0; i<=num_ref_idx_ic_active_minus1; i++) {	
pic_from_list_0_flag	u(1)
ref_idx_list_curr	ue(v)
}	
}	
}	

La semántica de combinación de la lista de imágenes de referencia es la siguiente:

ref_pic_list_combination_flag igual a 1, indica que la lista 0 de imágenes de referencia y la lista 1 de imágenes de referencia están combinadas para constituir una lista combinada adicional de imágenes de referencia usada para los bloques u otras unidades de predicción que están siendo uni-predichas. Cuando esta banderola es igual a 0, esto indica que la lista 0 de imágenes de referencia y la lista 1 de imágenes de referencia son idénticas, y de ese modo la lista 0 de imágenes de referencia puede ser usada como lista combinada de imágenes de referencia. La lista combinada de imágenes de referencia se determina que está vacía al comienzo del bucle definido en la Tabla 1.

num_ref_idx_lc_active_minus1+1 especifica el número de imágenes de referencia seleccionadas a partir de la lista 0 de imágenes de referencia o de la lista 1 de imágenes de referencia en la lista combinada de imágenes de referencia;

ref_pic_list_modification_flag_lc igual a 1 especifica que el elemento de sintaxis pic_from_list_0_flag y ref_idxlist_curr están presentes para especificar el mapeo para las entradas de la lista combinada de imágenes de referencia para las entradas de la lista 0 de imágenes de referencia y de la lista 1 de imágenes de referencia;

ref_pic_list_modification_flag_l0 igual a 0 especifica que estos elementos de sintaxis no están presentes. La lista combinada de imágenes de referencia se inicializa según se especifica en la sub-cláusula 8.2.2.4 de HEVC WD 5. pic_from_list_0_flag indica la imagen de referencia actual añadida a la lista combinada de imágenes de referencia proveniente de la lista 0 de imágenes de referencia o de la lista 1 de imágenes de referencia. Cuando esta banderola es igual a 1, la imagen proviene de la lista 0 de imágenes de referencia, y la CurrRefPicList es la lista 0 de imágenes de referencia; cuando esta banderola es igual a 0, la imagen proviene de la lista 1 de imágenes de referencia, y la CurrRefPicList es la lista 1 de imágenes de referencia; ref_idx_list_curr indica el índice de referencia de la imagen en la CurrRefPicList que va a ser añadida al final de la combinación de listas de imágenes de referencia.

Las listas L0 y L1 de imágenes de referencia pueden ser modificadas. Para permitir flexibilidad en el uso de las listas L0 y L1 de imágenes de referencia, tanto un proceso de construcción por defecto como un proceso de construcción modificado, están también soportados en HEVC. La construcción de la lista actual de imágenes de referencia y el proceso de modificación para L0 y L1 fueron presentados en el 7º encuentro de JCT-VT en Noviembre de 2011 [6][7] y adoptados en HEVC WD5 [4]. La sintaxis para la modificación de la lista de imágenes de referencia para la lista L0 y la lista L1 en HEVC WD5 se proporcionan en la Tabla 2 que sigue y se han representado en el diagrama de flujo de la Figura 8.

Tabla 2. Sintaxis de modificación de lista de imágenes de referencia para la Lista 0 y la Lista 1

	Descriptor
ref_pic_list_combination() {	
si(slice_type != 2) { // P slice o B slice	
ref_pic_list_modification_flag_l0	u(1)
si(ref_pic_list_modification_flag_l0)	
hacer {	
list_modification_idc	ue(v)
si(list_modification_idc != 3)	
ref_pic_set_idx	ue(v)
} mientras (list_modification_idc != 3)	
}	
si(slice_type == 1) { // B slice	
ref_pic_list_modification_flag_l1	u(1)
si(ref_pic_list_modification_flag_l1)	
hacer {	
list_modification_idc	ue(v)
si(list_modification_idc != 3)	
ref_pic_set_idx	ue(v)
} mientras (list_modification_idc != 3)	
}	
}	
}	

La semántica de modificación de lista de imágenes de referencia es como sigue:

Los elementos de sintaxis list_modification_idc y ref_pic_set_idx especifican el cambio desde las listas iniciales de imágenes de referencia a las listas de imágenes de referencia que van a ser usadas para la decodificación de la porción.

ref_pic_list_modification_flag_l0 igual a 1 especifica que el elemento de sintaxis list_modification_idc está presente para especificar la lista 0 de imágenes de referencia, mientras que ref_pic_list_modification_flag_l0 igual a 0 especifica que este elemento de sintaxis no está presente. Cuando ref_pic_list_modification_flag_l0 es igual a 1, el número de veces que list_modification_idc no es igual a 3 a continuación de ref_pic_list_modification_flag_l0 no podrá exceder de num_ref_idx_l0_active_minus1 + 1.

ref_pic_list_modification_flag_l1 igual a 1 especifica que el elemento de sintaxis list_modification_idc está presente para especificar la lista 1 de imágenes de referencia, mientras que si ref_pic_list_modification_flag_l1 es igual a 0 indica que este elemento de sintaxis no está presente. Cuando ref_pic_list_modification_flag_l1 es igual a 1, el número de veces que list_modification_idc no es igual a 3 a continuación de ref_pic_list_modification_flag_l1 no podrá exceder de num_ref_idx_l1_active_minus1 + 1.

list_modification_idc junto con ref_pic_set_idx especifica cuáles de las imágenes de referencia se re-mapean. Los valores de list_modification_idc se especifican en la Tabla 3. El valor de la primera list_modification_idc que sigue inmediatamente después de ref_pic_list_modification_flag_l0 o de ref_pic_list_modification_flag_l1 no podrá ser igual a 3.

Tabla 3. Operaciones de list_modification_idc para la modificación de listas de imágenes de referencia

list_modification_idc	modificación especificada
0	Para la lista 0: ref_pic_set_idx está presente y corresponde a un índice para RefPicSetStCurr0; para la lista 1: ref_pic_set_idx está presente y corresponde a un índice para RefPicSetStCurr1
1	Para lista 0: ref_pic_set_idx está presente y corresponde a RefPicSetStCurr1; Para la lista 1: ref_pic_set_idx está presente y corresponde a un índice para RefPicSetStCurr0
2	ref_pic_set_idx está presente y corresponde a un índice para RefPicSetL1Curr
3	Fin del bucle para modificación de la lista inicial de imágenes de referencia

5 ref_pic_set_idx especifica el índice, para RefPicSetStCurr0, RefPicSetStCurr1 o RefPicSetL1Curr de la imagen de referencia, referenciada por el índice actual en la lista de imágenes de referencia. El valor de ref_pic_set_idx deberá estar en la gama desde 0 a max_num_ref_frames, inclusive.

10 La Figura 8 muestra el diagrama de flujo de modificación de lista de imágenes de referencia para L0 y L1, usando L0 como ejemplo. El proceso de modificación detallado para L0 y L1, incluyendo las definiciones de los conjuntos de imágenes de referencia (RefPicSetStCurr0, RefPicSetStCurr1 y RefPicSetLtCurr), pueden ser encontrados en la HEVC WD5 [4] y en la parte de borrador de trabajo de [6][7]. Lo que sigue define los conjuntos de imágenes de referencia de la Figura 8 en términos simples:

15 RefPicSetStCurr0: imágenes de referencia de corto plazo con orden de visualización más pronta, es decir con anterioridad a la imagen actual (por ejemplo, Ref 1 y Ref 2 en la Figura 6)
 RefPicSetSrCurr1: imágenes de referencia de corto plazo con orden de visualización más tardía, es decir, después de la imagen actual (por ejemplo, Ref 4 y Ref 5 en la Figura 6)
 20 RefPicSetLtCurr: imágenes de referencia de largo plazo (no representadas en la Figura 6)

En 801, el índice en la lista L0 se inicia en cero. En 803, se lee ref_modification_idc. Ref_modification_idc puede tener cuatro valores, 0, 1, 2 y 3. Un valor de 3 indica que no se han de realizar más modificaciones, y que el proceso de modificación puede acabar. (Las modificaciones deseadas señalizadas por ref_modification_idc que tienen un valor de 0, 1 o 2, se explican a continuación en relación con las etapas 811, 813 y 815). De ese modo, en la etapa de decisión 805, si ref_modification_idc se establece en 3, no se lee más sintaxis. Si es cualquier otro valor, entonces, en 807 se lee ref_pic_set_idx. Éste es un índice en uno de los tres conjuntos de imágenes en la DPB (es decir, lo “anterior” a la imagen actual que es el conjunto descodificado de imágenes, el conjunto “posterior” de imágenes, o el conjunto de imágenes de largo plazo). (La selección de uno particular de los tres conjuntos ocurrirá en las etapas 811, 813 y 815, según se explica mejor en lo que sigue). En la etapa de decisión 809, se determina si la ref_modification_idc es 0, 1 o 2. Si es 0, entonces, en 811, la entrada en la lista L0 en el índice actual en la lista L0, RefIdxL0, se establece en la imagen de referencia de corto plazo con orden de visualización más pronta (es decir, RefPicSetStCurr0) que reside en la posición ref_pic_set_idx en el conjunto de imágenes de referencia más prontas de corto plazo en la DPB. Si en cambio es 1, entonces, en 813, la entrada en la lista L0 en el índice actual en la lista L0, RefIdxL0, se establece en la imagen de referencia de corto plazo con orden de visualización más tardía que la imagen actual que se está codificando (es decir, RefPicSetStCurr1) que está residiendo en la posición ref_pic_set_idx en el conjunto de imágenes de referencia más tardías de corto plazo en la DPB. Finalmente, si es 2, entonces, en 815, la entrada en la lista L0 en el índice actual en la lista L0, RefIdxL0, se establece para la imagen de referencia de largo plazo (es decir, RefPicSetLtCurr) que está residiendo en la posición ref_pic_set_idx en el conjunto de imágenes de referencia de largo plazo en la DPB.

40 En cada uno de los tres casos, el flujo avanza a continuación hasta 817, donde cualquiera de las entradas en la lista L0 tras la entrada recién modificada que se refiere a la misma imagen que la entrada recién modificada, se eliminan de L0. En 819, el índice en la lista L0 se incrementa y el flujo retorna a 803. El proceso continúa hasta que ref_modification_idc tiene un valor de 3, lo que indica que no se tienen que realizar más modificaciones adicionales.

De nuevo, usando L0 como ejemplo, la Figura 9 muestra los resultados del proceso de modificación de lista de imágenes de referencia perfilado por el diagrama de flujo de la Figura 8 para una DPB que contiene (1) imágenes de referencia Ref 2 y Ref 1 (en ese orden) en el conjunto de imágenes de referencia más inmediato de corto plazo, es decir, RefPicSetStCurr0 y (2) imágenes de referencia Ref 4 y Ref 5 (en ese orden) en el conjunto de imágenes de referencia más tardío de corto plazo, es decir, RefPicSetStCurr1. Por simplicidad, y sin pérdida de generalidad, el ejemplo de la Figura 9 no considera RefPicSetL1Curr, el cual se refiere al uso de imágenes de referencia de largo plazo, y solamente considera el uso de imágenes de referencia de corto plazo indicadas por RefPicSetStCurr0 y RefPicSetStCurr1.

55 Según se ha mostrado en la Figura 9, la lista L0 por defecto podría consistir en imágenes de referencia Ref 2, Ref

1, Ref 4 y Ref 5, en ese orden. En el ejemplo de la Figura 9, se desea una modificación simple de la última entrada en L0. El proceso de la Figura 8 requiere formación de bucle a través de las etapas 803 a 819, una vez por todas y cada una de las entradas en L0, incluyendo las tres primeras entradas, para las que no se requieren cambios y que señalizan `ref_modification_idc_` y `ref_pic_set_idx` para cada una, y a continuación señalizando además que el proceso ha terminado mediante señalización adicional distinta de `ref_modification_idc` de valor 3. De ese modo, se usan cinco etapas para llegar a la lista L0 objetivo modificada. En cada etapa, salvo en la última, se señalizan dos elementos de sintaxis (`list_modification_idc` y `ref_pic_set_idx`), y se mantiene y se incrementa una variable `RefIdx` adicional.

Además, comparando el proceso de modificación de lista de imágenes de referencia para LC (Tabla 1 anterior) y para L0/L1 (Tabla 2 y Tabla 3 anteriores), obsérvese que el proceso de modificación para LC en HEVC WD5 es diferente que para L0 y L1. En particular, el proceso de modificación para LC es más directo, puesto que cada entrada en la LC modificada está señalizada explícitamente, en vez de señalar dos elementos de sintaxis (`list_modification_idc` y `ref_pic_set_idx`) para cada entrada en la lista particular.

En la presente memoria se describen métodos para unificar estos procesos de modificación de lista y para proporcionar un proceso de modificación para L0 y L1 que requiera menos señalización y que es más directo.

En una realización, se proporciona un método para mejorar la eficacia del proceso de combinación de lista de imágenes de referencia. La Tabla 4 muestra un pseudo-código conforme a una realización de la invención para formar una lista combinada de imágenes de referencia. Los cambios a partir de la Tabla 1 (el pseudo-código para el método de HEVC WD5 de formación de la lista combinada LC), han sido marcados con un asterisco.

Tabla 4. Sintaxis de combinación de lista de imágenes de referencia

	Descriptor
<code>ref_pic_list_combination() {</code>	
<code>si(slice_type%5==1) {</code>	
<code>ref_pic_list_combination_flag</code>	<code>u(1)</code>
<code>si(ref_pic_list_combination_flag) {</code>	
<code>num_ref_idx_lc_active_minus1</code>	<code>ue(v)</code>
<code>ref_pic_list_modification_flag_lc</code>	<code>u(1)</code>
<code>si(ref_pic_list_modification_flag_lc)</code>	
<code>para (i=0; i<=num_ref_idx_lc_active_minus1; i++) {</code>	
<code>pic_from_list_0_flag</code>	<code>u(1)</code>
<code>si ((pic_from_list_0_flag==1 && num_ref_idx_l0_active_minus1 > 0)</code>	
<code>(pic_from_list_0_flag==0 && num_ref_idx_l1_active_minus1 > 0)</code>	
<code>ref_idx_list_curr</code>	<code>te(v) *</code>
<code>}</code>	
<code>}</code>	
<code>}</code>	

Obsérvese que la sintaxis `ref_idx_list_curr` está señalizada solamente cuando L0 (si `pic_from_lis_0_flag` es 1) o L1 (si `pic_from_list_1_flag` es 0) contiene más de 1 entrada puesto que no hay nada que necesite ser enviado si la lista correspondiente (ya sea L0 o ya sea L1) contiene solamente una entrada. De ese modo, la cantidad de señalización se reduce.

Adicionalmente, en vez de usar `ue(v)`, `te(v)` es una forma más eficiente para señalar `ref_idx_list_curr`, dado que el método de codificación de entropía `te(v)` (subcláusula 9.1 en H.264 [1]) está diseñado específicamente para codificar elementos de sintaxis como `ref_idx`. `Ue(v)` (conocido como código de Golomb Exponencial) puede usar tres bits para enviar el valor 1. Sin embargo, `te(v)` puede ser usado para determinar, en primer lugar, el número de valores posibles presentes en `ref_idx_list_curr` (buscando en L0 y L1), y si existen solamente dos valores, entonces el elemento de sintaxis puede ser enviado usando un bit. Si hay presentes más, entonces se puede usar `ue(v)`.

