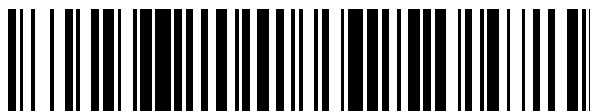


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 529 344**

51 Int. Cl.:

H03M 7/40 (2006.01)

H03M 7/42 (2006.01)

H04N 19/107 (2014.01)

H04N 19/61 (2014.01)

H04N 19/13 (2014.01)

H04N 19/91 (2014.01)

H04N 19/137 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **22.01.2003 E 03700324 (1)**

97 Fecha y número de publicación de la concesión europea: **19.11.2014 EP 1470724**

54 Título: **Codificación de coeficientes de transformación en codificadores y/o decodificadores de imagen/vídeo**

30 Prioridad:

22.01.2002 US 54610

02.04.2002 US 369500 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
19.02.2015

73 Titular/es:

NOKIA CORPORATION (100.0%)

Karakaari 7

02610 Espoo , FI

72 Inventor/es:

KARCZEWICZ, MARTA y

KURCEREN, RAGIP

74 Agente/Representante:

VALLEJO LÓPEZ, Juan Pedro

ES 2 529 344 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

Codificación de coeficientes de transformación en codificadores y/o decodificadores de imagen/vídeo

5 **Campo de la invención**

La presente invención se refiere a un método, un sistema, y un dispositivo para la codificación de vídeo digital. Más específicamente, la invención se refiere a la codificación de coeficientes de transformación cuantificados, producidos como resultado de la codificación por transformación basada en bloques en un codificador/decodificador de vídeo/imagen que usa la codificación de longitud variable (VLC).

Antecedentes de la invención

Las secuencias de vídeo digital, como las imágenes en movimiento habituales grabadas en una película, comprenden una secuencia de imágenes fijas, creándose la ilusión de movimiento al exhibir las imágenes una tras otra a una velocidad relativamente rápida, habitualmente de 15 a 30 tramas por segundo. Debido a la velocidad relativamente rápida de exhibición, las imágenes en tramas consecutivas tienden a ser bastante similares y, por lo tanto, contienen una cantidad considerable de información redundante. Por ejemplo, una escena típica puede comprender algunos elementos inmóviles, tales como un escenario de fondo, y algunas áreas móviles, que pueden adoptar distintas formas, por ejemplo, la cara de un presentador del telediario, el tráfico en movimiento, y así sucesivamente. Como alternativa, la propia cámara que graba la escena puede estar en movimiento, en cuyo caso todos los elementos de la imagen tienen el mismo tipo de movimiento. En muchos casos, esto significa que el cambio general entre una trama de vídeo y la siguiente es bastante pequeño.

Cada trama de una secuencia de vídeo digital no comprimida comprende una matriz de píxeles de imagen. Por ejemplo, en un formato de vídeo digital usado habitualmente, conocido como formato de intercambio común de un cuarto (QCIF), una trama comprende una matriz de 176 x 144 píxeles, en cuyo caso cada trama tiene 25.344 píxeles. A su vez, cada píxel está representado por un determinado número de bits que llevan información acerca del contenido de luminancia y/o color de la región de la imagen que corresponde al píxel. Habitualmente, se usa el denominado modelo de color YUV para representar el contenido de luminancia y cromaticidad de la imagen. El componente de luminancia, o Y, representa la intensidad (brillo) de la imagen, mientras que el contenido de color de la imagen está representado por dos componentes de cromaticidad o diferencia de color, denominados U y V.

Los modelos de color basados en una representación de luminancia/cromaticidad del contenido de la imagen proporcionan determinadas ventajas, en comparación con los modelos de color que se basan en una representación que implica colores primarios (es decir, rojo, verde y azul, RVA). El sistema visual humano es más sensible a las variaciones de intensidad que a las variaciones de color, y los modelos de color YUV explotan esta propiedad usando una resolución espacial menor para los componentes de cromaticidad (U, V) que para el componente de luminancia (Y). De esta manera, la cantidad de información necesaria para codificar la información de color en una imagen puede reducirse con una reducción aceptable en la calidad de imagen.

La menor resolución espacial de los componentes de cromaticidad se logra normalmente mediante submuestreo. Habitualmente, cada trama de una secuencia de vídeo se divide en los denominados "macrobloques", que comprenden información de luminancia (Y) e información de cromaticidad (U, V) asociada, que está espacialmente submuestreada. La figura 3 ilustra una manera en la que pueden formarse los macrobloques. La figura 3a muestra una trama de una secuencia de vídeo representada usando un modelo de color YUV, teniendo cada componente la misma resolución espacial. Los macrobloques se forman representando una región de 16 x 16 píxeles de imagen en la imagen original (figura 3b) como cuatro bloques de información de luminancia, comprendiendo cada bloque de luminancia una matriz de 8 x 8 valores de luminancia (Y) y dos componentes de cromaticidad (U y V) espacialmente correspondientes que se submuestran por un factor de dos en las direcciones x e y para producir las correspondientes matrices de 8 x 8 valores de cromaticidad (U, V), (véase la figura 3c). De acuerdo con determinadas recomendaciones de codificación de vídeo, tal como la recomendación H.26L de la Unión Internacional de Telecomunicaciones (ITU-T), el tamaño de bloque fundamental usado dentro de los macrobloques puede ser distinto a 8 x 8, por ejemplo, 4 x 8 o 4 x 4 (véase G. Bjontegaard, "H.26L Test Model Long Term Number 8 (TML-8) draft 0", VCEG-N10, junio de 2001, sección 2.3).

Una imagen QCIF comprende 11 x 9 macrobloques. Si los bloques de luminancia y los bloques de cromaticidad se representan con 8 bits de resolución (es decir, por números en el intervalo de 0 a 255), el número total de bits requeridos por macrobloque es $(16 \times 16 \times 8) + 2 \times (8 \times 8 \times 8) = 3.072$ bits. El número de bits necesarios para representar una trama de vídeo en el formato QCIF es, por lo tanto, $99 \times 3072 = 304.128$ bits. Esto significa que la cantidad de datos requeridos para transmitir/grabar/exhibir una secuencia de vídeo no comprimida en formato QCIF, representada usando un modelo de color YUV, a una velocidad de 30 tramas por segundo, es de más de 9 Mbps (millones de bits por segundo). Esta es una velocidad de datos extremadamente alta, y no es práctica para su uso en aplicaciones de grabación, transmisión y exhibición de vídeo, debido a la enorme capacidad de almacenamiento, capacidad del canal de transmisión y rendimiento de hardware requeridos.

Si los datos de vídeo deben transmitirse en tiempo real a través de una red de línea fija tal como una RDSI (Red Digital de Servicios Integrados) o una RTPC (Red Telefónica Pública Conmutada) convencional, el ancho de banda de transmisión de datos disponible es habitualmente del orden de 64 kbits/s. En la videotelefonía móvil, donde la transmisión tiene lugar, al menos en parte, a través de un enlace de comunicaciones por radio, el ancho de banda disponible puede ser tan bajo como 20 kbits/s. Esto significa que debe lograrse una reducción significativa en la cantidad de información usada para representar datos de vídeo con el fin de permitir la transmisión de secuencias de vídeo digital a través de redes de comunicación de bajo ancho de banda. Por esta razón, se han desarrollado técnicas de compresión de vídeo que reducen la cantidad de información transmitida, manteniendo a la vez una calidad de imagen aceptable.

Los métodos de compresión de vídeo se basan en la reducción de las partes redundantes y perceptivamente irrelevantes de las secuencias de vídeo. La redundancia en las secuencias de vídeo puede categorizarse en redundancia espacial, temporal y espectral. "Redundancia espacial" es la expresión usada para describir la correlación (similitud) entre píxeles cercanos dentro de una trama. La expresión "redundancia temporal" expresa el hecho de que es probable que los objetos que aparecen en una trama de una secuencia aparezcan en tramas subsiguientes, mientras que "redundancia espectral" se refiere a la correlación entre diferentes componentes de color de la misma imagen.

Normalmente, no puede lograrse una compresión lo suficientemente eficiente simplemente reduciendo las diversas formas de redundancia en una determinada secuencia de imágenes. Por lo tanto, la mayoría de los codificadores de vídeo actuales también reducen la calidad de aquellas partes de la secuencia de vídeo que son, subjetivamente, las menos importantes. Además, la redundancia del propio flujo de bits de vídeo comprimido se reduce por medio de una codificación eficiente sin pérdidas. En general, esto se logra usando una técnica conocida como codificación por entropía.

A menudo hay una cantidad significativa de redundancia espacial entre los píxeles que componen cada trama de una secuencia de vídeo digital. En otras palabras, el valor de cualquier píxel dentro de una trama de la secuencia es esencialmente el mismo que el valor de otros píxeles en su proximidad inmediata. Habitualmente, los sistemas de codificación de vídeo reducen la redundancia espacial usando una técnica conocida como "codificación por transformación basada en bloques", en la que se aplica una transformación matemática a los píxeles de una imagen macrobloque a macrobloque. La codificación por transformación transforma los datos de imágenes de una representación que comprende valores de píxeles a una forma que comprende un conjunto de valores de coeficientes, cada uno de los cuales es un factor de ponderación (multiplicador) para una función de base de la transformación en cuestión. Mediante el uso de determinadas transformaciones matemáticas, tales como la transformada discreta de coseno bidimensional (DCT), puede reducirse significativamente la redundancia espacial dentro de una trama de una secuencia de vídeo digital, produciendo de este modo una representación más compacta de los datos de imagen.