En otras palabras, si el elemento de sintaxis se codifica como `te(v)`, la gama de posibles valores para el elemento de sintaxis se determina en primer lugar. Si la gama de posibles valores para el elemento de sintaxis está entre 0 y 1, entonces solamente se usa un bit para codificar el elemento de sintaxis, ahorrando de ese modo carga de señalización. En otro caso, si el rango del elemento de sintaxis está entre 0 y x, siendo x mayor que 1, entonces se usa `ue(v)` para codificar el elemento de sintaxis.

Por lo tanto, el sistema realiza una determinación en base a los posibles valores de `ref_idx_list_curr`. Si existe solamente un valor posible para el elemento de sintaxis `ref_idx_list_curr`, entonces no se envía nada, ya que éste puede ser determinado tanto por el codificador como por el descodificador en base a otros valores. Si existen dos

valores posibles para el elemento de sintaxis `ref_idx_list_curr`, entonces se envía un bit. En otro caso, si existen tres o más posibles valores para el elemento de sintaxis `ref_idx_list_curr`, entonces se usa `ue(v)` para codificar `ref_idx_list_cur`.

5 Con ello, se realizan ahorros en la carga de señalización en comparación con HEVC WD5.

En una realización adicional, se divulga un proceso simple de modificación de lista de imágenes de referencia armonizado, que puede ser usado para modificar L0 y L1. De acuerdo con esta realización, el proceso de modificación de lista de imágenes de referencia para L0 y L1 utiliza la sintaxis mostrada en la Tabla 5. Los cambios en comparación con el pseudo-código de la Tabla 2 (es decir, la sintaxis de modificación de lista de imágenes de referencia para Lista 0 y Lista 1 en HEVC WD5), han sido marcados con un asterisco.

Tabla 5. Sintaxis de modificación de lista de imágenes de referencia para Lista 0 y Lista 1

	Descriptor
<code>ref_pic_list_modification() {</code>	
<code>si(slice_type != 2) { // P slice o B slice</code>	
<code>ref_pic_list_modification_flag_l0</code>	<code>u(1)</code>
<code>if(ref_pic_list_modification_flag_l0)</code>	
<code>para (i=0; i < num_ref_idx_l0_active_minus1; i++) {</code>	
<code>si (NumRpsCurrTempList > 1)</code>	
<code>ref_pic_set_idx</code>	<code>te(v) *</code>
<code>}</code>	
<code>}</code>	
<code>si(slice_type == 1) { // B slice</code>	
<code>ref_pic_list_modification_flag_l1</code>	<code>u(1)</code>
<code>si(ref_pic_list_modification_flag_l1)</code>	
<code>para (i=0, i <= num_ref_idx_l1_active_minus1; i++) {</code>	
<code>si (NumRpsCurrTempList > 1)</code>	
<code>ref_pic_set_idx</code>	<code>te(v) *</code>
<code>}</code>	
<code>}</code>	
<code>}</code>	

15 La semántica de modificación de lista de imágenes de referencia es como sigue:

El elemento de sintaxis `ref_pic_set_idx` se usa para especificar el cambio desde la lista de imágenes de referencia inicial a la lista de imágenes de referencia modificada.

20 `ref_pic_list_modification_flag_l0` igual a 1 especifica que el elemento de sintaxis `ref_pic_set_idx` está presente para especificar la lista 0 de imágenes de referencia.

`ref_pic_list_modification_flag_l0` igual a 0 especifica que este elemento de sintaxis no está presente.

`ref_pic_list_modification_flag_l1` igual a 1 especifica que el elemento de sintaxis `ref_pic_set_idx` está presente para especificar la lista 1 de imágenes de referencia.

25 `ref_pic_list_modification_flag_l1` igual a 0 especifica que este elemento de sintaxis no está presente.

`ref_pic_set_idx` especifica el índice de la imagen en `RefPicSetCurrTempListX` que ha de ser colocado en la posición actual de la lista LX de imágenes de referencia (donde X es 0 si se refiere a la lista L0 y X es 1 si se refiere a la lista L1). La sintaxis `ref_pic_set_idx` debe estar en el rango de 0 a `max_num_ref_frames-1` en la lista LX, inclusive. Si el elemento de sintaxis `ref_pic_set_idx` no está presente, se establece en 0.

30 El nuevo proceso reduce la señalización sustancialmente en algunos casos (y, en la práctica, probablemente en la mayor parte de los casos). Se ha constatado simplemente que, en vez de señalar para cada entrada en la lista el tipo de modificación que se va a realizar y el índice en la DPB de la imagen de referencia que va a ser usada, al igual que en la sintaxis de la Tabla 2 y en el diagrama de flujo de la Figura 8, el proceso inventivo señala solamente el índice en la DPB y no requiere ninguna señal adicional para indicar el final del proceso de modificación de lista.

35 El proceso descrito en la Tabla 5 anterior incluye el uso de la lista intermedia de imágenes de referencia para cada una de L0 y/o L1, `RefPicSetCurrTempListX`, donde la X representa 0 o 1 dependiendo de qué lista modificada esté siendo considerada. En este esquema, se proporciona un proceso de inicialización revisado para las listas de imágenes de referencia. Este proceso de inicialización se invoca cuando se descodifica una cabecera de una porción P o B. Cuando se descodifica una porción P o B, puede existir al menos una imagen de referencia en `RefPicSetStCurr0`, `RefPicSetStCurr1` o `RefPicSetStLtCurr`.

45 El procedimiento que sigue se implementa para construir `RefPicSetCurrTempList1`:

```

Cidx = 0
NumRpsCurrTempList = NumRpsStCurr1 + NumRpsLtCurr
para (i = 0); i < NumRpsStCurr0; cldx++, i++)
    RefPicSetCurrTempList0(cldx) = RefPicSetStCurr0(i)
5   para (l = 0; l < NumRpsStCurr1; cldx++, i++)
    RefPicSetCurrTempList0(cldx) = RefPicSetStCurr1(i)
    para (l = 0); l < NumRpsLtCurr; cldx++, i++)
        RefPicSetCurrTempList0(cldx) = RefPicSetLtCurr(i)

```

10 Si ref_pic_list_modification_flag_l0 es 0, entonces no se tiene que realizar ninguna modificación desde la lista 0 por defecto, y la RefPicList0 por defecto se construye tomando las primeras num_ref_idx_l0_active_minus1 + 1 entradas en RefPicSetCurrTempList0 por orden. Si, por otra parte, ref_pic_list_modification_flag_l0 es 1, el proceso de la Tabla 5 para modificación de la lista L0 de imágenes de referencia se invoca con RefPicSetCurrTempList0 y num_ref_idx_l0_active_minus1 como entradas, y RefPicList0 (L0) como salida.

15 De forma resumida, el pseudo-código anterior determina el número de imágenes de referencia en la DPB (es decir, NumRpsCurrTempList) sumando los números de las imágenes anterior, posterior y de largo plazo, y a continuación las pone por orden con las primeras imágenes anteriores (por orden en virtud de la distancia temporal más cercana a la distancia temporal más lejana desde la imagen actual), seguido por las imágenes posteriores (también por orden en virtud de la distancia temporal más cercana hasta la distancia temporal más lejana desde la imagen actual), seguido de las imágenes de referencia de largo plazo.

El procedimiento que sigue se implementa para construir RefPicSetCurrTempList1:

```

25   Cidx = 0
    NumRpsCurrTempList = NumRpsStCurr0 + NumRpsStCurr1 + NumRpsLtCurr
    para (i = 0); i < NumRpsStCurr1; cldx++, i++)
        RefPicSetCurrTempList1(cldx) = RefPicSetStCurr1(i)
    para (l = 0; l < NumRpsStCurr0; cldx++, i++)
30   RefPicSetCurrTempList1(cldx) = RefPicSetStCurr0(i)
    para (l = 0); l < NumRpsLtCurr; cldx++, i++)
        RefPicSetCurrTempList1(cldx) = RefPicSetLtCurr(i)

```

35 Si ref_pic_list_modification_flag_l1 es 0, entonces no se tendrá que realizar ninguna modificación a partir de la lista 1 por defecto y la RepPicList1 por defecto se reconstruye tomando las primeras num_ref_idx_l1_active_minus1 + 1 entradas en RefPicSetCurrTempList1 por orden. Si, por otra parte, ref_pic_list_modification_flag_l1 es 1, el proceso de modificación de la Tabla 5 para la modificación de lista L1 de imágenes de referencia se invoca con RefPicSetCurrTempList1 y num_ref_idx_l1_active_minus1 como entradas, y RefPicList1 como salida.

40 De forma resumida, el pseudo-código anterior determina el número de imágenes de referencia en la DPB (es decir, NumRpsCurrTempList) sumando los números de imágenes anteriores, posteriores y de largo plazo y colocándolas después por orden con las imágenes posteriores en primer lugar (por orden en virtud de la distancia temporal más cercana a la distancia temporal más lejana desde la imagen actual), seguido de las imágenes anteriores (también por orden en virtud de la distancia temporal más cercana a la distancia temporal más lejana desde la imagen actual), seguido de las imágenes de referencia de largo plazo.

45 Obsérvese que, la creación de las dos listas RpsCurrTempLX es beneficiosa incluso en aquellos casos en los no se tenga que realizar ninguna modificación en las listas L0 y L1 de imágenes de referencia debido a que, en esos casos, las listas L0 y L1 de imágenes de referencia pueden ser creadas de manera muy simple solamente tomando unas pocas primeras entradas en la RpsCurrTempLX puesto que éstas están ya en el orden por defecto para las listas L0 y L1, respectivamente.

50 El proceso de modificación para listas de imágenes de referencia según se refleja en la Tabla 5, admite como entradas la batería de imágenes de referencia mencionadas con anterioridad, RefPicSetCurrTempLX, y el tamaño de la lista de imágenes de referencia, num_ref_idx_IX_active_minus1 (siendo X, 0 o 1, dependiendo de qué lista se esté modificando). La salida de este proceso es una matriz que contiene la lista RefPicListX de imágenes de referencia modificada.

55 La Figura 10 es un diagrama de flujo que ilustra el proceso de modificación de lista de la Tabla 5, por ejemplo la lista L0. El proceso podría ser similar para la lista L1. El índice en la lista L0 se inicializa a cero en 1001. En 1003, se determina si la lista temporal RefPicSetCurrTempL0 contiene más de una entrada puesto que la señalización de ref_pic_set_idx no es necesaria si la lista contiene solamente una única entrada. Si la lista contiene solamente una entrada, el flujo avanza hasta 1004, donde ref_pic_set_idx no se señala y en cambio se establece en 0 por defecto. En otro caso, el flujo avanza hasta 1005, donde el índice, ref_pic_set_idx, en la lista intermedia

RefPicSetCurrTemplList0, se lee. En 1007, la entrada a la lista L0 modificada con el índice actual se establece en el valor de la lista RefPicSetCurrTemplList0 localizada en la posición de índice señalado ref_pic_set_idx. A continuación, el índice en la lista L0 modificada se incrementa (1009). En 1011, se determina si se ha alcanzado el final de L0. Si no es así, el flujo retorna a 1003. Si es así, el proceso termina.

5 Según se ha indicado previamente, si no se desea ninguna modificación de una lista, entonces el proceso de la Figura 10 no se lleva a cabo y las primeras num_ref_idx_lx_active_minus1 + 1 entradas de RefPicSetCurrTemplListX simplemente se convierten en la lista LX correspondiente.

10 La Figura 11 muestra cómo opera el esquema de lista de imágenes de referencia propuesto para el mismo ejemplo que en la Figura 9. Comparando la Figura 11 y la Figura 9, el proceso de modificación de la Figura 11 usa la mitad del número de elementos de sintaxis que en la Figura 9, es decir, solamente se señala ref_pic_set_idx para cada entrada en la lista L0, en vez de ref_pic_set_idx y list_modification_idc. Además, el proceso según se ha ilustrado en el diagrama de flujo de la Figura 10, es más directo que el proceso del diagrama de flujo de la Figura 8 en el sentido de que éste señala explícitamente cada entrada en la lista, y no necesita el complicado proceso de la Figura 8.

15 Los sistemas y métodos descritos en la presente memoria son muy adecuados para comunicaciones de corrientes de video tanto a través de redes alámbricas como a través de redes inalámbricas. Las redes alámbricas son bien conocidas. Una visión de conjunto de diversos tipos de dispositivos inalámbricos y de infraestructuras, se proporciona con respecto a las Figuras 12A-12E, donde varios elementos de la red pueden utilizar los sistemas y métodos descritos en la presente memoria. Más específicamente, estaciones de base tales como la estación transceptora de base (BTS), un Node-B, un eNodeB, un Node-B Doméstico, un eNodeB Doméstico, un controlador del lugar, un punto de acceso (AP), un enrutador inalámbrico, un elemento de red de reconocimiento multimedia (MANE), así como unidades de transmisión/recepción inalámbrica (WTRUs), pueden generar y/o procesar la señalización descrita en lo que antecede para transportar datos de video codificados desde una entidad a otra.

20 La Figura 12A es un diagrama de un ejemplo de sistema 100 de comunicaciones en donde pueden ser implementadas una o más de las realizaciones descritas. El sistema 100 de comunicaciones puede ser un sistema de acceso múltiple que proporciona contenido, tal como voz, datos, video, mensajería, radiodifusión, etc., a múltiples usuarios inalámbricos. El sistema 100 de comunicaciones puede habilitar a múltiples usuarios inalámbricos para acceder a tales contenidos a través de compartir recursos del sistema, incluyendo el ancho de banda inalámbrico. Por ejemplo, los sistemas 100 de comunicaciones pueden emplear uno o más métodos de acceso de canal, tal como acceso múltiple por división de código (CDMA), acceso múltiple por división de tiempo (TDMA), acceso múltiple por división de frecuencia (FDMA), FDMA ortogonal (OFDMA), FDMA de portadora simple (SC-FDMA), y similares.

25 Según se ha mostrado en la Figura 12A, el sistema 100 de comunicaciones puede incluir unidades de transmisión/recepción inalámbrica (WTRUs) 102a, 102b, 102c, 102d, una red de acceso de radio (RAN) 104, una red central 106, una red telefónica pública conmutada (PSTN) 108, Internet 110, y otras redes 112, aunque se podrá apreciar que las realizaciones descritas contemplan cualquier número de WTRUs, estaciones de base, redes, y/o elementos de red. Cada una de las WTRUs 102a, 102b, 102c, 102d puede ser cualquier tipo de dispositivo configurado para operar y/o comunicar en un entorno inalámbrico. A título de ejemplo, las WTRUs 102a, 102b, 102c, 102d pueden estar configuradas para transmitir y/o recibir señales inalámbricas y pueden incluir un equipo de usuario (UE), una estación móvil, una unidad de abonado fija o móvil, un radiobuscador, un teléfono celular, un asistente digital personal (PDA), un teléfono inteligente, un ordenador portátil, un netbook, un ordenador personal, un sensor inalámbrico, electrónica de consumo, y similares.

30 Los sistemas 100 de comunicaciones pueden incluir también una estación de base 114a y una estación de base 114b. Cada una de las estaciones de base 114a, 114b puede ser cualquier tipo de dispositivo configurado para intercomunicar inalámbricamente con al menos una de las WTRUs 102a, 102b, 102c, 102d para facilitar el acceso a una o más de las redes de comunicaciones, tal como la red central 106, Internet 110 y/o las redes 112. A título de ejemplo, las estaciones de base 114a, 114b pueden ser una estación transceptora de base (BTS), un Node-B, un eNodeB, un Node-B Doméstico, un eNodeB Doméstico, un controlador del sitio, un punto de acceso (AP), un enrutador inalámbrico, y similares. Mientras que las estaciones de base 114a, 114b han sido representadas, cada una de ellas, a modo de un elemento simple, se apreciará que las estaciones de base 114a, 114b pueden incluir cualquier número de estaciones de base interconectadas y/o de elementos de red.

35 La estación de base 114a puede ser parte de la RAN 104, la cual puede incluir también otras estaciones de base y/o elementos de red (no representados), tales como un controlador de estación de base (BSC), un controlador de red de radio (RNC), nodos de relé, etc. La estación de base 114a y/o la estación de base 114b pueden estar configuradas para transmitir y/o recibir señales inalámbricas dentro de una región geográfica particular, la cual puede ser mencionada como célula (no representada). La célula puede estar dividida además en sectores de célula. Por ejemplo, la célula asociada a la estación de base 114a puede estar dividida en tres sectores. De ese

modo, en una realización, la estación de base 114a puede incluir tres transceptores, es decir, uno por cada sector de la célula. En otra realización, la estación de base 114a puede emplear tecnología de múltiple entrada múltiple salida (MIMO) y, por lo tanto, puede utilizar múltiples transceptores por cada sector de la célula.

- 5 Las estaciones de base 114a, 114b pueden comunicar con una o más de las WTRUs 102a, 102b, 102c, 102d a través de una interfaz de aire 116, la cual puede ser cualquier enlace de comunicación inalámbrica adecuado (por ejemplo, radiofrecuencia (RF), microondas, infrarrojos (IR), ultravioleta (UV), luz visible, etc.). La interfaz de aire 116 puede ser establecida usando cualquier tecnología de acceso de radio (RAT) adecuada.
- 10 De manera más específica, según se ha indicado con anterioridad, el sistema 100 de comunicaciones puede ser un sistema de acceso múltiple y puede emplear uno o más esquemas de acceso de canal, tal como CDMA, TDMA, FDMA, OFDMA, SC-FDMA, y similares. Por ejemplo, la estación de base 114a en la RAN 104 y las WTRUs 102a, 102b, 102c pueden implementar una tecnología de radio tal como el Sistema Universal de Telecomunicaciones Móviles (UMTS), el Acceso de Radio Terrestre (UTRA), la cual puede establecer la interfaz
- 15 de aire 116 usando CDMA de banda ancha (WCDMA). La WCDMA puede incluir protocolos de comunicación tales como Acceso por Paquetes de Alta Velocidad (HSPA) y/o HSPA Evolucionado (HSPA+). El HSPA puede incluir Acceso por Paquetes de Enlace Descendente de Alta Velocidad (HSDPA) y/o Acceso por Paquetes de Enlace Ascendente de Alta Velocidad (HSUPA).
- 20 En otra realización, la estación de base 114a y las WTRUs 102a, 102b, 102c pueden implementar una tecnología de radio tal como Acceso de Radio Terrestre de UMTS Evolucionado (E-UTRA), la cual puede establecer la interfaz de aire 116 usando Evolución de Largo Plazo (LTE) y LTE Avanzado (LTE-A).
- 25 En otras realizaciones, la estación de base 114a y las WTRUs 102a, 102b, 102c pueden implementar tecnologías de radio tal como IEEE 802.16 (es decir, Interoperabilidad Mundial para Acceso de Microondas (WiMAX)), CDMA2000, CDMA2000 1X, CDMA2000 EV-DO, el Estándar 2000 Interim (IS-2000), el Estándar 95 Interim (IS-95), el Estándar 856 Interim (IS-856), el Sistema Global para Comunicaciones Móviles (GSM), tasas Aumentadas de Datos para GSM Evolution (EDGE), GSM EDGE (GERAN), y similares.
- 30 La estación de base 114b en la Figura 12A puede ser un enrutador inalámbrico, un Node-B Doméstico, un eNodeB Doméstico, o un punto de acceso, por ejemplo, y puede utilizar cualquier RAT adecuada para facilitar conectividad inalámbrica en un área localizada, tal como un sitio de oficinas, un domicilio, un vehículo, un campus, y similares. En una realización, la estación de base 114b y las WTRUs 102c, 102d pueden implementar una tecnología de radio tal como IEEE 802.11 para establecer una red de área local inalámbrica (WLAN). En otra
- 35 realización, la estación de base 114b y las WTRUs 102c, 102d pueden implementar una tecnología de radio tal como IEEE 802.15 para establecer una red de área personal inalámbrica (WPAN). Todavía en otra realización más, la estación de base 114b y las WTRUs 102c, 102d pueden utilizar una RAT de base celular (por ejemplo, WCDMA, CDMA2000, GSM, LTE, LTE-A, etc.) para establecer una picocélula o una femtocélula. Según se ha mostrado en la Figura 12A, la estación de base 114b puede tener una conexión directa a Internet 110. De ese
- 40 modo, la estación de base 114b puede no ser requerida para que acceda a Internet 110 a través de la red central 106.
- 45 La RAN 104 puede estar en comunicación con la red central 106, la cual puede ser cualquier tipo de red configurada para proporcionar voz, datos, aplicaciones, y/o servicios de voz sobre protocolo de Internet (VoIP) a una o más de las WTRUs 102a, 102b, 102c, 102d. Por ejemplo, la red central 106 puede proporcionar control de llamada, servicios de facturación, servicios basados en localización móvil, llamadas de pre-pago, conectividad de Internet, distribución de video, etc., y/o realizar funciones de seguridad de alto nivel, tales como autenticación de usuario. Aunque no se ha representado en la Figura 12A, se apreciará que la RAN 104 y/o la red central 106
- 50 pueden estar en comunicación directa o indirecta con otras RANs que empleen la misma RAT que la RAN 104 o una RAT diferente. Por ejemplo, además de estar conectada a la RAN 104, que puede estar utilizando una tecnología de radio de E-UTRA, la red central 106 puede estar también en comunicación con otra RAN (no representada) que emplee una tecnología de radio de GSM.
- 55 La red central 106 puede servir también como puerta de acceso para las WTRUs 102a, 102b, 102c, 102d, para acceder a la PSTN 108, a Internet 110, y/o a otras redes 112. La PSTN puede incluir redes de telefonía de circuito conmutado que proporcionen antiguos servicios de telefonía plana (POTS). Internet 110 puede incluir un sistema global de ordenador interconectas y dispositivos que usen protocolos de comunicación comunes, tales como el protocolo de control de transmisión (TCP), el protocolo de datagrama de usuario (UDP) y el protocolo de internet (IP) en la serie de protocolos de internet TCP/IP. Las redes 112 pueden incluir redes de comunicaciones
- 60 alámbricas o inalámbricas de propiedad y/u operadas por otros proveedores de servicio. Por ejemplo, las redes 112 pueden incluir otra red central conectada a una o más RANs, las cuales pueden emplear la misma RAT que la RAN 104 o una RAT diferente.
- 65 Algunas o todas las WTRUs 102a, 102b, 102c, 102d en el sistema 100 de comunicaciones, pueden incluir capacidades multimodo, es decir, las WTRUs 102a, 102b, 102c, 102d pueden incluir múltiples transceptores para

comunicar con diferentes redes inalámbricas a través de diferentes enlaces inalámbricos. Por ejemplo, la WTRU 102c mostrada en la Figura 12A puede estar configurada para comunicar con la estación de base 114a, la cual puede emplear una tecnología de radio de base celular, y con la estación de base 114b, la cual puede emplear una tecnología de radio de IEEE 802.

5 La Figura 12B es un diagrama de sistema de un ejemplo de WTRU 102. Según se ha mostrado en la Figura 12B, la WTRU 102 puede incluir un procesador 118, un transceptor 120, un elemento de transmisión/recepción 122, un altavoz/micrófono 124, un teclado 126, un visualizador/teclado táctil 128, memoria 106 no extraíble, memoria 132 extraíble, una fuente de alimentación 134, un chipset 136 para un sistema de posicionamiento global (GPS), y
10 otros periféricos 138. Se apreciará que la WTRU 102 puede incluir cualquier subcombinación de los elementos anteriores mientras se mantenga consistente con una realización.

15 El procesador 118 puede ser un procesador de propósito general, un procesador de propósito especial, un procesador convencional, un procesador de señal digital (DSP), una pluralidad de microprocesadores, uno o más microprocesadores en asociación con un DSP central, un controlador, un microcontrolador, Circuitos Integrados Específicos de la Aplicación (ASICs), circuitos de Matriz de Puerta Programable en Campo (FPGAs), cualquier otro tipo de circuito integrado (IC), una máquina de estado, y similares. El procesador 118 puede realizar codificación de señal, procesamiento de datos, control de potencia, procesamiento de entrada/salida, y/o cualquier otra funcionalidad que permita a la WTRU 102 operar en un entorno inalámbrico. El procesador 118
20 puede estar acoplado al transceptor 120, el cual puede estar acoplado al elemento 122 de transmisión/recepción. Mientras que la Figura 12B representa el procesador 118 y el transceptor 120 como componentes separados, se apreciará que el procesador 118 y el transceptor 120 pueden estar integrados conjuntamente en un empaquetamiento o chip electrónico.

25 El elemento 122 de transmisión/recepción puede estar configurado para transmitir señales a, o recibir señales desde, una estación de base (por ejemplo, la estación de base 114a) a través de la interfaz de aire 116. Por ejemplo, en una realización el elemento 122 de transmisión/recepción puede ser una antena configurada para transmitir y/o recibir señales de RF. En otra realización, el elemento 122 de transmisión/recepción puede ser un emisor/detector configurado para transmitir y/o recibir IR, UV o señales de luz visible, por ejemplo. Según otra
30 realización más, el elemento 122 de transmisión/recepción puede estar configurado para transmitir y recibir tanto RF como señales luminosas. Se apreciará que el elemento 122 de transmisión/recepción puede estar configurado para transmitir y/o recibir cualquier combinación de señales inalámbricas.

35 Adicionalmente, aunque el elemento 122 de transmisión/recepción esté representado en la Figura 12B como un solo elemento, la WTRU 102 puede incluir cualquier número de elementos 122 de transmisión /recepción. Más específicamente, la WTRU 102 puede emplear tecnología MIMO. De ese modo, en una realización, la WTRU 102 puede incluir dos o más elementos 122 de transmisión/recepción (por ejemplo, múltiples antenas) para transmitir y recibir señales inalámbricas a través de la interfaz de aire 116.

40 El transceptor 120 puede estar configurado para modular las señales que van a ser transmitidas por medio del elemento 122 de transmisión/recepción, y para demodular las señales que sean recibidas por el elemento 122 de transmisión/recepción. Según se ha indicado anteriormente, la WTRU 102 puede tener capacidades multimodo. De ese modo, el transceptor 120 puede incluir múltiples transceptores para permitir que la WTRU 102 comunique
45 a través de múltiples RATs, tales como UTRA e IEEE 802.11, por ejemplo.

El procesador 118 de la WTRU 102 puede estar acoplado a, y puede recibir datos de entrada de usuario desde, el altavoz/micrófono 124, el teclado 126, y/o el visualizador/teclado táctil 128 (por ejemplo, una unidad de visualización por display de cristal líquido (LCD) o una unidad de visualización por diodo orgánico de emisión de luz (OLED)). El procesador 118 puede también presentar a la salida datos de usuario para el altavoz/micrófono
50 124, el teclado 126 y/o el visualizador/teclado táctil 128. Adicionalmente, el procesador 118 puede acceder a información desde, y almacenar datos en, cualquier tipo de memoria adecuada, tal como la memoria 106 no extraíble y la memoria 132 extraíble. La memoria 106 no extraíble puede incluir una memoria de acceso aleatorio (RAM), memoria de sólo lectura (ROM), un disco duro, o cualquier otro tipo de dispositivo de almacenaje de memoria. La memoria 132 extraíble puede incluir una tarjeta de módulo de identidad de abonado (SIM), un lápiz de memoria, una tarjeta de memoria digital de seguridad (SD), y similares. En otras realizaciones, el procesador 118 puede acceder a información procedente desde, y almacenar datos en, una memoria que no esté físicamente ubicada en la WTRU 102, tal como un servidor o un ordenador doméstico (no representado).
55

60 El procesador 118 puede recibir energía desde una fuente de alimentación 134, y puede estar configurado para distribuir y/o controlar la potencia respecto a los otros componentes de la WTRU 102. La fuente de alimentación 134 puede ser cualquier dispositivo adecuado para alimentar la WTRU 102. Por ejemplo, la fuente de alimentación 134 puede incluir una o más pilas secas (por ejemplo, de níquel-cadmio (Ni/Cd), de níquel-zinc (NiZn), híbridas de metal y níquel (NiMH), de ion litio (Li-ion), etc.), células solares, pilas de combustible, y similares.
65

El procesador 118 puede estar también acoplado al chipset 136 de GPS, el cual puede estar configurado para proporcionar información de posición (por ejemplo, longitud y latitud) con relación a la posición actual de la WTRU 102. Adicionalmente a, o en lugar de, la información procedente del chipset 136 de GPS, la WTRU 102 puede recibir información de posición a través de la interfaz de aire 116 desde una estación de base (por ejemplo, las estaciones de base 114a, 114b) y/o determinar su posición en base al tiempo que tardan las señales en ser recibidas desde dos o más estaciones de base contiguas. Se apreciará que la WTRU 102 puede adquirir información de posición por medio de cualquier método de determinación de posición adecuado mientras que sea acorde con una realización.

El procesador 118 puede además estar acoplado a otros periféricos 138, los cuales pueden incluir uno o más módulos de software y/o hardware que proporcionen características adicionales, funcionalidad y/o conectividad alámbrica o inalámbrica. Por ejemplo, los periféricos 138 pueden incluir un acelerómetro, un compás electrónico, un transceptor por satélite, una cámara digital (para fotografías o vídeo), un puerto de bus serie universal (USB), un dispositivo vibrador, un transceptor de televisión, un teléfono de manos libres, un módulo de Bluetooth®, una unidad de radio de frecuencia modulada (FM), un reproductor de música digital, un reproductor multimedia, un módulo reproductor de juegos de vídeo, un navegador de internet, y similares.