Las tramas de una secuencia de vídeo que se comprimen usando la codificación por transformación basada en bloques, sin referencia a ninguna otra trama dentro de la secuencia, se denominan tramas INTRA-codificadas, o tramas-I. Además, y allí donde sea posible, los bloques de tramas INTRA-codificadas se predicen a partir de los bloques codificados anteriormente dentro de la misma trama. Esta técnica, conocida como INTRA-predicción, tiene el efecto de reducir adicionalmente la cantidad de datos requeridos para representar una trama INTRA-codificada.

En general, los sistemas de codificación de vídeo no solo reducen la redundancia espacial dentro de las tramas individuales de una secuencia de vídeo, sino que también hacen uso de una técnica conocida como "predicción compensada de movimiento", para reducir la redundancia temporal en la secuencia. Usando la predicción compensada de movimiento, el contenido de imagen de algunas (a menudo muchas) tramas en una secuencia de vídeo digital se "predice" a partir de una o más tramas adicionales en la secuencia, conocidas como tramas de "referencia" o de "anclaje". La predicción del contenido de imagen se logra rastreando el movimiento de objetos o regiones de una imagen, entre una trama a codificar (comprimir) y la o las tramas de referencia, usando "vectores de movimiento". En general, la o las tramas de referencia pueden preceder a la trama a codificar, o pueden seguirla en la secuencia de vídeo. Como en el caso de la INTRA-codificación, la predicción compensada de movimiento de una trama de vídeo se realiza, habitualmente, macrobloque a macrobloque.

Las tramas de una secuencia de vídeo que se comprimen usando la predicción compensada de movimiento se denominan, en general, tramas INTER-codificadas o tramas-P. La predicción compensada de movimiento, por sí sola, rara vez proporciona una representación suficientemente precisa del contenido de imagen de una trama de vídeo y, por lo tanto, habitualmente es necesario proporcionar una denominada trama de "error de predicción" (PE) con cada trama INTER-codificada. La trama de error de predicción representa la diferencia entre una versión decodificada de la trama INTER-codificada y el contenido de imagen de la trama a codificar. Más específicamente, la trama de error de predicción comprende valores que representan la diferencia entre los valores de píxel en la trama a codificar y los correspondientes valores de píxel reconstruidos formados basándose en una versión predicha de la trama en cuestión. En consecuencia, la trama de error de predicción tiene características similares a una imagen fija, y la codificación por transformación basada en bloques puede aplicarse con el fin de reducir su redundancia espacial y, por lo tanto, la cantidad de datos (número de bits) requeridos para representarla.

Con el fin de ilustrar el funcionamiento de un sistema de codificación de vídeo con mayor detalle, a continuación se hará referencia a las figuras 1 y 2. La figura 1 es un diagrama esquemático de un codificador de vídeo genérico que emplea una combinación de INTRA-codificación e INTER-codificación para producir un flujo de bits de vídeo comprimido (codificado). En la figura 2 se ilustra, y se describirá más adelante en el texto, un decodificador correspondiente.

El codificador 100 de vídeo comprende una entrada 101 para recibir una señal de vídeo digital desde una cámara u otra fuente de vídeo (no mostrada). También comprende una unidad 104 de transformación, que está dispuesta para realizar una transformada discreta de coseno (DCT) basada en bloques, un cuantificador 106, un cuantificador 108 inverso, una unidad 110 de transformación inversa, dispuesta para realizar una transformada discreta de coseno inversa (IDCT) basada en bloques, unos combinadores 112 y 116, y un almacén 120 de tramas. El codificador comprende además un estimador 130 de movimiento, un codificador 140 de campo de movimiento y un predictor 150 de compensación de movimiento. Los conmutadores 102 y 114 se hacen funcionar conjuntamente por el gestor 160 de control para conmutar el codificador entre un modo INTRA de codificación de vídeo y un modo INTER de codificación de vídeo. El codificador 100 también comprende un codificador 170 multiplexor de vídeo que forma un único flujo de bits a partir de los diversos tipos de información producida por el codificador 100 para su transmisión posterior a un terminal de recepción remoto o, por ejemplo, para su almacenamiento en un medio de almacenamiento masivo, tal como un disco duro de ordenador (no mostrado).

El codificador 100 funciona de la siguiente manera. Cada trama de vídeo no comprimido suministrada desde la fuente de vídeo a la entrada 101 se recibe y se procesa macrobloque a macrobloque, preferentemente en orden de exploración de trama. Cuando comienza la codificación de una nueva secuencia de vídeo, la primera trama a codificar se codifica como una trama INTRA-codificada. A continuación, se programa el codificador para codificar cada trama en un formato INTER-codificado, a menos que se cumpla una de las siguientes condiciones: 1) se considera que el macrobloque actual de la trama que se está codificando es tan disímil a los valores de píxel en la trama de referencia usada en su predicción que se produce una información de error de predicción excesiva, en cuyo caso el macrobloque actual se codifica en un formato INTRA-codificado; 2) ha finalizado un intervalo predefinido de repetición de trama INTRA; o 3) se recibe retroalimentación desde un terminal receptor que indica una solicitud de una trama a proporcionar en formato INTRA-codificado.

La aparición de la condición 1) se detecta monitorizando la salida del combinador 116. El combinador 116 forma una diferencia entre el macrobloque actual de la trama que se está codificando y su predicción, producida en el bloque 150 de predicción compensada de movimiento. Si una medición de esta diferencia (por ejemplo, una suma de las diferencias absolutas de valores de píxel) supera un umbral predeterminado, el combinador 116 informa al gestor 160 de control a través de una línea 119 de control, y el gestor 160 de control hace funcionar los conmutadores 102 y 114 a través de la línea 113 de control, con el fin de conmutar el codificador 100 al modo INTRA-codificación. De esta manera, una trama que en caso contrario se codifica en formato INTER-codificado puede comprender macrobloques INTRA-codificados. La aparición de la condición 2) se monitoriza por medio de un temporizador o contador de tramas implementado en el gestor 160 de control de tal manera que, si el temporizador expira, o el contador de tramas alcanza un número predeterminado de tramas, el gestor 160 de control hace funcionar los conmutadores 102 y 114 a través de la línea 113 de control para conmutar el codificador al modo INTRA-codificación. La condición 3) se desencadena si el gestor 160 de control recibe una señal de retroalimentación desde, por ejemplo, un terminal de recepción, a través de la línea 121 de control, que indica que se requiere una actualización de INTRA-trama por el terminal de recepción. Tal condición puede surgir, por ejemplo, si una trama transmitida anteriormente se ha degenerado gravemente por las interferencias durante su transmisión, haciendo imposible decodificarla en el receptor. En esta situación, el decodificador de recepción emite una solicitud para que la siguiente trama se codifique en formato INTRA-codificado, reiniciando de este modo la secuencia de codificación.

A continuación, se describirá el funcionamiento del codificador 100 en modo INTRA-codificación. En modo INTRA-codificación, el gestor 160 de control hace funcionar el conmutador 102 para aceptar la entrada de vídeo desde la línea 118 de entrada. La entrada de la señal de vídeo se recibe macrobloque a macrobloque desde la entrada 101 a través de la línea 118 de entrada. A medida que se reciben, los bloques de valores de luminancia y cromaticidad que componen el macrobloque se pasan al bloque 104 de transformación DCT, que realiza una transformada discreta de coseno bidimensional en cada bloque de valores, produciendo una matriz bidimensional de coeficientes DCT para cada bloque. El bloque 104 de transformación DCT produce una matriz de valores de coeficientes para cada bloque, dependiendo el número de valores de coeficientes de la naturaleza de los bloques que componen el macrobloque. Por ejemplo, si el tamaño de bloque fundamental usado en el macrobloque es 4 x 4, el bloque 104 de transformación DCT produce una matriz 4 x 4 de coeficientes DCT para cada bloque. Si el tamaño del bloque es 8 x 8, se produce una matriz 8 x 8 de coeficientes DCT.

Los coeficientes DCT para cada bloque se pasan al cuantificador 106, donde se cuantifican usando un parámetro QP de cuantificación. La selección del parámetro QP de cuantificación se controla mediante el gestor 160 de control a través de la línea 115 de control. La cuantificación introduce una pérdida de información, ya que los coeficientes cuantificados tienen una menor precisión numérica que los coeficientes generados originalmente por el bloque 104 de transformación DCT. Esto proporciona un mecanismo adicional por el que puede reducirse la cantidad de datos

requeridos para representar cada imagen de la secuencia de vídeo. Sin embargo, a diferencia de la transformación DCT, que es esencialmente sin pérdidas, la pérdida de información introducida por la cuantificación provoca una degradación irreversible en la calidad de imagen. Cuanto mayor es el grado de cuantificación aplicado a los coeficientes DCT, mayor es la pérdida de calidad de imagen.