La Figura 12C es un diagrama de sistema de la RAN 104 y de la red central 106 conforme a una realización. Según se ha indicado con anterioridad, la RAN 104 puede emplear tecnología de radio UTRA para comunicar con las WTRUs 102a, 102b, 102c, a través de la interfaz de aire 116. La RAN 104 puede estar también en comunicación con la red central 106. Según se muestra en la Figura 12C, la RAN 104 puede incluir Node-Bs 140a, 140b, 140c, cada uno de los cuales puede incluir uno o más transceptores para comunicar con las WTRUs 102a, 102b, 102c a través de la interfaz de aire 116. Los Node-Bs 140a, 140b, 140c pueden estar asociados, cada uno de ellos, a una célula particular (no representada) dentro de la RAN 104. La RAN 104 puede incluir también RNCs 142a, 142b. Se apreciará que la RAN 104 puede incluir cualquier número de Node-Bs y de RNCs mientras siga siendo acorde con una realización.

Según se ha mostrado en la Figura 12C, los Node-Bs 140a, 140b pueden estar en comunicación con el RNC 142a. Adicionalmente, el Node-B 140c puede estar en comunicación con el RNC 142b. Los Node-Bs 140a, 140b, 140c pueden comunicar con los respectivos RNCs 142a, 142b a través de una interfaz lub. Los RNCs 142a, 142b pueden estar en comunicación unos con otros a través de una interfaz lur. Cada uno de los RNCs 142a, 142b puede estar configurado para controlar los respectivos Node-Bs 140a, 140b, 140c a los que estén conectados. Adicionalmente, cada uno de los RNCs 142a, 142b puede estar configurado para llevar a cabo o soportar otra funcionalidad, tal como control de potencia de bucle externo, control de carga, control de admisión, programación por paquetes, control de traspaso, macrodiversidad, funciones de seguridad, encriptación de datos, y similares.

La red central 106 mostrada en la Figura 12C puede incluir una puerta de acceso multimedia (MGW) 144, un centro de conmutación móvil (MSC) 146, un nodo de soporte de GPRS de servidor (SGSN) 148, y/o un nodo de soporte de GPR de matriz de puerta de acceso (GGSN) 150. Mientras que cada uno de los elementos anteriores ha sido representado como parte de la red central 106, se apreciará que uno cualquiera de esos elementos puede ser en propiedad y/o estar operado por una entidad distinta del operador de la red central.

El RNC 142a en la RAN 104 puede estar conectado al MSC 146 en la red central 106 a través de una interfaz luCS. El MSC 146 puede estar conectado a la MGW 144. El MSC 146 y la MGW 144 pueden dotar a las WTRUs 102a, 102b, 102c con acceso a las redes de circuito conmutado, tal como la PSTN 108, para facilitar las comunicaciones entre las WTRUs 102a, 102b, 102c y los servicios de comunicaciones tradicionales de telefonía fija.

El RNC 142a en la RAN 104 puede estar también conectado al SGSN 148 en la red central 106 a través de una interfaz luPS. El SGSN 148 puede estar conectado al GGSN 150. El SGSN 148 y el GGSN 150 pueden dotar a las WTRUs 102a, 102b, 102c de acceso a las redes conmutadas por paquetes, tal como Internet 110, para facilitar las comunicaciones entre las WTRUs 102a, 102b, 102c y los dispositivos habilitados en IP.

Según se ha indicado con anterioridad, la red central 106 puede estar también conectada a las redes 112, las cuales pueden incluir otras redes alámbricas o inalámbricas que sean de propiedad y/o estén operadas por otros proveedores de servicio.

La Figura 12D es un diagrama de sistema de la RAN 104 y la red central 106 conforme a otra realización. Según se ha indicado anteriormente, la RAN 104 puede emplear tecnología de radio de E-UTRA para comunicar con las WTRUs 102a, 102b, 102c a través de la interfaz de aire 116. La RAN 104 puede estar también en comunicación con la red central 106.

La RAN 104 puede incluir eNodeBs 160a, 160b, 160c, a través de los cuales se apreciará que la RAN 104 puede incluir cualquier número de eNode-Bs mientras se mantenga consistente con una realización. Los eNode-Bs 160a, 160b, 160c pueden incluir, cada uno de ellos, uno o más transceptores para comunicar con las WTRUs 102a, 102b, 102c a través de la interfaz de aire 116. En una realización, los eNode-Bs 106a, 106b, 106c pueden

implementar tecnología MIMO. Por lo tanto, el eNode-B 160a, por ejemplo, puede usar múltiples antenas para transmitir señales inalámbricas hasta, y recibir señales inalámbricas desde, la WTRU 102a.

5 Cada uno de los eNode-Bs 160a, 160b, 160c puede estar asociado a una célula particular (no representada) y puede estar configurado para manejar decisiones de gestión de recursos de radio, decisiones de traspaso, programación de usuarios en el enlace ascendente y/o enlace descendente, y similares. Según se muestra en la Figura 12D, los eNode-Bs 160a, 160b, 160c pueden comunicar entre sí a través de una interfaz X2.

10 La red central 106 mostrada en la Figura 12D puede incluir una puerta de acceso 162 para gestión de movilidad (MME), una puerta de acceso 164 para el servidor, y una puerta de acceso 166 de red de datos por paquetes (PDN). Mientras que cada uno de los elementos anteriores ha sido representado como parte de la red central 106, se apreciará que cualquiera de esos elementos puede ser de propiedad y/u operado por una entidad distinta al operador de la red central.

15 El MME 162 puede estar conectado a cada uno de los eNode-Bs 160a, 160b, 160c en la RAN 104 a través de una interfaz S1 y puede servir como nodo de control. Por ejemplo, el MME 162 puede ser responsable de autenticar usuarios de las WTRUs 102a, 102b, 102c, soportar activación/desactivación, seleccionar una puerta de acceso de servidor particular durante un anexo inicial de las WTRUs 102a, 102b, 102c, y similares. El MME 162 puede proporcionar también una función de plano de control para conmutar entre la RAN 104 y otras RANs (no representadas) que empleen otras tecnologías de radio, tal como GSM o WCDMA.

20 La puerta de acceso 164 del servidor puede estar conectada a cada uno de los eNode-Bs 160a, 160b, 160c en la RAN 104 a través de la interfaz S1. La puerta de acceso 164 del servidor puede enrutar generalmente, y enviar, paquetes de datos de usuario hasta/desde las WTRUs 102a, 102b, 102c. La puerta de acceso 164 del servidor puede realizar también otras funciones, tal como anclaje de planos de usuario durante los traspasos inter-eNode B, disparo de rasdiobúsqueda cuando se encuentran disponibles datos de enlace descendente para las WTRUs 102a, 102b, 102c, gestión y almacenaje de contextos de las WTRUs 102a, 102b, 102c, y similares.

25 La puerta de acceso 164 del servidor puede también estar conectada a la puerta de acceso 166 de PDN, la cual puede dotar a las WTRUs 102a, 102b, 102c con acceso a las redes conmutadas por paquetes, tal como Internet 110, para facilitar las comunicaciones entre las WTRUs 102a, 102b, 102c y los dispositivos habilitados de IP.

30 La red central 106 puede facilitar comunicaciones con otras redes. Por ejemplo, la red central 106 puede dotar a las WTRUs 102a, 102b, 102c con acceso a las redes de circuito conmutado, tal como la PSTN 108, para facilitar comunicaciones entre las WTRUs 102a, 102b, 102c y dispositivos de comunicaciones convencionales de telefonía fija. Por ejemplo, la red central 106 puede incluir, o puede comunicar con, una puerta de acceso de IP (por ejemplo, un servidor de subsistema multimedia de IP (IMS)), que sirva como interfaz entre la red central 106 y la PSTN 108. Adicionalmente, la red central 106 puede dotar a las WTRUs 102a, 102b, 102c con acceso a las redes 112, las cuales pueden incluir otras redes alámbricas o inalámbricas que sean de propiedad y/o que estén operadas por otros proveedores de servicio.

35 La Figura 12E es un diagrama de sistema de la RAN 104 y la red central 106 conforme a otra realización. La RAN 104 puede ser una red de servicio de acceso (ASN) que emplee tecnología de radio de IEEE 802.16 para comunicar con las WTRUs 102a, 102b, 102c a través de la interfaz de aire 116. Según se va a discutir mejor más adelante, los enlaces de comunicaciones entre entidades funcionales diferentes de las WTRUs 102a, 102b, 102c, la RAN 104 y la red central 106, pueden ser definidas como puntos de referencia.

40 Según se ha mostrado en la Figura 12E, la RAN 104 puede incluir estaciones de base 170a, 170b, 170c, y una puerta de acceso 172 de ASN, aunque se apreciará que la RAN 104 puede incluir cualquier número de estaciones de base y de puertas de acceso de ASN mientras sean acordes con una realización. Las estaciones de base 170a, 170b, 170c pueden estar asociadas, cada una de ellas, con una célula particular (no representada) en la RAN 104 y pueden incluir, cada una de ellas, uno o más transceptores para comunicar con las WTRUs 102a, 102b, 102c a través de la interfaz de aire 116. En una realización, las estaciones de base 170a, 170b, 170c pueden implementar tecnología MIMO. De ese modo, la estación de base 170a, por ejemplo, puede usar múltiples antenas para transmitir señales inalámbricas a, y recibir señales inalámbricas desde, la WTRU 102a. Las estaciones de base 170a, 170b, 170c pueden proporcionar también funciones de gestión de movilidad, tal como disparo de manos libres, establecimiento de túnel, gestión de recursos de radio, clasificación de tráfico, aplicación de políticas de calidad de servicio (QoS), y similares. La puerta de acceso 172 de ASN puede servir como punto de agregación de tráfico y puede ser responsable de radiobúsqueda, almacenamiento en caché de perfiles de abonado, enrutamiento hasta la red central 106, y similares.

45 La interfaz de aire 116 entre las WTRUs 102a, 102b, 102c y la RAN 104 puede ser definida como un punto de referencia R1 que implemente la especificación IEEE 802.16. Adicionalmente, cada una de las WTRUs 102a, 102b, 102c puede establecer una interfaz lógica (no representada) con la red central 106. La interfaz lógica entre las WTRUs 102a, 102b, 102c y la red central 106 puede ser definida como punto de referencia R2, el cual puede

ser usado para autenticación, autorización gestión de configuración de anfitrión de IP, y/o gestión de movilidad.

El enlace de comunicaciones entre cada una de las estaciones de base 170a, 170b, 170c puede ser definido como un punto de referencia R8 que incluye protocolos para facilitar los trasposos de WTRU y la transferencia de datos entre estaciones de base. El enlace de comunicaciones entre las estaciones de base 170a, 170b, 170c y la puerta de acceso 172 de ASN puede ser definido como punto de referencia R6. El punto de referencia R6 puede incluir protocolos para facilitar gestión de movilidad en base a eventos de movilidad asociados a cada una de las WTRUs 102a, 102b, 102c.

Según se ha mostrado en la Figura 12E, la RAN 104 puede estar conectada a la red central 106. El enlace de comunicaciones entre la RAN 104 y la red central 106 puede definirse como punto de referencia R3 que incluye protocolos para facilitar capacidades de transferencia de datos y de gestión de movilidad, por ejemplo. La red central 106 puede incluir un agente doméstico de IP móvil (MIP-HA) 174, un servidor 176 de autenticación, autorización y contabilidad (AAA) 176, y una puerta de acceso 178. Mientras que cada uno de los elementos que anteceden ha sido representado como parte de la red central 106, se apreciará que uno cualquiera de esos elementos puede ser de propiedad y/u operado por una entidad distinta del operador de la red central.

El MIP-HA 174 puede ser responsable de la gestión de la dirección de IP, y puede habilitar las WTRUs 102a, 102b, 102c en cuanto a itinerancia para diferentes ASNs y/o diferentes redes centrales. El MIP-HA 174 puede dotar a las WTRUs 102a, 102b, 102c con acceso a las redes conmutadas por paquetes, tal como Internet 110, para facilitar comunicación entre las WTRUs 102a, 102b, 102c y los dispositivos habilitados de IP. El servidor 176 de AAA puede ser responsable de la autenticación del usuario y de soportar servicios de usuario. La puerta de acceso 178 puede facilitar interfuncionamiento con otras redes. Por ejemplo, la puerta de acceso 178 puede proporcionar a las WTRUs 102a, 102b, 102c acceso a redes de circuito conmutado, tal como la PSTN 108, para facilitar comunicación entre las WTRUs 102a, 102b, 102c y dispositivos de comunicación convencionales de telefonía fija. Adicionalmente, la puerta de acceso 178 puede proporcionar a las WTRUs 102a, 102b, 102c acceso a las redes 112, las cuales pueden incluir otras redes alámbricas o inalámbricas que sean de propiedad y/o estén operadas por otros proveedores de servicio.

Aunque no se ha representado en la Figura 12E, se apreciará que la RAN 104 puede estar conectada a otras ASNs y la red central 106 puede estar conectada a otras redes centrales. El enlace de comunicaciones entre la RAN 104 y otras ASNs puede ser definido como un punto de referencia R4, el cual puede incluir protocolos para coordinar la movilidad de las WTRUs 102a, 102b, 102c entre la RAN 104 y las otras ASNs. El enlace de comunicaciones entre la red central 106 y las otras redes centrales puede ser definido como una referencia R5, la cual puede incluir protocolos para facilitar el interfuncionamiento entre redes centrales domésticas y redes centrales visitadas.

Realizaciones

En una realización, se implementa un método de generación de listas L0 y L1 de imágenes de referencia para descodificar una imagen de video dentro de datos de video, comprendiendo el método generar una primera lista ordenada de imágenes de referencia a partir de una memoria intermedia de imágenes de referencia (DPB), RefPicSetCurrTempList0, en donde la lista está ordenada con las imágenes de referencia en la DPB que son temporalmente anteriores a la imagen actual, si las hay, listadas por orden en virtud de la distancia temporal desde la imagen actual, seguido de las imágenes de referencia en la DBP que son temporalmente posteriores a la imagen actual, si las hay, listadas por orden en virtud de la distancia temporal desde la imagen actual, seguido de imágenes de referencia de largo plazo en la DPB, si las hay, listadas en el orden en el que éstas se encuentran almacenadas en la DPB; generar una segunda lista ordenada de imágenes de referencia a partir de la DPB, RefPicSetCurrTempList1, en donde la lista está ordenada con las imágenes de referencia de la DPB que son temporalmente posteriores a la imagen actual, si las hay, listadas en primer lugar por orden en virtud de la distancia temporal desde la imagen actual, seguido de las imágenes de referencia de la DPB que son temporalmente anteriores a la imagen actual, si las hay, listadas por orden en virtud de la distancia temporal desde la imagen actual, seguido de imágenes de referencia de largo plazo en la DPB, si las hay, listadas en el orden en que se encuentran almacenadas en la DPB; y, generar al menos una de las listas L0 y L1 seleccionando imágenes de referencia a partir de RefPicSetCurrTempList0 y de RefPicSetCurrTempList1, respectivamente.

De acuerdo con esta realización, el método puede comprender además determinar si cualquiera de las listas L0 y L1 son listas modificadas, en donde, si la lista L0 debe ser una lista modificada, entonces generar lista L0 comprende, para cada imagen de referencia en la lista L0 de imágenes de referencia, recibir un primer índice en la primera lista ordenada y listar la imagen de referencia identificada como ese índice en la primera lista ordenada en la entrada correspondiente en L0; y, si la lista L0 debe ser una lista modificada, entonces generar la lista L1 comprende, para cada entrada de imagen de referencia en la lista L1 de imágenes de referencia, recibir un segundo índice en la segunda lista ordenada y listar la imagen de referencia identificada en ese índice en la segunda lista ordenada en la entrada correspondiente en L1.

Una o más de las realizaciones precedentes puede comprender, además, si la lista L0 no ha de ser una lista

modificada, generar entonces la lista L0 comprende tomar entradas desde RefPicSetCurrTempList0 por orden hasta un primer número especificado de entradas; y, si la lista L1 no ha de ser una lista modificada, entonces generar la lista L1 comprende tomar entradas desde RefPicSetCurrTempList1 por orden hasta un segundo número específico de entradas.

5 Una o más de las realizaciones precedentes puede comprender además el hecho de que la determinación comprende leer el elemento de sintaxis ref_pic_list_modification_flag_I0 con respecto a la lista L0 y el elemento de sintaxis ref_pic_list_modification_flag_I1 con respecto a la lista L1.

10 Una o más de las realizaciones precedentes puede comprender además el hecho de que el primer índice y el segundo índice estén en la gama desde cero hasta el número de imágenes en la DPB.

Una o más de las realizaciones precedentes puede comprender además el hecho de que un elemento de sintaxis ref_pic_set_idx se use para especificar el primer índice y el segundo índice.

15 Una o más de las realizaciones precedentes pueden comprender además: leer un elemento de sintaxis, ref_pic_list_modification_flag_I1, en donde ref_pic_list_modification_flag_I1 igual a un primer valor especifica que el elemento de sintaxis ref_pic_set_idx está presente para especificar L1 y en donde ref_pic_list_modification_flag_I1 igual a un segundo valor especifica que el elemento de sintaxis no está presente para especificación de L1.

20 Una o más de las realizaciones precedentes pueden comprender, además: leer un elemento de sintaxis ref_pic_list_modification_flag_I0, en donde ref_pic_list_modification_flag_I0 igual a un primer valor especifica que el elemento de sintaxis ref_pic_seat_idx está presente para especificar L0, y en donde ref_pic_list_modification_flag_I0 igual a un segundo valor especifica que este elemento de sintaxis no está presente para especificación de L0.

25 Una o más de las realizaciones precedentes pueden comprender, además: leer un elemento de sintaxis, ref_pic_list_modification_flag_I1, en donde ref_pic_list_modification_flag_I1 igual a un primer valor especifica que el elemento de sintaxis ref_pic_set_idx está presente para especificación de L1, y en donde ref_pic_list_modification_flag_I1 igual a un segundo valor especifica que este elemento de sintaxis no está presente para especificación de L1.

30 Una o más de las realizaciones anteriores puede comprender, además: en donde, si el primer índice no está presente, éste se establece en cero y, si el segundo índice no está presente, éste se establece en cero.

35 En otra realización o en relación con cualquiera de las realizaciones descritas en lo que antecede, un método para inicialización de un descodificador para listas de imágenes de referencia para descodificar una cabecera de porción P o B, puede comprender:

40 construir una primera lista temporal, RefPicSetCurrTempList0, mediante:

```

45     cldx = 0
        NumRpsCurrTempList = NumRpStCurr0 + NumRpsStCurr1 + NumRpsLtCurr
        para {i = 0; i < NumRpsStCurr0; cldx++, i++}
            RefPicSetCurrTempList0 [cldx] = RefPicSetStCurr0 [i]
        para (i = 0; i < NumRpsStCurr1; cldx++, i++)
            RefPicSetCurrTempList0 [cldx] = RefPicSetStCurr1 [i]
50     para (i = 0; i < NumRpsLtCurr; cldx++, i++)
            RefPicSetCurrTempList0 [cldx] = RefPicSetLtCurr [i]

```

55 Una o más de las realizaciones precedentes puede comprender además construir una lista, L0 (RefPicList0), si una banderola, ref_pic_list_modification_flag_I0, es 0, tomando las primeras num_ref_idx_I0_active_minus1 + 1 entradas en RefPicSetCurrTempList0.

Una o más de las realizaciones precedentes pueden comprender además construir la lista L0, si la banderola, ref_pic_list_modification_flag_I0, es 1, invocando una modificación del proceso de listas de imágenes con RefPicSeatCurrTempList0 y num_ref_idx_I0_active_minus1 como entradas.

60 Una o más de las realizaciones anteriores pueden comprender, si se descodifica una cabecera de porción B, construir una segunda lista temporal, RefPicSetCurrTempList1 mediante:

```

65     cldx = 0
        NumRpsCurrTempList = NumRpStCurr0 + NumRpsStCurr1 + NumRpsLtCurr
        para {i = 0; i < NumRpsCurr1; cldx++, i++}

```

```

RefPicSetCurrTempList1 [cldx] = RefPicSetCurr1 [i]
para (i = 0; i < NumRpsCurr0; cldx++, i++)
RefPicSetCurrTempList1 [cldx] = RefPicSetCurr0 [i]
para (i = 0; i < NumRpsLtCurr; cldx++, i++)
5 RefPicSetCurrTempList1 [cldx] = RefPicSetLtCurr [i]

```

Una o más de las realizaciones precedentes pueden comprender además construir una lista, L1 (RefPicList1), si una banderola (ref_pic_list_modification_flag_l1) es 0 tomando las primeras num_ref_idx_l1_active_minus1 + 1 entradas en RefPicSetCurrTempList1.

10 Una o más de las realizaciones precedentes pueden comprender además construir la lista L1 (RefPicList1), si la banderola (ref_pic_list_modification_flag_l1) es 1 invocando un proceso de modificación de lista de imágenes de referencia con RefPicSetCurrTempList1 y num_ref_idx_l1_active_minus1 como entradas.

15 Una o más de las realizaciones precedentes pueden comprender, además: en donde, el proceso de modificación de lista de imágenes de referencia se usa para generar RefPicListX, en donde X designa la correspondiente lista 0 o 1, en donde:

```

20 establecer refIdxLX como índice en la lista de imágenes de referencia RefPicListX, y
repetir iterativamente hasta que refIdxLX sea mayor que num_ref_idx_LX_active_minus1 + 1
- RefPicListX [refIdxLX++] = RefPicSetCurrTempLX [ref_pic_set_idx]

```

25 En otra realización o en relación con cualquiera de las realizaciones descritas en lo que antecede, un método de señalización de modificaciones para una pluralidad de listas de imágenes de referencia puede comprender señalar las modificaciones para la pluralidad de listas de imágenes de referencia usando una sintaxis de señalización unificada.

30 Una o más de las realizaciones precedentes pueden comprender, además: en donde, la pluralidad de listas de imágenes de referencia incluye L0, L1 y una lista combinada, LC.

Una o más de las realizaciones precedentes pueden comprender, además: en donde, la sintaxis de señalización unificada incluye codificar un índice de una imagen de referencia usando un método de codificación de entropía.

35 Una o más de las realizaciones precedentes puede comprender, además: en donde, la sintaxis de señalización unificada incluye codificar el índice de una imagen de referencia usando te(v).

40 En otra realización o en relación con cualquiera de las realizaciones descritas con anterioridad, un método puede comprender: determinar un número de entradas en una lista de imágenes de referencia; generar un mensaje que incluya un valor identificativo de una entrada en la lista de imágenes de referencia, en donde el valor está representado por un solo bit si el número de entradas en la lista de imágenes de referencia es dos, estando el valor representado por múltiples bits si el número de entradas en la lista de imágenes de referencia es tres o más, y omitiendo el mensaje el valor si el número de entradas en la lista de imágenes de referencia es uno.

45 Una o más de las realizaciones precedentes puede comprender, además: en donde, el valor es ue(v) cuando el número de entradas es tres o más valores.

Una o más de las realizaciones precedentes pueden comprender, además: en donde, el índice se especifica mediante el elemento de sintaxis ref_idx_list_curr.

50 En otra realización o en relación con cualquiera de las realizaciones descritas con anterioridad, un método de creación de una lista combinada, LC, de imágenes de referencia para ser usada para la descodificación de una porción B a partir de una primera lista de imágenes de referencia, L0, y una segunda lista de imágenes de referencia, L1, puede comprender: determinar si L0 contiene más de una entrada; determinar si L1 contiene más de una entrada; si L0 o L1 contienen más de una entrada, usar un elemento de sintaxis ref_idx_list_curr para indicar las entradas en al menos una de entre L0 y L1 a ser añadidas en LC; si L0 contiene solamente una entrada, establecer ref_idx_list_curr en 0; si L1 contiene solamente una entrada, establecer ref_idx_list_curr en 0; y, crear LC usando el valor de ref_idx_list_curr.

60 Una o más de las realizaciones precedentes pueden comprender, además: en donde, la determinación de si L0 contiene más de una entrada comprende determinar si un elemento de sintaxis num_ref_idx_l0_active_minus1 es cero o mayor, y la determinación de si L1 contiene más de una entrada comprende determinar si un elemento de sintaxis num_ref_idxl1_active_minus1 es cero o mayor.

Conclusión

[1] ITU-T Rec .264 y ISO/IEC/MPEG 4, parte 10, Codificación avanzada de video para servicios audiovisuales genéricos, Noviembre de 2007.

[2] SMPTE 421M, "VC-1Formato Comprimido de Corriente de Bits de Video y Proceso de Decodificación", Abril de 2008.

[3] JM software.hhi.de/suehring/tml/download/jm18.2.zip, Noviembre de 2011.

[4] B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, T. Wiegand, WD5: Borrador de Trabajo 5 de Codificación de Video de Alta Eficiencia, documento núm. JCTVC-G1103, Noviembre de 2011.

[5] K. McCann, S. Sekiguci, B. Bross, W.-J. Han, HM5: HEVC Modelo de Prueba 5 Descripción del Codificador. Documento núm. JCTVC-G1102, Diciembre de 2011.

[6] J. Boyce, R. Sjoberg, Y. K. Wang, informe BoG: Memoria intermedia de imágenes de referencia y construcción de lista. Documento núm. JCTVC-G1002, Noviembre de 2011.

[7] D. Flynn, R. Sjoberg, et al., JCTVC Informe de AhG: Memoria intermedia de imágenes de referencia y construcción de lista. Documento núm. JCTVC-G1002, Noviembre de 2011.

[8] Y. Suzuki, et al. Extensión de simplificación uni-predicción en porciones B. Documento núm. JCTVC-D421, Enero de 2011.

[9] B. Bross, W.-J. Han, J.-R. Ohm, G.J. Sullivan, T. Wiegand. Borrador de Trabajo WD9 de Codificación de Video de Alta Eficacia. Documento núm. JCVTC-K1103, Octubre 2012.

Aunque las características y los elementos han sido descritos en lo que antecede en combinaciones particulares, un experto en la materia podrá apreciar que cada una de las características o de los elementos pueden ser usados solos o en cualquier combinación con las otras características y elementos. Adicionalmente, los métodos descritos pueden ser implementados en un programa informático, software o firmware incorporado en un medio legible con ordenador para su ejecución por medio de un ordenador o un procesador. Ejemplos de medios de almacenaje legibles con ordenador no transitorios incluyen, aunque sin limitación, una memoria de solo lectura (ROM), memoria de acceso aleatorio (RAM), un registro, memoria caché, dispositivos de memoria semiconductores, medios magnéticos tales como discos duros internos y discos extraíbles, medios magneto-ópticos, y medios ópticos tales como discos CD-ROM, y discos versátiles digitales (DVDs). Se puede usar un procesador asociado a software para implementar un transceptor de radiofrecuencia para su uso en una WTRU, un UE, un terminal, una estación de base, un RNC, o cualquier otro ordenador anfitrión.

Además, en las realizaciones descritas con anterioridad, se observan plataformas de procesamiento, sistemas de computación, controladores y otros dispositivos que contienen procesadores. Estos dispositivos pueden contener al menos una Unidad Central de Procesamiento ("CPU") y memoria. Conforme a la práctica de los expertos en la materia sobre programación informática, se puede hacer referencia a actos y representaciones simbólicas de operadores o instrucciones mediante las diversas CPUs y memorias. Tales actos y operaciones o instrucciones pueden ser mencionadas para ser "ejecutadas", "ejecutadas con ordenador" o "ejecutadas con CPU".

Un experto en la materia podrá apreciar que los actos y las operaciones representadas simbólicamente o las instrucciones, incluyen la manipulación de señales eléctricas por medio de la CPU. Un sistema eléctrico representa bits de datos que pueden causar una transformación o reducción resultante de las señales eléctricas y el mantenimiento de bits de datos en posiciones de memoria en un sistema de memoria para reconfigurar con ello o alterar de otro modo la operación de la CPU, así como otro procesamiento de señales. Las posiciones de memoria donde se mantienen los bits de datos son posiciones físicas que tienen propiedades eléctricas, magnéticas, ópticas u orgánicas particulares, correspondientes a, o representativas de, los bits de datos.

Los bits de datos pueden ser también mantenidos en un medio legible con ordenador que incluye discos magnéticos, discos ópticos, o cualquier otro medio de almacenaje masivo volátil (por ejemplo, Memoria de Acceso Aleatorio (RAM)) o no volátil (por ejemplo, Memoria de Solo Lectura ("ROM")), legible por medio de la CPU. El medio legible con ordenador puede incluir medios legibles con ordenador cooperantes o interconectados, que existen exclusivamente en el sistema de procesamiento o que están distribuidos entre múltiples sistemas de procesamiento interconectados que pueden ser locales o remotos respecto al sistema de procesamiento. Se entiende que los ejemplos de realización no se limitan a las memorias mencionadas con anterioridad y que otras plataformas y memorias pueden soportar los métodos descritos.

Ningún elemento, acto o instrucción usados en la descripción de la presente solicitud, deberán ser entendidos como críticos o esenciales a menos que se describan explícitamente como tales. También, según se usa en la presente memoria, el artículo "un" se entiende que incluye uno o más objetos. Donde solamente se destine un objeto, se usa el término "uno" o un lenguaje similar. Además, los términos "cualquiera de" seguido de un listado de una pluralidad de objetos y/o una pluralidad de categorías de objetos, según se usa en la presente memoria, está destinado a incluir "cualquiera de", "cualquier combinación de", "cualquier múltiplo de", y/o "cualquier combinación de múltiplos de" los objetos y/o las categorías de objetos", individualmente o conjuntamente con otros objetos y/u otras categorías de objetos. Además, según se usa en la presente memoria, el término "conjunto" se entiende que incluye cualquier número de objetos, incluido el cero. Además, según se usa en la

presente memoria, el término “número” se pretende que incluya cualquier número, incluyendo el cero.

Además, las reivindicaciones no deberán ser leídas como limitadas al orden o a los elementos descritos a menos que se destinen a tal efecto.

- 5 Aunque los sistemas y métodos de la presente memoria han sido descritos en términos de sistema de comunicación multibanda UWB, se contempla que puedan ser implementados en software o en microprocesadores/ordenadores de propósito general (no representados). En determinadas realizaciones, una o más de las funciones de los diversos componentes pueden ser implementadas en software que controle un
- 10 ordenador de propósito general.