Los coeficientes DCT cuantificados para cada bloque se pasan desde el cuantificador 106 al codificador 170 multiplexor de vídeo, como se indica con la línea 125 en la figura 1. El codificador 170 multiplexor de vídeo ordena los coeficientes de transformación cuantificados para cada bloque, usando un procedimiento de escaneo en zigzag. Esta operación convierte la matriz bidimensional de coeficientes de transformación cuantificados en una matriz unidimensional. Los ordenamientos típicos de escaneo en zigzag, tal como el de una matriz 4 x 4 mostrada en la figura 4, ordenan los coeficientes aproximadamente en orden ascendente de frecuencia espacial. Esto también tiende a ordenar los coeficientes de acuerdo con sus valores, de tal manera que es más probable que los coeficientes colocados antes en la matriz unidimensional tengan mayores valores absolutos que los coeficientes colocados a continuación en la matriz. Esto se debe a que las frecuencias especiales inferiores tienden a tener mayores amplitudes dentro de los bloques de imágenes. En consecuencia, los valores que se producen hacia el final de la matriz unidimensional de coeficientes de transformación cuantificados suelen ser ceros.

Habitualmente, el codificador 170 multiplexor de vídeo representa cada coeficiente cuantificado distinto de cero en la matriz unidimensional con dos valores, denominados *nivel* y *racha*. *Nivel* es el valor del coeficiente cuantificado y *racha* es el número de coeficientes de valor cero consecutivos que preceden al coeficiente en cuestión. Los valores de *nivel* y de *racha* para un coeficiente determinado se ordenan de tal manera que el valor de *nivel* preceda al valor de *racha* asociado. Un valor de *nivel* igual a cero se usa para indicar que no hay más valores de coeficientes distintos de cero en el bloque. Este valor de *nivel* 0 se denomina símbolo EOB (fin de bloque).

Como alternativa, cada coeficiente cuantificado de valor distinto a cero en la matriz unidimensional puede representarse mediante 3 valores (*racha*, *nivel*, *último*). El significado de *nivel* y de *racha* es idéntico al del esquema anterior, mientras que *último* indica si hay cualquier otro coeficiente de valor distinto a cero en la matriz unidimensional a continuación del coeficiente actual. En consecuencia, no hay necesidad de un símbolo EOB diferente.

Los valores de *racha* y de *nivel* (*último*, si corresponde) se comprimen adicionalmente usando la codificación por entropía. La codificación por entropía es una operación sin pérdidas que aprovecha el hecho de que los símbolos dentro de un conjunto de datos a codificar tienen, en general, diferentes probabilidades de ocurrencia. Por lo tanto, en vez de usar un número fijo de bits para representar cada símbolo, se asigna un número variable de bits, de tal manera que los símbolos que tienen más probabilidades de aparecer se representan con palabras clave que tienen menos bits. Por esta razón, la codificación por entropía a menudo se denomina codificación de longitud variable (VLC). Puesto que es más probable que se produzcan determinados valores de *niveles* y de *rachas* que otros valores, las técnicas de codificación por entropía pueden usarse de manera eficaz para reducir el número de bits requeridos para representar los valores de *racha* y de *nivel*.

Una vez que se han codificado por entropía los valores de *racha* y de *nivel*, el codificador 170 multiplexor de vídeo los combina adicionalmente con información de control, también codificada por entropía usando un método de codificación de longitud variable adecuado para el tipo de información en cuestión, para formar un único flujo de bits comprimido de la información 135 de imagen codificada. Es este flujo de bits, incluyendo las palabras clave de longitud variable representativas de los pares (*racha*, *nivel*), lo que se transmite desde el codificador.

Aunque la codificación por entropía se ha descrito en relación con las operaciones realizadas por el codificador 170 multiplexor de vídeo, debe observarse que en implementaciones alternativas puede proporcionarse una unidad de codificación por entropía diferente.

También se forma una versión decodificada localmente en el codificador 100. Esto se hace pasando los coeficientes de transformación cuantificados para cada bloque, emitidos por el cuantificador 106, a través del cuantificador 108 inverso, y aplicando una transformación DCT inversa en el bloque 110 de transformación inversa. De esta manera se construye una matriz reconstruida de valores de píxel para cada bloque del macrobloque. Los datos de imágenes decodificados resultantes se introducen en el combinador 112. En modo INTRA-codificación, el conmutador 114 se configura de tal manera que la entrada al combinador 112 a través del conmutador 114 sea cero. De esta manera, la operación realizada por el combinador 112 es equivalente a pasar los datos de imágenes decodificados sin alteración.

A medida que los macrobloques subsiguientes de la trama actual se reciben y se someten a las etapas de codificación y de decodificación local descritas anteriormente en los bloques 104, 106, 108, 110 y 112, se construye una versión decodificada de la trama INTRA-codificada en el almacén 120 de tramas. Cuando el último macrobloque de la trama actual se ha INTRA-codificado y posteriormente decodificado, el almacén 120 de tramas contiene una trama completamente decodificada, disponible para su uso como una trama de referencia de predicción en la codificación de una trama de vídeo recibida posteriormente en formato INTER-codificado.

A continuación, se describirá el funcionamiento del codificador 100 en modo INTER-codificación. En modo INTER-codificación, el gestor 160 de control hace funcionar el conmutador 102 para recibir su entrada desde la línea 117, que comprende la salida del combinador 116. El combinador 116 recibe la señal de entrada de vídeo macrobloque a macrobloque desde la entrada 101. A medida que el combinador 116 recibe los bloques de valores de luminancia y cromaticidad que componen el macrobloque, forma los bloques correspondientes de información de errores de predicción. La información de errores de predicción representa la diferencia entre el bloque en cuestión y su predicción, producida en el bloque 150 de predicción compensada de movimiento. Más específicamente, la información de errores de predicción para cada bloque del macrobloque comprende una matriz bidimensional de valores, cada uno de los cuales representa la diferencia entre un valor de píxel en el bloque de información de luminancia o cromaticidad que se está decodificando y un valor de píxel decodificado obtenido al formar una predicción compensada de movimiento para el bloque, de acuerdo con el procedimiento descrito a continuación. Por lo tanto, en una situación en la que cada macrobloque comprende, por ejemplo, un montaje de 4 x 4 bloques que comprende valores de luminancia y cromaticidad, la información de errores de predicción para cada bloque del macrobloque comprende, de manera similar, una matriz 4 x 4 de valores de errores de predicción.

La información de errores de predicción para cada bloque del macrobloque se pasa al bloque 104 de transformación DCT, que realiza una transformada discreta de coseno bidimensional en cada bloque de valores de errores de predicción, para producir una matriz bidimensional de coeficientes de transformación DCT para cada bloque. El bloque 104 de transformación DCT produce una matriz de valores de coeficientes para cada bloque de errores de predicción, dependiendo el número de valores de coeficientes de la naturaleza de los bloques que compongan el macrobloque. Por ejemplo, si el tamaño de bloque fundamental usado en el macrobloque es 4 x 4, el bloque 140 de transformación DCT produce una matriz 4 x 4 de coeficientes DCT para cada bloque de errores de predicción. Si el tamaño del bloque es 8 x 8, se produce una matriz 8 x 8 de coeficientes DCT.

Los coeficientes de transformación para cada bloque de errores de predicción se pasan al cuantificador 106, donde son cuantificados usando un parámetro QP de cuantificación, de manera análoga a la descrita anteriormente en relación con el funcionamiento del codificador en modo INTRA-codificación. De nuevo, la selección del parámetro QP de cuantificación se controla por el gestor 160 de control a través de la línea 115 de control.

Los coeficientes DCT cuantificados que representan la información de errores de predicción para cada bloque del macrobloque se pasan desde el cuantificador 106 al codificador 170 multiplexor de vídeo, como se indica por la línea 125 en la figura 1. Como en el modo INTRA-codificación, el codificador 170 multiplexor de vídeo ordena los coeficientes de transformación para cada bloque de errores de predicción usando el procedimiento de escaneo en zigzag descrito anteriormente (véase la figura 4) y, a continuación, representa cada coeficiente cuantificado distinto de cero como un valor de *nivel* y de *racha*. Comprime adicionalmente los valores de *racha* y de *nivel* usando la codificación por entropía, de una manera análoga a la descrita anteriormente en relación con el modo INTRA-codificación. El codificador 170 multiplexor de vídeo también recibe una información de vector de movimiento (descrita a continuación) desde el bloque 140 de codificación de campo de movimiento a través de la línea 126, e información de control desde el gestor 160 de control. Codifica por entropía la información de vector de movimiento y la información de control, y forma un único flujo 135 de bits de la información de imágenes codificada, que comprende el vector de movimiento codificado por entropía, el error de predicción y la información de control.

Los coeficientes DCT cuantificados que representan la información de errores de predicción para cada bloque del macrobloque, también se pasan desde el cuantificador 106 al cuantificador 108 inverso. En este caso, se cuantifican de manera inversa y los bloques resultantes de los coeficientes DCT cuantificados de manera inversa se aplican al bloque 110 de transformación DCT inversa, donde se someten a la transformación DCT inversa para producir bloques localmente decodificados de valores de errores de predicción. Los bloques localmente decodificados de valores de errores de predicción se introducen, a continuación, en el combinador 112. En modo INTER-codificación, el conmutador 114 se configura de tal manera que el combinador 112 también recibe los valores de píxel predichos para cada bloque del macrobloque, generados por el bloque 150 de predicción compensada de movimiento. El combinador 112 combina cada uno de los bloques localmente decodificados de los valores de errores de predicción con un bloque correspondiente de valores de píxel predichos para producir bloques de imágenes reconstruidas y almacenarlos en el almacén 120 de tramas.

A medida que se reciben los macrobloques subsiguientes de la señal de vídeo desde la fuente de vídeo y se someten a las etapas de codificación y de decodificación descritas anteriormente en los bloques 104, 106, 108, 110, 112, se construye una versión decodificada de la trama en el almacén 120 de tramas. Cuando se ha procesado el último macrobloque de la trama, el almacén 120 de tramas contiene una trama completamente decodificada, disponible para su uso como una trama de referencia de predicción al codificar una trama de vídeo recibida posteriormente en formato INTER-codificado.