REIVINDICACIONES

1. Un aparato codificador de video para su uso en la codificación de una imagen predicha dentro de una corriente de bits (120, 202) de datos de video, que comprende:

5 un procesador configurado para generar una lista temporal ordenada de imágenes de referencia (301) a partir de una memoria intermedia de imágenes descodificadas, DPB, en donde la lista está ordenada con cualesquiera imágenes de referencia en la DPB que son temporalmente anteriores a una imagen que está actualmente siendo codificada y listada por orden en virtud de la distancia temporal desde la imagen que está siendo actualmente codificada, seguida de cualesquiera imágenes de referencia en la DPB que son temporalmente posteriores a la imagen que está siendo actualmente codificada y listada por orden en virtud de la distancia temporal desde la imagen que está siendo actualmente codificada, seguido de cualesquiera imágenes de referencia de largo plazo en la DPB listadas en el orden en el que están almacenadas en la DPB; y,

10 el procesador está configurado además para generar una lista de imágenes de referencia seleccionando imágenes de referencia a partir de la lista temporal ordenada de imágenes de referencia, en donde, cuando la lista de imágenes de referencia va a ser una lista modificada, generar la lista de imágenes de referencia comprende, para cada entrada en la lista de imágenes de referencia, seleccionar una imagen de referencia a partir de la lista ordenada temporal de imágenes de referencia para que ocupen la entrada en la lista de imágenes de referencia, listar la imagen de referencia seleccionada a la entrada en la lista de imágenes de referencia, generar un índice correspondiente en la lista ordenada temporal de imágenes de referencia en donde la índice identifica la imagen de referencia seleccionada dentro de la lista ordenada temporal de imágenes de referencia, y codificar el índice en la corriente de bits.

25 2. El aparato codificador de video de la reivindicación 1, en donde el procesador está configurado además para escribir datos en la corriente de bits indicativos de si la lista de imágenes de referencia debe ser una lista modificada.

30 3. El aparato codificador de video de la reivindicación 2, en donde el procesador está además configurado para, si la lista de imágenes de referencia no debe ser una lista modificada, generar la lista de imágenes de referencia tomando entradas a partir de la lista ordenada temporal de imágenes de referencia por orden hasta un primer número específico de entradas.

35 4. El aparato codificador de video de la reivindicación 3, en donde el índice está en la gama de cero hasta un número de imágenes en la DPB.

5. El aparato codificador de video de la reivindicación 2, en donde el índice se especifica mediante un elemento de sintaxis `ref_pic_set_idx`.

40 6. El aparato codificador de video de la reivindicación 5, en donde el procesador está además configurado para escribir en la corriente de bits un elemento de sintaxis, `ref_pic_list_modification_flag_l0`, en donde si `ref_pic_list_modification_flag_l0` es igual a un primer valor, especifica que el elemento de sintaxis `ref_pic_set_idx` está presente en la corriente de bits para su uso en la generación de la lista de imágenes de referencia, y en donde si `ref_pic_list_modification_flag_l0` es igual a un segundo valor, especifica que el elemento de sintaxis `ref_pic_set_idx` no está presente en la corriente de bits para su uso en la generación de la lista de imágenes de referencia.

45

7. Un método de generación de una lista de imágenes de referencia para su uso en la codificación de una imagen predicha dentro de una corriente de bits (120, 202) de datos de video, comprendiendo el método:

50 generar una lista temporal ordenada de imágenes de referencia (301) a partir de una memoria intermedia de imágenes descodificadas, DPB, en donde la lista está ordenada con cualesquiera imágenes de referencia de la DPB que sean temporalmente anteriores a una imagen que está siendo actualmente codificada y listada por orden en virtud de la distancia temporal a partir de la imagen que está siendo actualmente codificada, seguido de cualesquiera imágenes de referencia en la DPB que sean temporalmente posteriores a la imagen que está siendo actualmente codificada y listada por orden en virtud de la distancia temporal desde la imagen que está siendo actualmente codificada, seguido de cualesquiera imágenes de referencia de largo plazo en la DPB, listadas en el orden en que éstas se encuentran almacenadas en la DPB; y,

55 generar una lista de imágenes de referencia mediante selección de imágenes de referencia a partir de la lista temporalmente ordenada de imágenes de referencia, en donde, cuando la lista de imágenes de referencia debe ser una lista modificada, generar la lista de imágenes de referencia comprende, para cada una de las entradas de la lista de imágenes de referencia, seleccionar una imagen de referencia a partir de la lista temporal ordenada de imágenes de referencia para que ocupen la entrada en la lista de imágenes de referencia, listar la imagen de referencia seleccionada a la entrada en la lista de imágenes de referencia, generar un índice correspondiente en la lista temporal ordenada de imágenes de referencia en donde el índice identifica la imagen de referencia seleccionada dentro de la lista temporal ordenada de imágenes de

60

65

referencia, y codificar el índice en la corriente de bits.

- 5 8. El método de la reivindicación 7, que comprende además:
escribir datos en la corriente de bits indicativos de si la lista de imágenes de referencia debe ser una lista modificada.
- 10 9. El método de la reivindicación 8, en donde:
si la lista de imágenes de referencia no debe ser una lista modificada, generar la lista de imágenes de referencia
comprende tomar entradas a partir de la lista temporal ordenada de imágenes de referencia por orden hasta un
primer número específico de entradas.
- 15 10. El método de la reivindicación 8, en donde el índice está en la gama de cero a un número de imágenes en la
DPB.
- 20 11. El método de la reivindicación 8, en donde el índice está especificado por un elemento de sintaxis,
ref_pic_set_idx.
12. El método de la reivindicación 11, que comprende además:
escribir en la corriente de bits un elemento de sintaxis, ref_pic_list_modification_flag_I0, en donde si
ref_pic_list_modification_flag_I0 es igual a un primer valor, especifica que el elemento de sintaxis ref_pic_set_idx
está presente en la corriente de bits para su uso en la generación de la lista de imágenes de referencia, y en donde
si ref_pic_list_modification_flag_I0 es igual a un segundo valor, especifica que el elemento de sintaxis
ref_pic_set_idx no está presente en la corriente de bits para su uso en la generación de la lista de imágenes de
referencia.

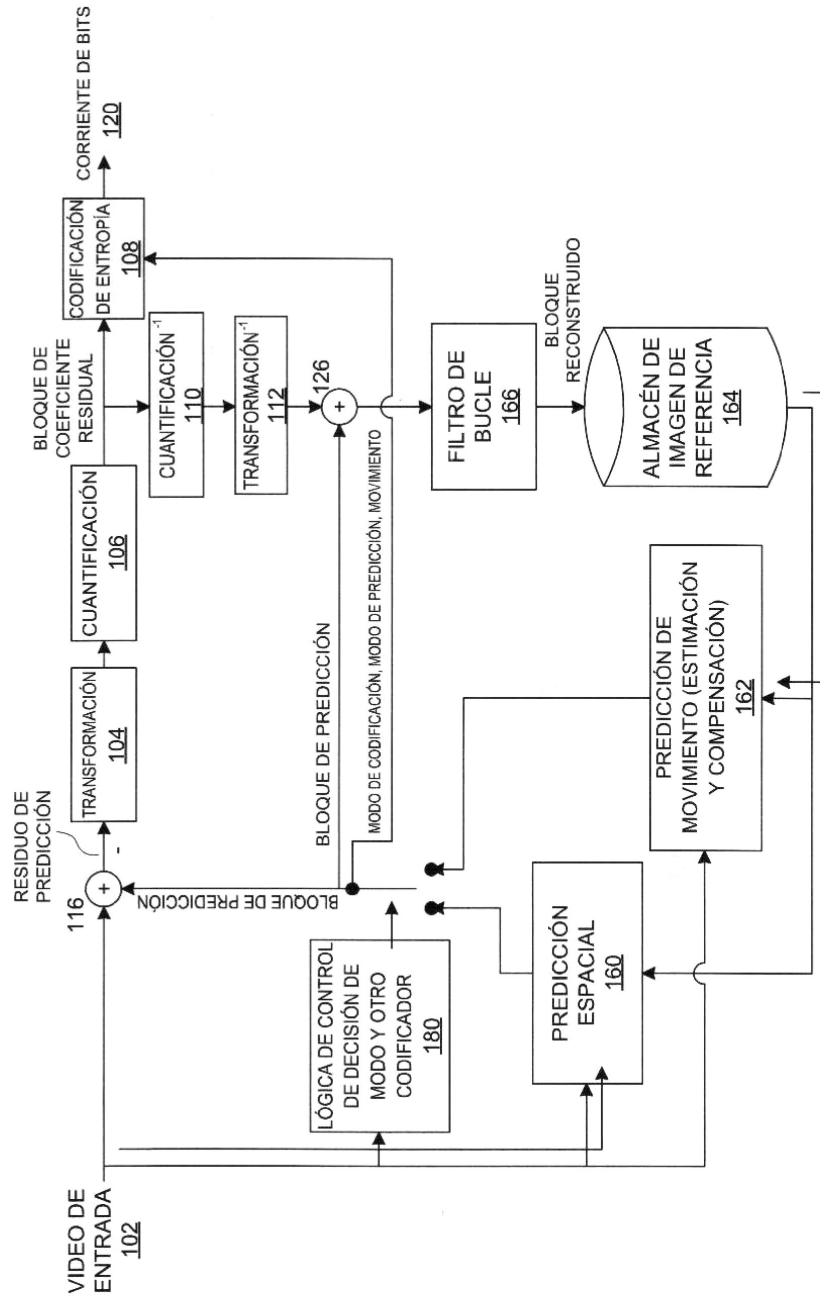


FIG. 1 (TÉCNICA ANTERIOR)

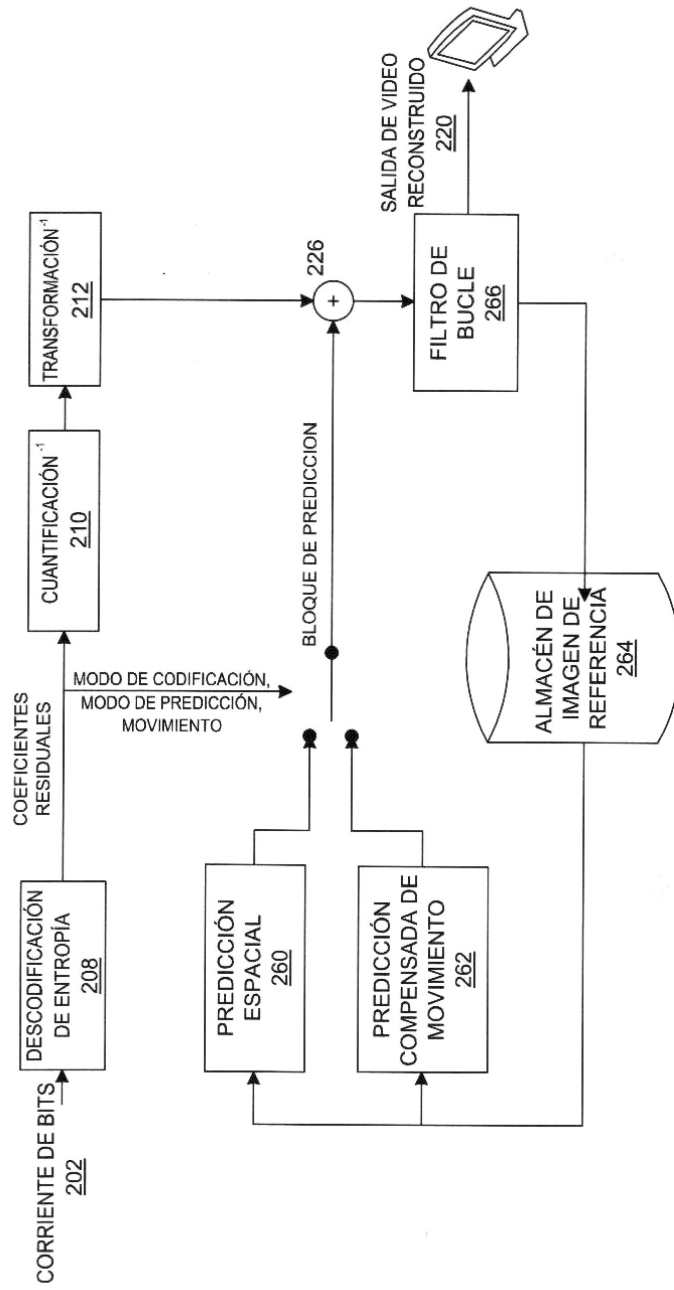


FIG. 2 (TÉCNICA ANTERIOR)

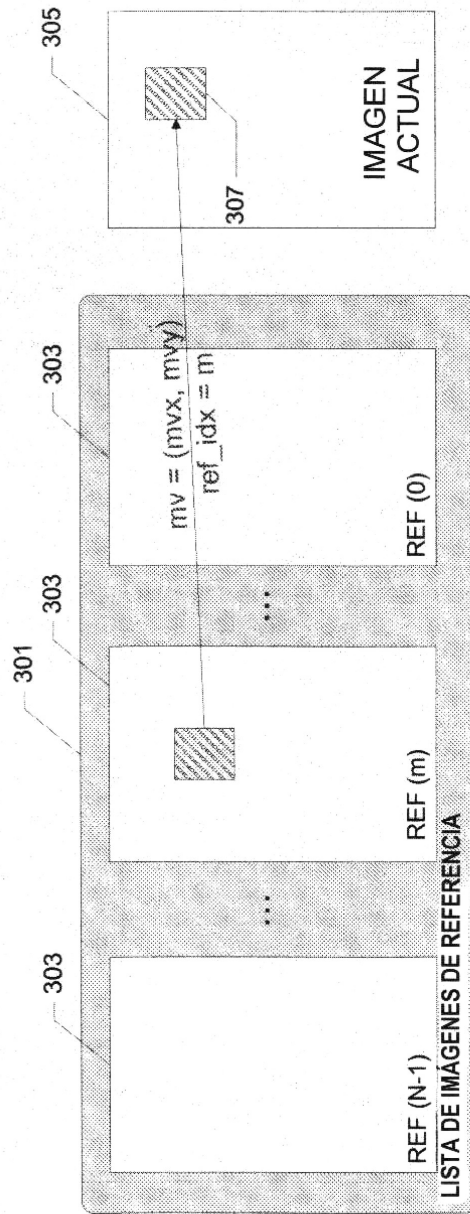


FIG. 3 (TÉCNICA ANTERIOR)