A continuación, se describirá la formación de una predicción para un macrobloque de la trama actual. Cualquier trama codificada en formato INTER-codificado requiere una trama de referencia para la predicción compensada de movimiento. Esto significa, necesariamente, que cuando se codifica una secuencia de vídeo, la primera trama a codificar, ya sea la primera trama en la secuencia, o alguna otra trama, debe codificarse en formato INTRA-codificado. Esto, a su vez, significa que cuando el codificador 100 de vídeo se conmuta a modo INTER-codificación

por el gestor 160 de control, una trama de referencia completa, formada por la decodificación local de una trama codificada anteriormente, ya está disponible en el almacén 120 de tramas del codificador. En general, la trama de referencia se forma decodificando localmente o una trama INTRA-codificada o una trama INTER-codificada.

La primera etapa en la formación de una predicción para un macrobloque de la trama actual se realiza por el bloque 130 de estimación de movimiento. El bloque 130 de estimación de movimiento recibe los bloques de valores de luminancia y cromaticidad que componen el macrobloque actual de la trama a codificar a través de la línea 128. A continuación, realiza una operación de correspondencia de bloques, con el fin de identificar una región en la trama de referencia que se corresponda sustancialmente con el macrobloque actual. Con el fin de realizar la operación de correspondencia de bloques, el bloque de estimación de movimiento accede a los datos de la trama de referencia almacenados en el almacén 120 de tramas a través de la línea 127. Más específicamente, el bloque 130 de estimación de movimiento realiza la correspondencia de bloques calculando los valores de diferencia (por ejemplo, las sumas de las diferencias absolutas) que representan la diferencia en valores de píxel entre el macrobloque bajo examen y las regiones candidatas a la mayor coincidencia de píxeles a partir de una trama de referencia almacenada en el almacén 120 de tramas. Se produce un valor de diferencia para las regiones candidatas en todos los desplazamientos posibles dentro de una región de búsqueda predefinida de la trama de referencia, y el bloque 130 de estimación de movimiento determina el valor de diferencia calculado más pequeño. El desplazamiento entre el macrobloque en la trama actual y el bloque candidato de valores de píxel en la trama de referencia que produce el valor de diferencia más pequeño define el vector de movimiento para el macrobloque en cuestión.

Una vez que el bloque 130 de estimación de movimiento ha producido un vector de movimiento para el macrobloque, entrega el vector de movimiento al bloque 140 de codificación del campo de movimiento. El bloque 140 de codificación del campo de movimiento aproxima el vector de movimiento recibido del bloque 130 de estimación de movimiento usando un modelo de movimiento que comprende un conjunto de funciones de base y coeficientes de movimiento. Más específicamente, el bloque 140 de codificación del campo de movimiento representa el vector de movimiento como un conjunto de valores de coeficientes de movimiento que, cuando se multiplican por las funciones de base, forman una aproximación del vector de movimiento. Habitualmente, se usa un modelo de movimiento traslacional que solo tiene dos coeficientes de movimiento y funciones de base, pero también pueden usarse modelos de movimiento de mayor complejidad.

Los coeficientes de movimiento se pasan desde el bloque 140 de codificación del campo de movimiento al bloque 150 de predicción compensada de movimiento. El bloque 150 de predicción compensada de movimiento también recibe la región candidata de mayor coincidencia de los valores de píxel identificados por el bloque 130 de estimación de movimiento desde el almacén 120 de tramas. Usando la representación aproximada del vector de movimiento generada por el bloque 140 de codificación del campo de movimiento y los valores de píxel de la región candidata con la mayor coincidencia de píxeles de la trama de referencia, el bloque 150 de predicción compensada de movimiento genera una matriz de valores de píxel predichos para cada bloque del macrobloque. Cada bloque de valores de píxel predichos se pasa al combinador 116, donde los valores de píxel predichos se restan de los valores de píxel reales (introducidos) en el bloque correspondiente del macrobloque actual. De esta manera, se obtiene un conjunto de bloques de errores de predicción para el macrobloque.

A continuación, se describirá el funcionamiento del decodificador 200 de vídeo, mostrado en la figura 2. El decodificador 200 comprende un decodificador 270 multiplexor de vídeo, que recibe un flujo 135 de bits de vídeo codificado desde el codificador 100 y lo desmultiplexa en sus partes constituyentes, un cuantificador 210 inverso, un transformador 220 DCT inverso, un bloque 240 de predicción compensada de movimiento, un almacén 250 de tramas, un combinador 230, un gestor 260 de control y una salida 280.

El gestor 260 de control controla el funcionamiento del decodificador 200 en respuesta a si se está decodificando una trama INTRA-codificada o una trama INTER-codificada. Una señal de control de disparo INTRA/INTER, que hace que el decodificador conmute entre los modos de decodificación, se deriva, por ejemplo, de la información del tipo de imagen proporcionada en una parte de encabezamiento de cada trama de vídeo comprimida recibida desde el codificador. La señal de control de disparo INTRA/INTER se extrae del flujo de bits de vídeo codificado por el decodificador 270 multiplexor de vídeo, y se pasa al gestor 260 de control a través de la línea 215 de control.

La decodificación de una trama INTRA-codificada se realiza macrobloque a macrobloque, decodificándose cada macrobloque esencialmente tan pronto como se recibe la información codificada relacionada con el mismo en el flujo 135 de bits de vídeo. El decodificador 270 multiplexor de vídeo separa la información codificada para los bloques del macrobloque de la posible información de control relacionada con el macrobloque en cuestión. La información codificada para cada bloque de un macrobloque INTRA-codificado comprende unas palabras clave de longitud variable que representan los valores de *nivel* y de *racha* codificados por entropía para los coeficientes DCT cuantificados distintos de cero del bloque. El decodificador 270 multiplexor de vídeo decodifica las palabras clave de longitud variable usando un método de decodificación de longitud variable, correspondiente al método de codificación usado en el codificador 100 y, por lo tanto, recupera los valores de *nivel* y de *racha*. A continuación, reconstruye la matriz de valores de coeficientes de transformación cuantificados para cada bloque del macrobloque y los pasa al cuantificador 210 inverso. Toda información de control relacionada con el macrobloque también se decodifica en el decodificador multiplexor de vídeo, usando un método de decodificación adecuado, y se pasa al

gestor 260 de control. En particular, la información relacionada con el nivel de cuantificación aplicada a los coeficientes de transformación se extrae del flujo de bits codificado por el decodificador 270 multiplexor de vídeo y se suministra al gestor 260 de control a través de la línea 217 de control. El gestor de control, a su vez, traslada esta información al cuantificador 210 inverso a través de la línea 218 de control. El cuantificador 210 inverso cuantifica de manera inversa los coeficientes DCT cuantificados para cada bloque del macrobloque de acuerdo con la información de control y, a continuación, suministra los coeficientes DCT cuantificados de manera inversa al transformador 220 DCT inverso.

El transformador 220 DCT inverso realiza una transformación DCT inversa en los coeficientes DCT cuantificados de manera inversa para cada bloque del macrobloque para formar un bloque decodificado de información de imagen que comprende valores de píxel reconstruidos. Puesto que no se usa una predicción compensada de movimiento en la codificación/decodificación de los macrobloques INTRA-codificados, el gestor 260 de control controla el combinador 230 de tal manera que evita que se use cualquier información de referencia en la decodificación del macrobloque INTRA-codificado. Los valores de píxel reconstruidos para cada bloque del macrobloque se pasan a la salida 280 de vídeo del decodificador donde, por ejemplo, pueden suministrarse a un dispositivo de visualización (no mostrado). Los valores de píxel reconstruidos para cada bloque del macrobloque también se almacenan en el almacén 250 de tramas. A medida que se decodifican y se almacenan los macrobloques subsiguientes de la trama INTRA-codificada, se monta progresivamente una trama decodificada en el almacén 250 de tramas y, por lo tanto, queda disponible para su uso como una trama de referencia para la predicción compensada de movimiento en relación con la decodificación de las tramas INTER-codificadas recibidas posteriormente.

Las tramas INTER-codificadas también se decodifican macrobloque a macrobloque, decodificándose cada macrobloque INTER-codificado esencialmente tan pronto como se recibe la información codificada relacionada con el mismo en el flujo 135 de bits. El decodificador 270 multiplexor de vídeo separa la información de errores de predicción codificada para cada bloque de un macrobloque INTER-codificado de la información de vector de movimiento codificada y la posible información de control relacionada con el macrobloque en cuestión. Como se ha explicado anteriormente, la información de errores de predicción codificada para cada bloque del macrobloque comprende unas palabras clave de longitud variable que representan los valores de *nivel* y de *racha* codificados por entropía para los coeficientes de transformación cuantificados distintos de cero del bloque de errores de predicción en cuestión. El decodificador 270 multiplexor de vídeo decodifica las palabras clave de longitud variable usando un método de decodificación de longitud variable correspondiente al método de codificación usado en el codificador 100 y, de este modo, recupera los valores de *nivel* y de *racha*. A continuación, reconstruye una matriz de valores de coeficientes de transformación cuantificados para cada bloque de errores de predicción y los pasa al cuantificador 210 inverso. La información de control relacionada con el macrobloque INTER-codificado también se decodifica en el decodificador 270 multiplexor de vídeo usando un método de decodificación adecuado, y se pasa al gestor 260 de control. La información relacionada con el nivel de cuantificación aplicado a los coeficientes de transformación de los bloques de errores de predicción se extrae del flujo de bits codificado y se suministra al gestor 260 de control a través de la línea 217 de control. El gestor de control, a su vez, traslada esta información al cuantificador 210 inverso a través de la línea 218 de control. El cuantificador 210 inverso cuantifica de manera inversa los coeficientes DCT cuantificados que representan la información de errores de predicción para cada bloque del macrobloque de acuerdo con la información de control y, a continuación, suministra los coeficientes DCT cuantificados de manera inversa al transformador 220 DCT inverso. Los coeficientes DCT cuantificados inversos que representan la información de errores de predicción para cada bloque se transforman de manera inversa, a continuación, en el transformador 220 DCT inverso, para producir una matriz de valores de errores de predicción reconstruidos para cada bloque del macrobloque.

La información de vector de movimiento codificada asociada con el macrobloque se extrae del flujo 135 de bits de vídeo codificado por el decodificador 270 multiplexor de vídeo y se decodifica. La información de vector de movimiento decodificada obtenida de este modo, se pasa a través de la línea 225 de control al bloque 240 de predicción compensada de movimiento, que reconstruye un vector de movimiento para el macrobloque usando el mismo modelo de movimiento que el usado para codificar el macrobloque INTER-codificado en el codificador 100. El vector de movimiento reconstruido aproxima el vector de movimiento determinado originalmente por el bloque 130 de estimación de movimiento del codificador. El bloque 240 de predicción compensada de movimiento del decodificador usa el vector de movimiento reconstruido para identificar la localización de una región de píxeles reconstruidos en una trama de referencia de predicción almacenada en el almacén 250 de tramas. La trama de referencia puede ser, por ejemplo, una trama INTRA-codificada decodificada anteriormente, o una trama INTER-codificada decodificada anteriormente. En cualquier caso, la región de píxeles indicada por el vector de movimiento reconstruido se usa para formar una predicción para el macrobloque en cuestión. Más específicamente, el bloque 240 de predicción compensada de movimiento forma una matriz de valores de píxel para cada bloque del macrobloque copiando los valores de píxel correspondientes de la región de píxeles identificada en la trama de referencia. La predicción, es decir, los bloques de valores de píxel derivados de la trama de referencia, se pasan desde el bloque 240 de predicción compensada de movimiento al combinador 230, donde se combinan con la información de errores de predicción decodificada. En la práctica, los valores de píxel de cada bloque predicho se añaden a los valores de errores de predicción reconstruidos correspondientes emitidos por el transformador 220 DCT inverso. De esta manera, se obtiene una matriz de valores de píxel reconstruidos para cada bloque del macrobloque. Los valores de píxel reconstruidos se pasan a la salida 280 de vídeo del decodificador y también se

almacenan en el almacén 250 de tramas. A medida que los macrobloques subsiguientes de la trama INTER-codificada se decodifican y se almacenan, una trama decodificada se monta progresivamente en el almacén 250 de tramas y, por lo tanto, queda disponible para su uso como una trama de referencia para la predicción compensada de movimiento de otras tramas INTER-codificadas.

A continuación, se examinará con mayor detalle la codificación por entropía de los valores de *racha* y de *nivel* asociados con los coeficientes de transformación cuantificados usando la técnica de codificación de longitud variable (VLC).

Habitualmente, un par de *racha-nivel* se codifica como un único símbolo. Esto se logra definiendo un mapeo entre cada posible par de *racha-nivel* y los números de código definidos. Un ejemplo de un mapeo de pares de *racha-nivel* con números de código se ilustra, como una tabla de búsqueda, en la tabla 3. Usando los mapeos en la tabla de búsqueda, se asigna un número de código a cada par (*racha, nivel*) del bloque, y los números de código resultantes se codifican por la codificación VLC. Por ejemplo, en el modelo conjunto (JM) 2.0 "Working Draft Number 2, Doc. JVT-B118, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG Pattaya", 2ª reunión del JVT, Ginebra, Suiza, 29 de enero - 1 de febrero, 2002 (norma H.26L de codificación de vídeo) se definen dos tablas de búsqueda diferentes que mapean los pares (*racha, nivel*) con números de código. La selección de la tabla que debe usarse se basa en el valor QP y el modo de codificación del bloque. Más específicamente, si el modo de codificación es INTRA y el QP es menor que 24, se usa una tabla diferente que en caso contrario.

La patente de Estados Unidos 5.640.420 desvela un codificador de longitud variable que usa dos tablas VLC. El codificador convierte la señal de entrada en coeficientes DCT, cuantifica los coeficientes y produce dos subconjuntos de un conjunto de coeficientes de transformación cuantificados. El primer subconjunto comprende unos coeficientes cuantificados, que representan unos componentes de baja frecuencia de la señal de entrada. Respectivamente, el segundo subconjunto comprende unos coeficientes cuantificados, que representan unos componentes de alta frecuencia de la señal de entrada. El primer subconjunto se convierte, además, en un primer conjunto de uno o más códigos RLC y el segundo subconjunto se convierte en un segundo conjunto de uno o más códigos RLC. El codificador codifica cada código RLC de los conjuntos primero y segundo para generar un primer y un segundo conjunto de una o más palabras clave VLC. La selección de a qué subconjunto pertenece un coeficiente cuantificado, se realiza simplemente dividiendo el proceso de escaneo en zigzag en dos partes: los coeficientes que tienen un orden de escaneo menor que un umbral N pertenecen al primer subconjunto y todos los coeficientes restantes pertenecen al segundo subconjunto. Los valores de los coeficientes no afectan a la selección de a qué subconjunto pertenece cada coeficiente. Por lo tanto, la selección de la tabla VLC tampoco depende de los valores de los coeficientes.

El documento de Detlev Marpe, Gabi Blättermann y Thomas Wiegand "Improved CABAC", VCEG-018, ITU - sector de normalización de las telecomunicaciones, GRUPO DE ESTUDIO 16, pregunta 6, grupo de expertos de codificación de vídeo (VCEG), 15ª reunión: Pattaya, Tailandia, 4-6 de diciembre 2001, describe un método de cómo reducir la cantidad de información necesaria para transmitir la información de *racha-nivel-senal*. Se basa en el truncamiento de las palabras clave que se usan para codificar el valor *racha*. El truncamiento puede realizarse cuando se garantiza que solo se necesita un determinado número de índices para codificar el valor *racha*. Como ejemplo, si el COEFF_COUNT es 9 y el bloque a codificar comprende 16 coeficientes, el valor de *racha* máxima (*racha* máxima) es entonces 7. A partir de la tabla 1 puede deducirse que el último bit del prefijo del índice 7 (1110) no necesitará transmitirse porque ninguno de los índices menores de 7 comienzan con 111. También se actualiza un nuevo valor para la *racha* máxima basándose en el valor de *racha* actual.

Este documento también desvela que los valores del número de coeficientes de los bloques cercanos se usan para diseñar modelos de contexto para aumentar la eficiencia de codificación mientras que se codifica el número del valor de coeficiente del bloque actual. Más específicamente, si A y B indican los COEFF_COUNTs del bloque a la izquierda y arriba de un bloque determinado, entonces el modelo de contexto $ctx_coeff_count(C)$ del COEFF_COUNT actual de símbolo C se define por $ctx_coeff_count(C) = ((A == 0)? 0:1) + 2 * ((B == 0)? 0:1)$.

El modelo conjunto (JM) 2.0 de la norma H.26L del JVT hace uso del denominado "esquema de codificación de longitud variable universal", al que se hace referencia como UVLC. En consecuencia, un codificador de vídeo implementado de acuerdo con H.26L JM 2.0 usa un único conjunto de palabras clave de longitud variable para codificar por entropía toda la información (elementos sintácticos) a transmitir en el flujo de bits desde el codificador. Aunque las palabras clave usadas son las mismas, se define un número predeterminado de diferentes mapeos entre símbolos de datos y palabras clave para la codificación de diferentes tipos de información. Por ejemplo, se proporcionan dos mapeos entre símbolos de datos y palabras clave para los bloques que contienen información de luminancia, dependiendo la elección del mapeo del tipo de operación de escaneo en zigzag (escaneo simple o doble) usada para ordenar los coeficientes de transformación DCT cuantificados. Para mayores detalles de los esquemas de escaneo en zigzag simple y doble definidos de acuerdo con la norma H.26L, véase "Working Draft Number 2, Doc. JVT-B118, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG Pattaya", 2ª reunión del JVT, Ginebra, Suiza, 29 de enero - 1 de febrero, 2002. También se proporcionan diferentes mapeos para la información de tipo macrobloque (MB_Type), los datos de vectores de movimiento (MVD) y la información del modo INTRA-predicción, entre otros (véase la tabla 3 a continuación).