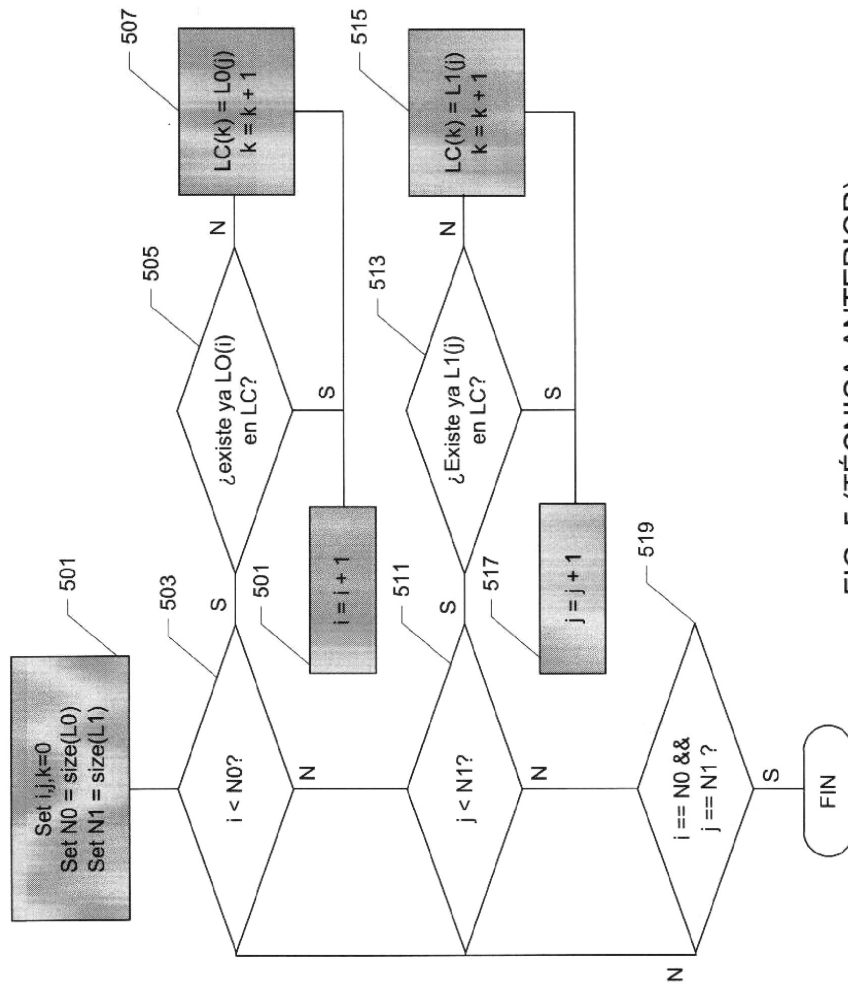


FIG. 5 (TÉCNICA ANTERIOR)

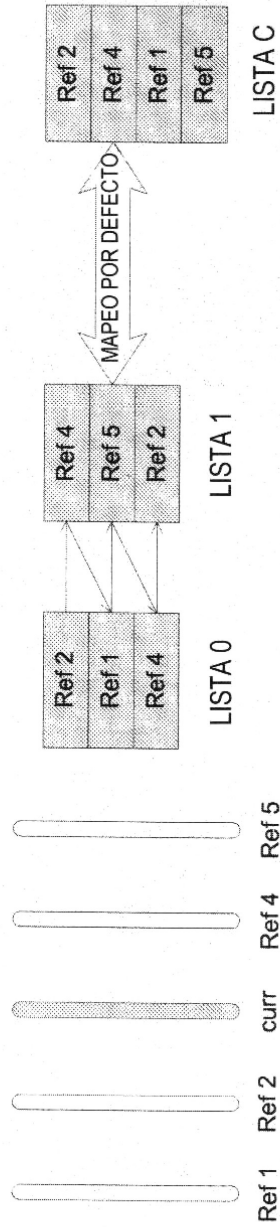


FIG. 6 (TÉCNICA ANTERIOR)

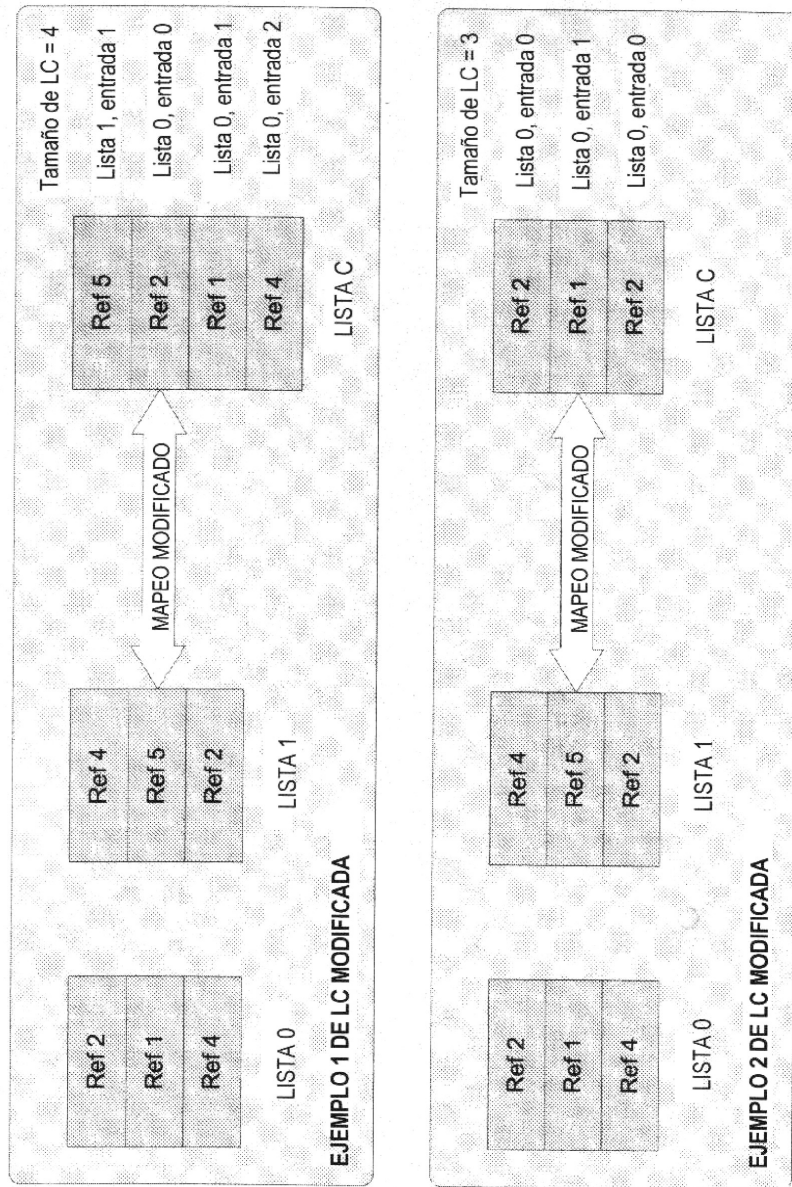


FIG. 7 (TÉCNICA ANTERIOR)

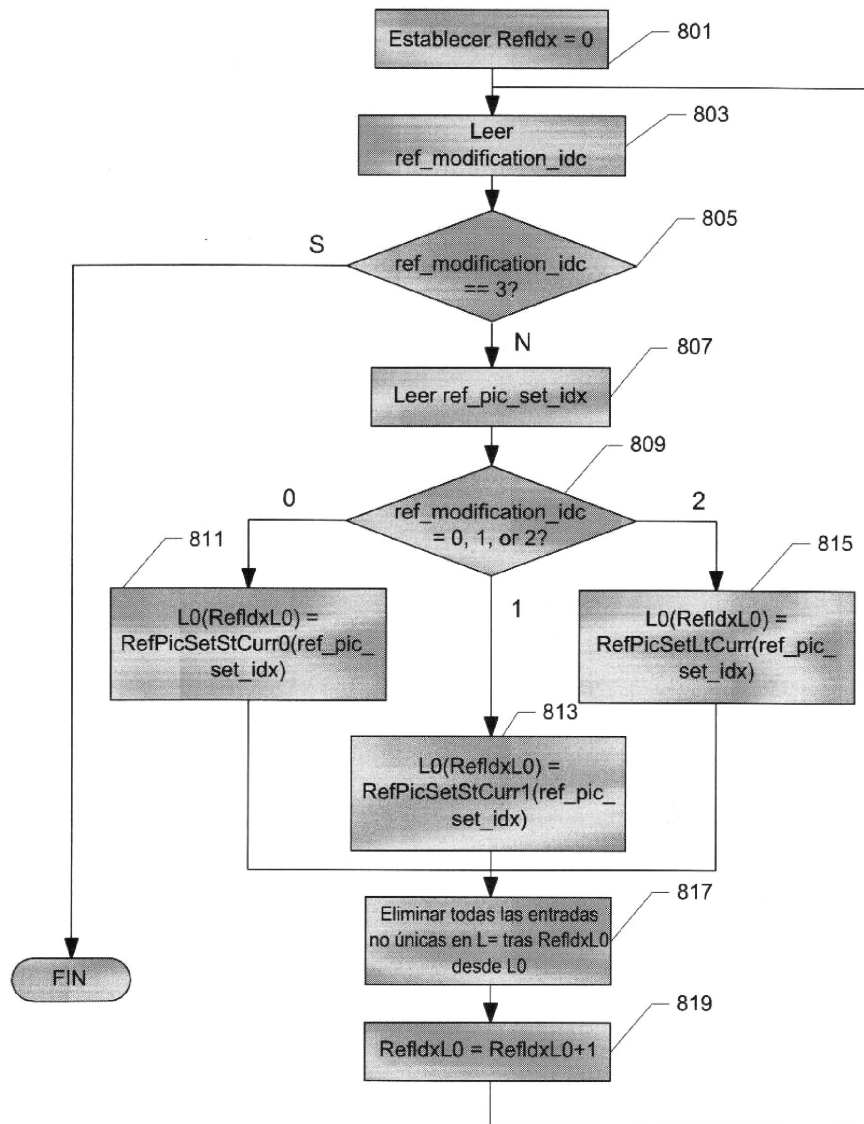


FIG. 8 (TÉCNICA ANTERIOR)

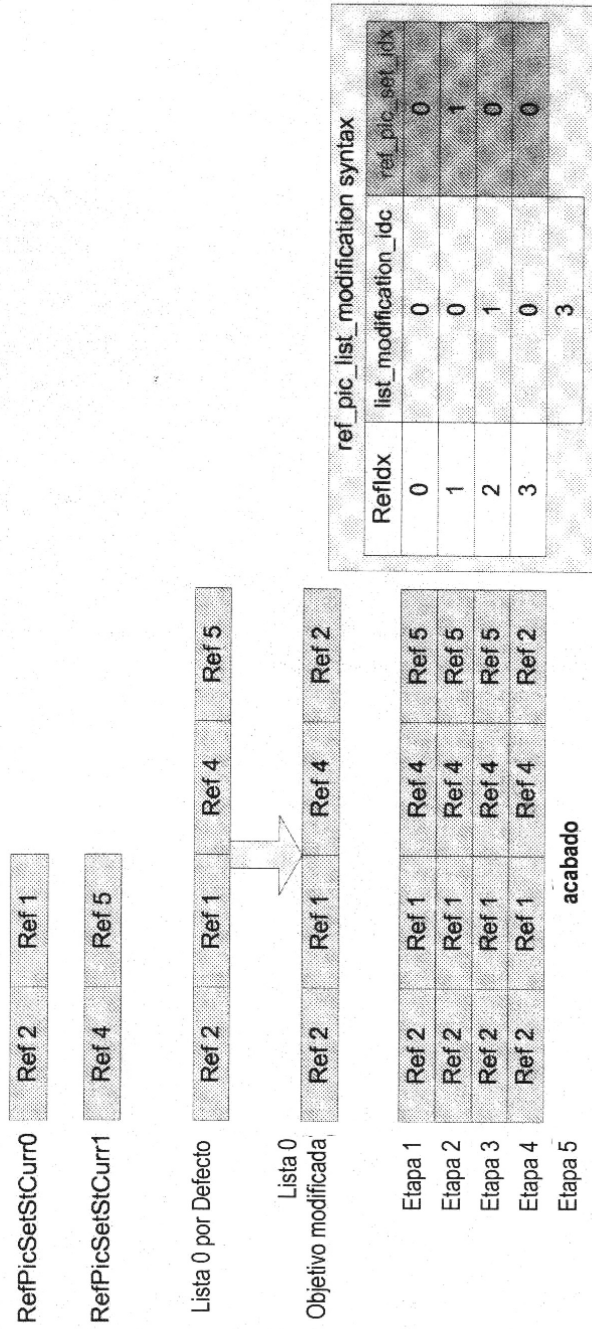


FIG. 9 (TÉCNICA ANTERIOR)