Las palabras clave UVLC definidas en H.26L JM 2.0 pueden escribirse de la siguiente forma comprimida, mostrada en la tabla 1 a continuación, donde los términos x_n pueden tomar o el valor 0 o 1:

Tabla 1: Esquema para la generación de palabras clave UVLC de acuerdo con la norma H.26L

				1				
			0	1	x_0			
		0	0	1	x_1	x_0		
	0	0	0	1	x_2	x_1	x_0	
0	0	0	0	1	x_3	x_2	x_1	x_0

La tabla 2 presenta las 16 primeras palabras clave UVLC, generadas de acuerdo con el esquema presentado en la tabla 1, mientras que la tabla 3 muestra algunos de los diferentes mapeos entre símbolos de datos y palabras clave proporcionados de acuerdo con la norma H.26L. Para una descripción completa de los mapeos, el lector debería remitirse nuevamente a "Working Draft Number 2, Doc. JVT-B118, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG Pattaya", 2ª reunión del JVT, Ginebra, Suiza, 29 de enero - 1 de febrero, 2002.

Tabla 2: Las 16 primeras palabras clave UVLC de la norma H.26L generadas de acuerdo con el esquema presentado en la tabla 1

Índice de palabras clave	Palabras clave VLC
0	1
1	010
2	011
3	00100
4	00101
5	00110
6	00111
7	0001000
8	0001001
9	0001010
10	000010000
11	000010001
12	000010010
13	000010011
14	000010100
15	000010101
....	...

Tabla 3: ejemplo de mapeos entre símbolos de datos y palabras clave UVLC como se define en H.26L TML-5

Número de código	MB_Type		INTRA_prediction_mode		MVD	Tcoeff_chroma_AC Tcoeff_luma escaneo simple		Tcoeff_luma escaneo doble	
	INTRA	INTER	Prob0	Prob1		Nivel	Racha	Nivel	Racha
0	Intra 4 x 4	Omitir	0	0	0	EOB	-	EOB	-
1	0,0,0 ³	16 x 16	1	0	1	1	0	1	0
2	1,0,0	8 x 8	0	1	-1	-1	0	-1	0
3	2,0,0	16 x 8	0	2	2	1	1	1	1
4	3,0,0	8 x 16	1	1	-2	-1	1	-1	1
5	0,1,0	8 x 4	2	0	3	1	2	2	0
6	1,1,0	4 x 8	3	0	-3	-1	2	-2	0
7	2,1,0	4 x 4	2	1	4	2	0	1	2
8	3,1,0	Intra 4 x 4	1	2	-4	-2	0	-1	2
9	0,2,0	0,0,0 ³	0	3	5	1	3	3	0
10	1,2,0	1,0,0	0	4	-5	-1	3	-3	0
11	2,2,0	2,0,0	1	3	6	1	4	4	0
12	3,2,0	3,0,0	2	2	-6	-1	4	-4	0
13	0,0,1	0,1,0	3	1	7	1	5	5	0
14	1,0,1	1,1,0	4	0	-7	-1	5	-5	0
15	2,0,1	2,1,0	5	0	8	3	0	1	3
16	3,0,1	3,1,0	4	1	-8	-3	0	-1	3
17	0,1,1	0,2,0	3	2	9	2	1	1	4
18	1,1,1	1,2,0	2	3	-9	-2	1	-1	4
19	2,1,1	2,2,0	1	4	10	2	2	2	1
20	3,1,1	3,2,0	0	5	-10	-2	2	-2	1
21	0,2,1	0,0,1	1	5	11	1	6	3	1
22	1,2,1	1,0,1	2	4	-11	-1	6	-3	1
23	2,2,1	2,0,1	3	3	12	1	7	6	0
24	3,2,1	3,0,1	4	2	-12	-1	7	-6	0
25		0,1,1	5	1	13	1	8	7	0
26		1,1,1	5	2	-13	-1	8	-7	0
27		2,1,1	4	3	14	1	9	8	0
28		3,1,1	3	4	-14	-1	9	-8	0
29		0,2,1	2	5	15	4	0	9	0
30		1,2,1	3	5	-15	-4	0	-9	0
31		2,2,1	4	4	16	5	0	10	0
32		3,2,1	5	3	-16	-5	0	-10	0
33			5	4	17	3	1	4	1
34			4	5	-17	-3	1	-4	1
35			5	5	18	3	2	2	2
36					-18	-3	2	-2	2
37					19	2	3	2	3
38					-19	-2	3	-2	3
39					20	2	4	2	4
40					-20	-2	4	-2	4
41					21	2	5	2	5
42					-21	-2	5	-2	5
43					22	2	6	2	6
44					-22	-2	6	-2	6
45					23	2	7	2	7
46					-23	-2	7	-2	7
47					24	2	8	11	0
..					..	..	..	..	..

El esquema de codificación de longitud variable universal adoptado en la norma H.26L proporciona una serie de ventajas técnicas. En particular, el uso de un único conjunto de palabras clave VLC, que pueden construirse de acuerdo con una regla sencilla, tal como la presentada en la tabla 1, permite que las palabras clave se creen bit a bit. Esto evita la necesidad de almacenamiento de una tabla de palabras clave en el codificador y el decodificador, y reduce de este modo los requisitos de memoria tanto del codificador como del decodificador. Debe observarse, por supuesto, que en implementaciones alternativas, las tablas de palabras clave pueden generarse y almacenarse en el codificador y el decodificador. Los diferentes mapeos entre símbolos de datos y palabras clave permiten al menos

una adaptación limitada del esquema de codificación UVLC a las diferentes propiedades estadísticas de los diversos tipos de datos relacionados con la imagen y la información de control que se codifican por entropía. Los diversos mapeos entre símbolos de datos y palabras clave pueden almacenarse en el codificador y el decodificador y, por lo tanto, no hay ninguna necesidad de transmitir la información de mapeo en el flujo de bits desde el codificador al decodificador. Esto ayuda a mantener la capacidad de recuperación de los errores.

Sin embargo, el esquema de codificación por entropía UVLC sufre algunas desventajas. En particular, las palabras clave fijas y el grado limitado de adaptabilidad proporcionado por los diferentes mapeos entre símbolos de datos y palabras clave dan como resultado, inevitablemente, una compresión de datos menos que óptima. Esto es debido al hecho de que, en la codificación de imágenes, la frecuencia de ocurrencia (es decir, la probabilidad) de diferentes coeficientes de transformación y, por lo tanto, la probabilidad de diferentes pares (*racha*, *nivel*), cambia en función del contenido de las imágenes y el tipo de imagen que se está codificando. Por lo tanto, si se usa un único conjunto de palabras clave de longitud variable y se proporciona un único mapeo entre los símbolos de datos a codificar/decodificar y los VLC, no puede lograrse, en general, una eficiencia óptima de codificación.

Por esta razón, se han propuesto modificaciones del esquema básico de codificación UVLC. Más específicamente, se ha sugerido incluir más de un conjunto de palabras clave VLC. Una propuesta como esta se realizó por Gisle Bjontegaard en el documento Q. 15/SG16 "Use of Adaptive Switching Between Two VLCs for INTRA Luma Coefficients", Doc. Q15-K-30, agosto de 2000. Esta propuesta sugería el uso de un segundo conjunto de palabras clave VLC para la codificación por entropía de determinados tipos de información. De acuerdo con el documento Q15-K-30, el segundo conjunto propuesto de palabras clave VLC, denominado VLC2, se construye de acuerdo con el esquema mostrado a continuación en la tabla 4. Se usa, en particular, para codificar los valores de *racha* y de *nivel* asociados con los coeficientes de transformación cuantificados de los bloques de luminancia INTRA-codificados de 4 x 4 píxeles escaneados usando el esquema de doble escaneo en zigzag de acuerdo con la norma H.26L. La tabla 5 muestra, de forma explícita, las 16 primeras palabras clave de VLC2.

Tabla 4: Esquema para la generación de palabras clave VLC2 de acuerdo con el documento Q15-K-30 de Q.15/SG16

			1	1	0				
			1		1		0		
			1		1		1		
		0		1		X ₁		X ₀	
	0	0		1		X ₂		X ₁	X ₀
0	0	0		1		X ₃		X ₂	X ₁ X ₀

Tabla 5: Las 16 primeras palabras clave VLC2 del documento Q15-K-30 de Q.15/SG16, generadas de acuerdo con el esquema presentado en la tabla 4

Índice de palabras clave	Palabras clave VLC2
0	10
1	110
2	111
3	0100
4	0101
5	0110
6	0111
7	001000
8	001001
9	001010
10	001011
11	001100
12	001101
13	001110
14	001111
15	0010000

Como puede verse comparando las tablas 4 y 5 con las tablas 1 y 2, respectivamente, la palabra clave más corta de VLC2 tiene dos bits, frente al único bit adjudicado a la palabra clave más corta en el conjunto original de palabras clave UVLC. Debido al requisito general de decodificabilidad, en cuanto a que ninguna palabra clave VLC puede ser un prefijo de ninguna otra, este cambio tiene un efecto significativo sobre la estructura de las otras palabras clave VLC2. En particular, la asignación de dos bits a la palabra clave más corta permite que las palabras clave que tengan índices más altos se representen con menos bits. Por ejemplo, haciendo referencia a la tabla 2, puede verse que la palabra clave UVLC con índice 7 tiene 7 bits, mientras que la palabra clave VLC2 con índice 7 tiene solo 6 bits.

Los métodos de codificación VLC descritos anteriormente en los que las palabras clave se seleccionan por una conmutación basada en el contexto entre más de un conjunto de palabras clave, proporcionan una mejora en la eficiencia de compresión de datos en comparación con los métodos en los que se usa un único conjunto de palabras clave VLC. Sin embargo, en muchas aplicaciones de compresión de datos, y especialmente en aplicaciones relacionadas con la codificación de vídeo, hay un deseo continuo de mejorar la eficiencia de compresión de datos. Por lo tanto, todavía hay un considerable interés en el desarrollo de nuevos métodos de codificación de longitud variable que proporcionen una mayor adaptabilidad al tipo y las propiedades estadísticas de los símbolos de datos a codificar, que mantengan a la vez una baja complejidad computacional, bajos requisitos de memoria y una buena capacidad de recuperación de los errores. El método de acuerdo con la presente invención se ha desarrollado en este contexto.

Sumario de la invención

El método de acuerdo con la presente invención trata de abordar los problemas mencionados anteriormente relativos a mejorar la capacidad de adaptación de los métodos de codificación de longitud variable al tipo y las propiedades estadísticas de los símbolos de datos a codificar, a la vez que mantener una baja complejidad computacional y una buena capacidad de recuperación de los errores.

En términos generales, la invención se refiere a la codificación de longitud variable de los coeficientes de transformación cuantificados producidos como resultado de una codificación por transformación basada en bloques en un codificador de vídeo. Debe apreciarse, sin embargo, que la codificación de longitud variable y los métodos de decodificación de acuerdo con la invención pueden aplicarse de manera más general a otros tipos de datos a codificar. En particular, la invención se refiere a un método de codificación VLC en el que el número de coeficientes N_c de valor distinto a cero en un bloque de coeficientes de transformación cuantificados se identifica y se señala explícitamente en el flujo de bits transmitido a un decodificador correspondiente. Como se ha explicado anteriormente, la señalización explícita de N_c evita la necesidad de enviar un símbolo EOB y esto puede dar lugar a una mejora en la eficiencia de codificación. Más específicamente, la invención propone un método por el que el valor N_c que representa el número de coeficientes de valor distinto a cero en un bloque se codifica en sí mismo con VLC. Esto proporciona una mejora aún mayor en la eficiencia de codificación.

El método implica el uso de al menos dos conjuntos de palabras clave VLC para codificar con VLC el valor N_c y define reglas para la conmutación entre los conjuntos de palabras clave, que tienen en cuenta el número de coeficientes de valor distinto a cero en un bloque (o bloques) cercano del bloque cuyo valor N_c debe codificarse. Las reglas de conmutación se diseñan de manera que un flujo de bits que comprende palabras clave VLC representativas de N_c , y se forma de acuerdo con el método de la invención, pueda decodificarse sin requerir ninguna información relativa a la elección de las palabras clave que van a incluirse en el flujo de bits. En otras palabras, de acuerdo con la invención, no hay necesidad de proporcionar una indicación explícita del conjunto a partir del cual se ha elegido cada palabra clave representativa de un valor N_c . Esta propiedad mejora la eficiencia de la compresión de datos.

Los expertos en la materia apreciarán que el método de acuerdo con la invención puede aplicarse, en general, a la codificación VLC de símbolos de datos que se representan como pares (*racha*, *nivel*) o a cualquier representación equivalente del conjunto de símbolos de datos a codificar.

De acuerdo con un primer aspecto de la invención, se proporciona un método de codificación de un conjunto de símbolos de datos que comprende un determinado primer número de símbolos de datos que tienen un primer valor y un determinado segundo número de símbolos de datos que tienen valores distintos de dicho primer valor, en el que un conjunto de símbolos a codificar con longitud variable se forma a partir de dicho conjunto de símbolos de datos, y la codificación de longitud variable se aplica al conjunto de símbolos a codificar con longitud variable. De acuerdo con el método, al menos una característica de la codificación de longitud variable aplicada al conjunto de símbolos a codificar con longitud variable se adapta de acuerdo con el número de los símbolos a codificar con longitud variable y se proporciona una indicación del número de símbolos codificados de longitud variable. De manera ventajosa, la indicación es una palabra clave de longitud variable representativa del número de símbolos a codificar con longitud variable seleccionados a partir de uno de al menos dos conjuntos alternativos de palabras clave de longitud variable de acuerdo con el número de símbolos a codificar con longitud variable.

De acuerdo con un segundo aspecto de la invención, se proporciona un método de decodificación de palabras clave

de longitud variable que representa un conjunto de símbolos de datos, comprendiendo el conjunto de símbolos de datos un determinado primer número de símbolos de datos que tienen un primer valor y un determinado segundo número de símbolos de datos que tienen valores distintos de dicho primer valor, codificándose el conjunto de símbolos de datos mediante la formación de un conjunto de símbolos de datos a codificar con longitud variable a partir de dicho conjunto de símbolos de datos, formándose dichas palabras clave de longitud variable aplicando una codificación de longitud variable al conjunto de símbolos a codificar con longitud variable. De acuerdo con el método, al menos una característica de la decodificación aplicada a las palabras clave de longitud variable se adapta de acuerdo con una indicación del número de símbolos en el conjunto de símbolos que se han codificado con longitud variable. Ventajosamente, la indicación es una palabra clave de longitud variable representativa del número de símbolos que se han codificado con longitud variable seleccionada a partir de uno de al menos dos conjuntos alternativos de palabras clave de longitud variable de acuerdo con el número de símbolos que se han codificado con longitud variable.

De acuerdo con un tercer aspecto de la invención, se proporciona un codificador para codificar un conjunto de símbolos de datos que comprende un determinado primer número de símbolos de datos que tienen un primer valor y un determinado segundo número de símbolos de datos que tienen valores distintos de dicho primer valor, en el que un conjunto de símbolos a codificar con longitud variable está dispuesto para formarse a partir de dicho conjunto de símbolos de datos, y el codificador comprende medios de codificación para codificar con longitud variable dicho conjunto de símbolos a codificar con longitud variable. De acuerdo con la invención, el codificador está dispuesto para adaptar al menos una característica de la codificación de longitud variable aplicada al conjunto de símbolos a codificar con longitud variable de acuerdo con el número de los símbolos a codificar con longitud variable y está dispuesto, además, para proporcionar una indicación del número de símbolos a codificar con longitud variable. Ventajosamente, la indicación es una palabra clave de longitud variable representativa del número de símbolos a codificar con longitud variable seleccionada de uno de al menos dos conjuntos alternativos de palabras clave de longitud variable de acuerdo con el número de símbolos a codificar con longitud variable.

De acuerdo con un cuarto aspecto de la invención, se proporciona un decodificador para decodificar palabras clave de longitud variable que representan un conjunto de símbolos de datos, comprendiendo el conjunto de símbolos de datos un determinado primer número de símbolos de datos que tienen un primer valor y un determinado segundo número de símbolos de datos que tienen valores distintos de dicho primer valor, codificándose el conjunto de símbolos de datos mediante la formación de un conjunto de símbolos a codificar con longitud variable y formándose dichas palabras clave de longitud variable aplicando una codificación de longitud variable al conjunto de símbolos a codificar con longitud variable. De acuerdo con la invención, el decodificador está dispuesto para adaptar al menos una característica de la decodificación aplicada a las palabras clave de longitud variable de acuerdo con una indicación del número de símbolos en el conjunto de símbolos que se han codificado con longitud variable. Ventajosamente, la indicación es una palabra clave de longitud variable representativa del número de símbolos que se han codificado con longitud variable seleccionada a partir de uno de al menos dos conjuntos alternativos de palabras clave de longitud variable de acuerdo con el número de símbolos que se han codificado con longitud variable.

De acuerdo con un quinto aspecto de la invención, se proporciona un terminal multimedia que comprende un codificador de acuerdo con el tercer aspecto de la invención. De acuerdo con un sexto aspecto de la invención, se proporciona un terminal multimedia que comprende un decodificador de acuerdo con el cuarto aspecto de la invención.

Preferentemente, el terminal multimedia de acuerdo con los aspectos quinto y/o sexto de la invención es un terminal multimedia móvil dispuesto para comunicarse con una red de telecomunicaciones móviles por medio de una conexión de radio.

Para decirlo más exactamente, el método de codificación de acuerdo con la presente invención se caracteriza principalmente como se define por la reivindicación 1. El método de decodificación de acuerdo con la presente invención se caracteriza principalmente como se define por la reivindicación 11. El codificador de acuerdo con la presente invención se caracteriza principalmente como se define por la reivindicación 20. El decodificador de acuerdo con la presente invención se caracteriza principalmente como se define por la reivindicación 32.

Breve descripción de los dibujos

A continuación, se describirán las realizaciones de la invención a modo de ejemplo con referencia a los dibujos adjuntos, en los que:

- Figura 1 es un diagrama de bloques esquemático de un codificador de vídeo genérico de acuerdo con la técnica anterior;
- Figura 2 es un diagrama de bloques esquemático de un decodificador de vídeo genérico de acuerdo con la técnica anterior y que se corresponde con el codificador mostrado en la figura 1;
- Figura 3 ilustra la formación de un macrobloque de acuerdo con la técnica anterior;
- Figura 4 ilustra un orden de escaneo en zigzag ejemplar;

- Figura 5 muestra un bloque de imágenes actual a codificar y sus bloques de imágenes, inmediatamente cercanos, codificados previamente;
- Figura 6 es un diagrama de bloques esquemático de un codificador de vídeo de acuerdo con una realización de la invención;
- Figura 7 es un diagrama de bloques esquemático de un decodificador de vídeo de acuerdo con una realización de la invención y que se corresponde con el codificador mostrado en la figura 6; y
- Figura 8 es un diagrama de bloques esquemático de un terminal de comunicaciones multimedia en el que puede implementarse el método de acuerdo con la invención.