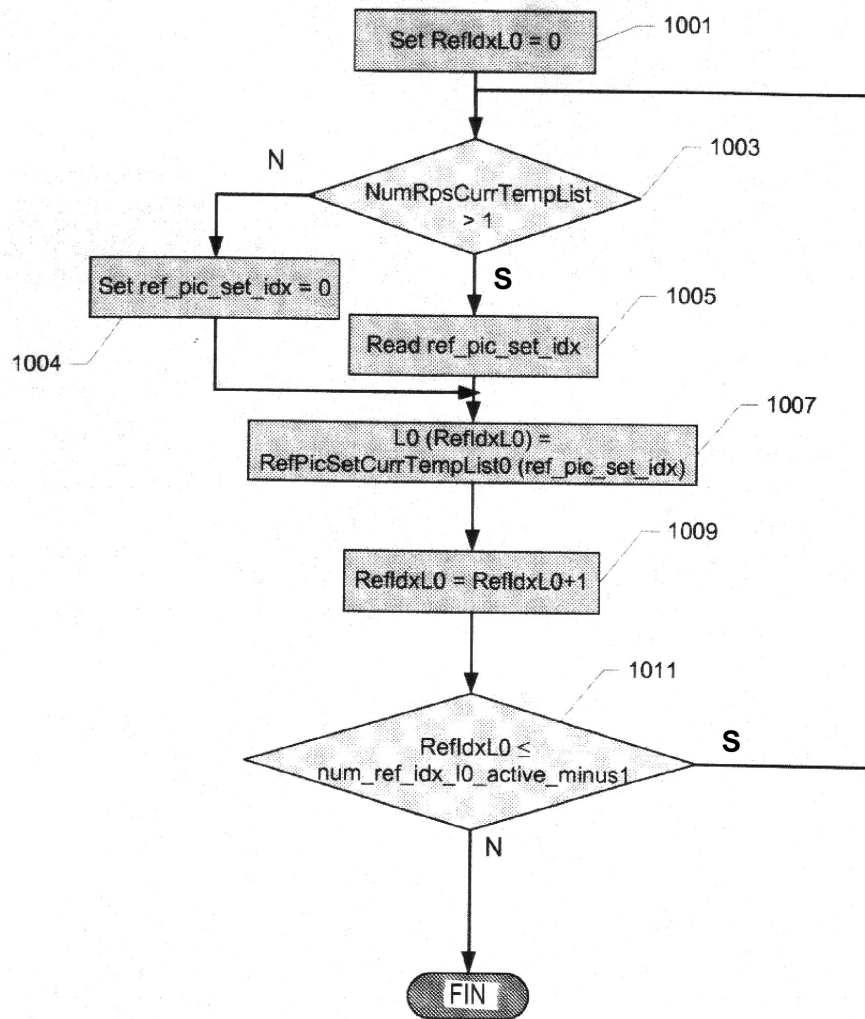


FIG. 10

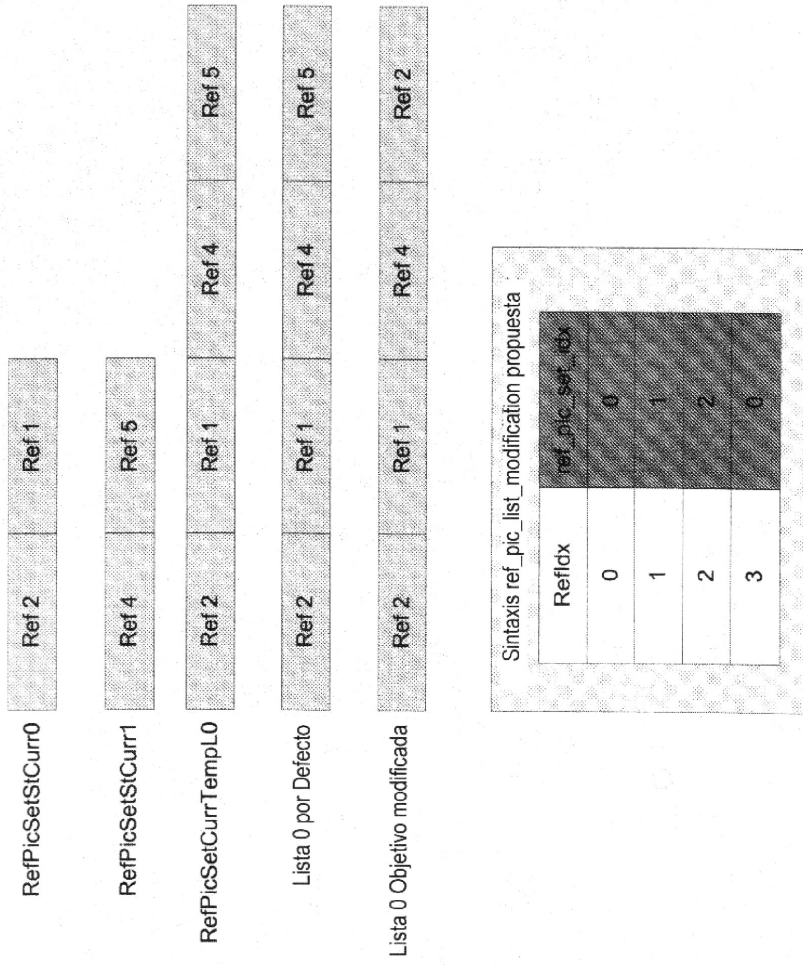


FIG. 11

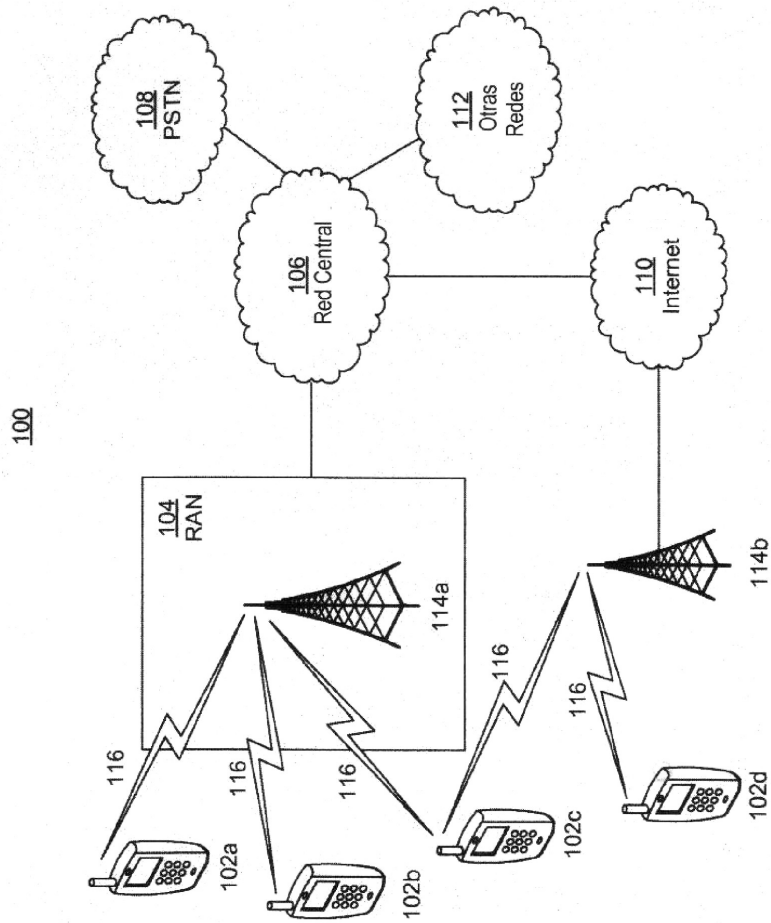


FIG. 12A (TÉCNICA ANTERIOR)

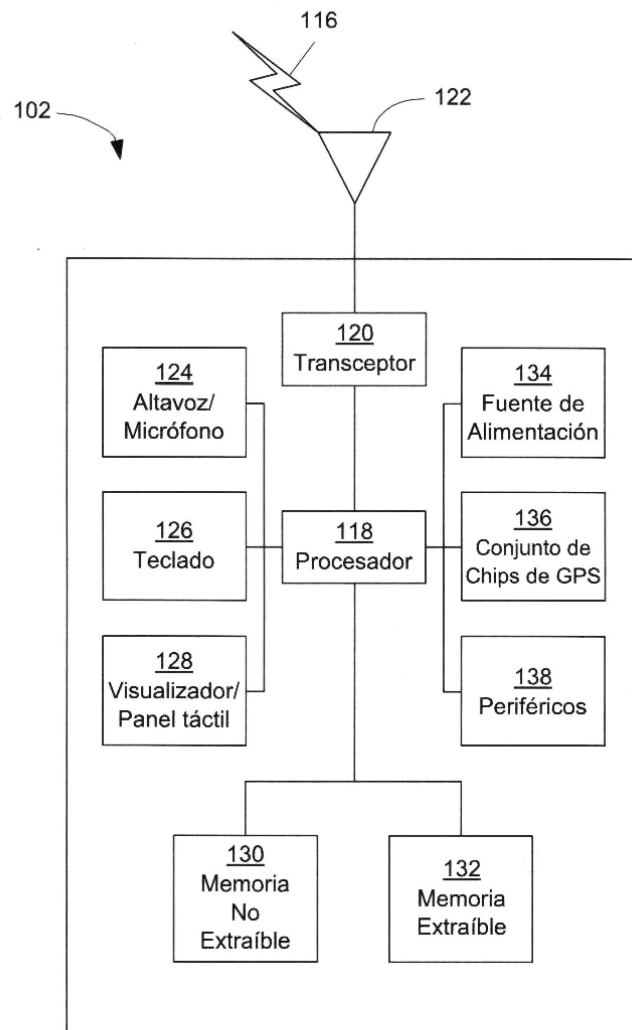


FIG. 12B (TÉCNICA ANTERIOR)

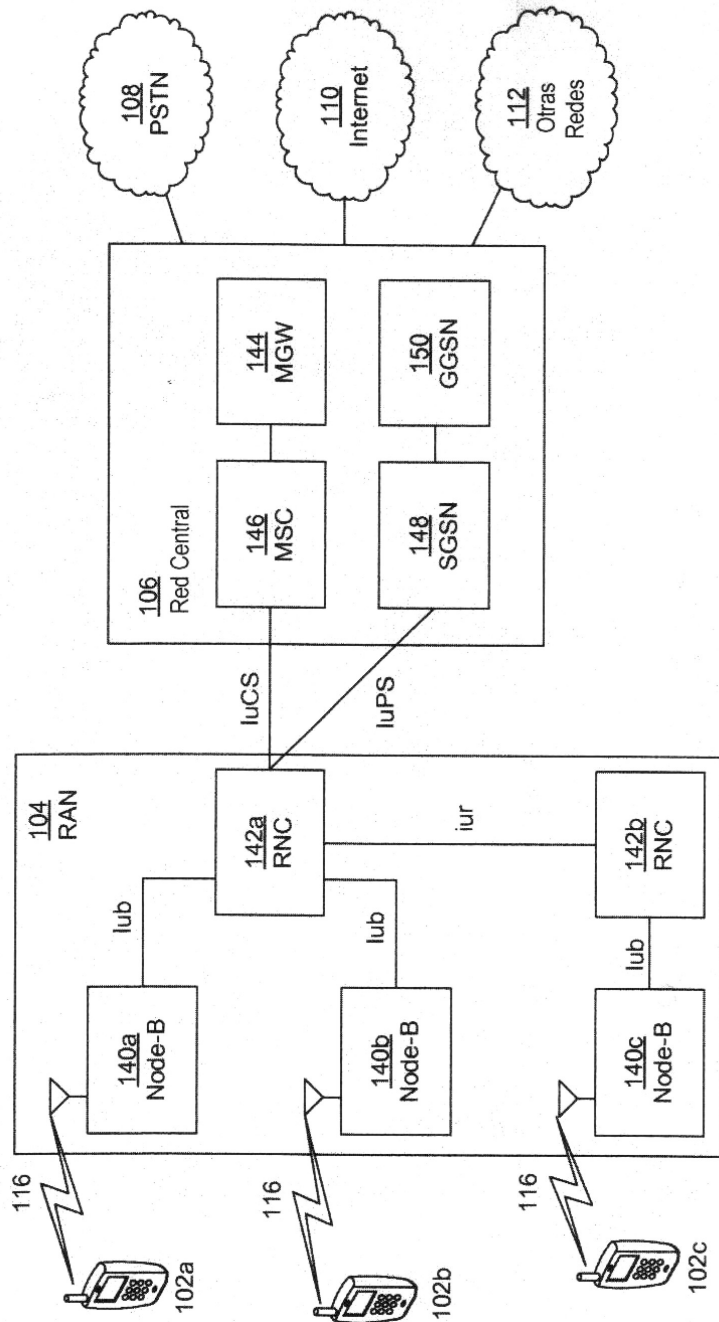


FIG. 12C (TÉCNICA ANTERIOR)

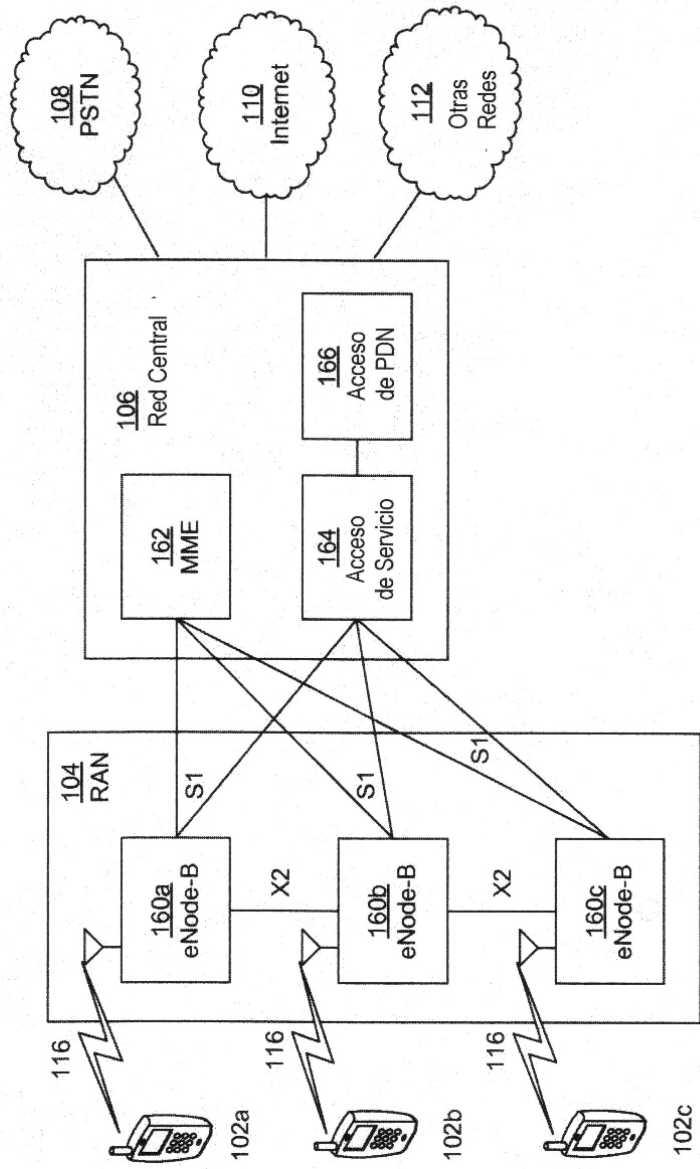


FIG. 12D (TÉCNICA ANTERIOR)

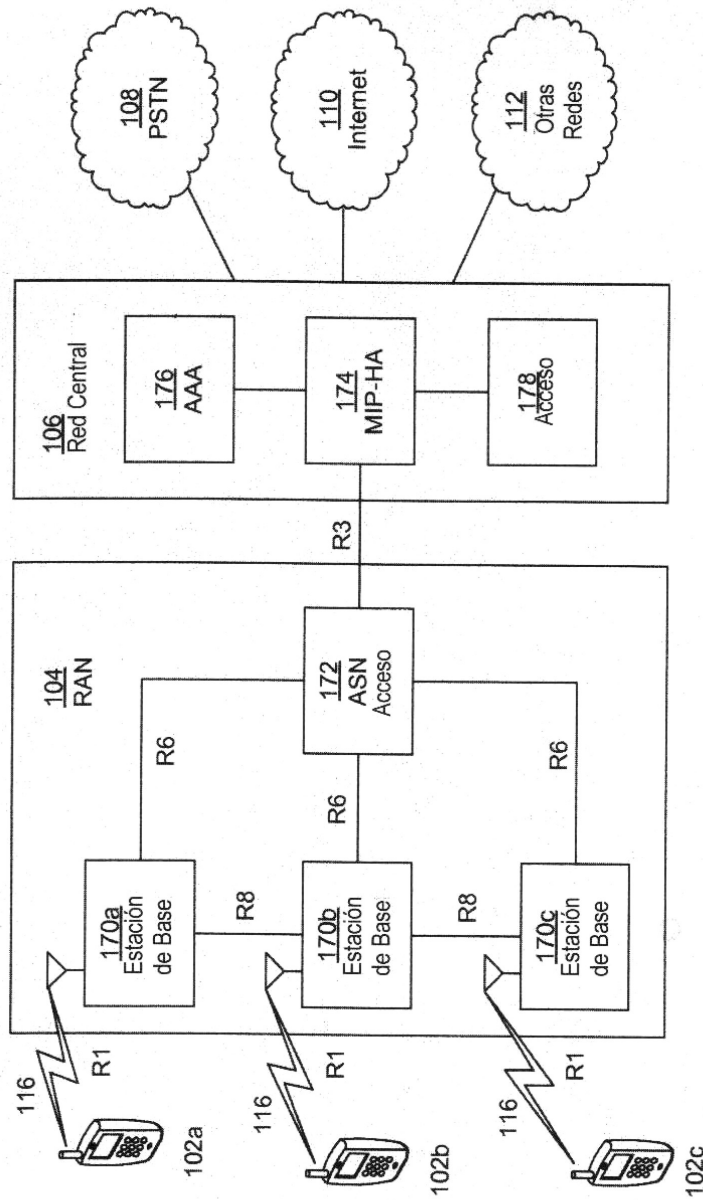


FIG. 12E (TÉCNICA ANTERIOR)