Descripción detallada de la invención

En la siguiente descripción detallada de la invención, se presenta una realización ejemplar del método de acuerdo con la invención. La realización ejemplar se refiere a la codificación de longitud variable de pares (*racha*, *nivel*) que representan coeficientes de transformación de valor distinto a cero, producidos como resultado de una codificación por transformación basada en bloques en un codificador de vídeo, así como su decodificación subsiguiente en un decodificador de vídeo correspondiente.

La figura 6 es un diagrama de bloques esquemático de un codificador 600 de vídeo en el que puede aplicarse el método de acuerdo con la presente realización de la invención. La estructura del codificador de vídeo mostrado en la figura 6 es esencialmente idéntica a la del codificador de vídeo de la técnica anterior mostrado en la figura 1, con las modificaciones adecuadas en aquellas partes del codificador que realizan las operaciones de codificación de longitud variable. Todas las partes del codificador de vídeo que implementan funciones y que funcionan de manera idéntica al codificador de vídeo de la técnica anterior descrito anteriormente, se identifican con idénticos números de referencia. En la presente descripción se supone que todas las operaciones de codificación de longitud variable se realizan en el codificador 670 multiplexor de vídeo. Sin embargo, debe apreciarse que, en las realizaciones alternativas de la invención, pueden proporcionarse por separado una unidad o unidades de codificación de longitud variable. También debe observarse que el método de acuerdo con la invención puede aplicarse a determinados símbolos de datos producidos por el codificador de vídeo (por ejemplo, los pares (*racha*, *nivel*) asociados con los coeficientes de transformación de valores distintos a cero) y que pueden usarse otros métodos de codificación VLC para codificar otros símbolos de datos.

A continuación, se considerará en detalle el funcionamiento del codificador 600 de vídeo. Al codificar una trama de vídeo digital, el codificador 600 funciona de manera similar a la descrita anteriormente en relación con la figura 1 para generar tramas de vídeo comprimidas INTRA-codificadas e INTER-codificadas. Como se ha explicado anteriormente en el texto, en modo INTRA-codificación, se aplica una transformada discreta de coseno (DCT) a cada bloque de datos de imagen (valores de píxel) con el fin de producir una matriz bidimensional correspondiente de valores de coeficientes de transformación. La operación DCT se realiza en el bloque 104 de transformación y los coeficientes producidos de este modo se pasan posteriormente al cuantificador 106, donde se cuantifican. En modo INTER-codificación, la transformación DCT realizada en el bloque 104 se aplica a los bloques de valores de errores de predicción. Los coeficientes de transformación producidos como resultado de esta operación también se pasan al cuantificador 106, donde también se cuantifican. Las tramas INTER-codificadas pueden contener bloques de imágenes INTRA-codificados. En algunas situaciones, la codificación por transformación no se aplica a bloques de imágenes específicos. Por ejemplo, si se usa la INTRA-predicción en modo INTRA-codificación, algunos bloques de imágenes se predicen en el codificador a partir de uno o más bloques de imágenes codificados anteriormente. En este caso, el codificador proporciona al decodificador una indicación de los bloques anteriores que deben usarse en la predicción y no emite ningún dato de coeficientes de transformación. En modo INTER-codificación, la diferencia entre la predicción para un determinado bloque y los datos de imagen del propio bloque puede ser tan pequeña que sea ventajoso, en términos de la relación de compresión de datos, no transmitir ninguna información de errores de predicción.

De acuerdo con esta realización, el método de acuerdo con la invención se aplica a aquellos bloques de imágenes que se someten a la codificación por transformación y a la posterior cuantificación de coeficientes de transformación.

Cuando el codificador 670 multiplexor de vídeo recibe un bloque (matriz bidimensional) de coeficientes de transformación cuantificados, determina el número de coeficientes con valores distintos a cero en la matriz. De acuerdo con la invención, este número, denominado N_c , se codifica con longitud variable y se transmite al decodificador en el flujo 635 de bits. Se proporcionan al menos dos conjuntos de palabras clave de longitud variable para la codificación de longitud variable del valor N_c . En una realización de la invención, estas son, por ejemplo, las palabras clave VLC definidas respectivamente en las tablas 2 y 5, descritas previamente en el texto. Más específicamente, N_c se codifica con longitud variable usando una palabra clave seleccionada a partir de uno de los al menos dos conjuntos de palabras clave de longitud variable en respuesta a los valores de los símbolos N_c codificados previamente. Ventajosamente, los símbolos N_c codificados previamente usados para asignar las palabras clave VLC se eligen a partir de los bloques que se encuentran cerca del bloque que se codifica.

La figura 5 representa un bloque de imágenes a codificar con VLC y sus bloques cercanos. Los términos N_{c_L} , N_{c_TL} , N_{c_T} y N_{c_TR} indican, respectivamente, el número de coeficientes cuantificados distintos a cero en los bloques a la izquierda, arriba-izquierda, arriba y arriba-derecha del bloque de imágenes actual a codificar.

El valor N_c para el bloque de imágenes actual se codifica con longitud variable usando una palabra clave de uno de los conjuntos disponibles de códigos VLC, elegida teniendo en cuenta el o los valores N_c ya codificados del o de los bloques inmediatamente cercanos. El siguiente código presenta un procedimiento ejemplar para asignar un código de longitud variable a un valor N_c , teniendo en cuenta los valores N_c de los bloques de imágenes cercanos codificados previamente:

$$VLC_N = N_{c_map}[fn(N_{c_L}, N_{c_TL}, N_{c_T}, N_{c_TR})] \quad (1)$$

En la expresión (1), fn es una función (un ejemplo de la cual se define a continuación en la ecuación (2)), N_{c_map} es una matriz que mapea valores enteros a diferentes códigos VLC disponibles y VLC_N indica el número de la tabla a partir de la que se seleccionan las palabras clave. Ejemplos de fn incluyen el "promedio" y la "media". En una realización de la invención, se usa el "promedio" de N_{c_L} y N_{c_T} . De acuerdo con esta realización ejemplar, la asignación presentada en la expresión (1) se convierte en:

$$VLC_N = N_{c_map}[(N_{c_L} + N_{c_T}) / 2] \quad (2)$$

El siguiente ejemplo ilustra adicionalmente la codificación de N_c de acuerdo con el método de la invención. Se asume que el valor N_c actual a codificar es 3, los valores N_{c_L} y N_{c_T} cercanos están dados por 4 y 2, respectivamente, y N_{c_map} está dado por $N_{c_map} = [1 \ 1 \ 1 \ 1 \ 1 \ 2 \ 2 \ 2 \ 2]$ donde $VLC1$ y $VLC2$ se corresponden con las palabras clave de la tabla 2 y la tabla 5, respectivamente. Aplicando la ecuación (2), VLC_N es igual a 1. En este caso $N_c = 3$ se codifica como 00100 usando las palabras clave VLC de la tabla 2.

El codificador 670 multiplexor de vídeo representa a continuación los coeficientes de valor distinto a cero como pares (*racha*, *nivel*). Esto puede hacerse de una manera conocida a partir de la técnica anterior. Más específicamente, la matriz bidimensional de coeficientes de transformación cuantificados se escanea en primer lugar usando un orden de escaneo predefinido, tal como el mostrado en la figura 4, para producir una matriz unidimensional ordenada. Cada valor de coeficiente distinto a cero en la matriz unidimensional ordenada se representa a continuación con un valor de *racha* y un valor de *nivel*, donde el valor de *nivel* representa el valor del coeficiente y el valor de *racha* representa el número de coeficientes de valor distinto a cero consecutivos que preceden al coeficiente de valor distinto a cero. A diferencia del método de la técnica de codificación de longitud de *racha* descrito anteriormente, de acuerdo con el método de la invención, no se necesita transmitir un símbolo EOB. La posibilidad de no transmitir un símbolo EOB surge porque, de acuerdo con el método de la invención, el número de coeficientes de transformación cuantificados de valor distinto a cero se indica explícitamente al decodificador. Por lo tanto, cuando los pares (*racha*, *nivel*) se decodifican para un bloque de imágenes determinado, el decodificador puede contar el número de pares decodificados. Cuando el número de pares decodificados (correspondiente al número de coeficientes de valor distinto a cero) alcanza el número de coeficientes de valor distinto a cero indicados al decodificador, puede detenerse la decodificación para el bloque en cuestión.

A continuación, el codificador 670 multiplexor de vídeo aplica la codificación de longitud variable a los valores de *racha* y de *nivel*, de tal manera que se asigna una única palabra clave VLC a cada par (*racha*, *nivel*). De acuerdo con la invención, la codificación VLC se realiza seleccionando una de un conjunto de tablas (de consulta) de codificación de longitud variable almacenadas en el codificador 600. Cada una de las tablas de codificación define un conjunto de palabras clave de longitud variable, así como un mapeo entre las palabras clave y los posibles valores de pares (*racha*, *nivel*) que puedan surgir en los datos de coeficientes de transformación cuantificados para un bloque de imágenes. De acuerdo con la invención, el codificador 670 multiplexor de vídeo selecciona una tabla de codificación para su uso de acuerdo con el número de coeficientes de valor distinto a cero en la matriz de coeficientes de transformación cuantificados para el bloque de imágenes que se codifica. Usando los mapeos definidos en la tabla de consulta seleccionada, el codificador multiplexor de vídeo asigna una palabra clave a cada par (*racha*, *nivel*) y transmite la palabra clave al decodificador en el flujo 635 de bits.

El número de tablas de codificación VLC diferentes puede variar en las diferentes realizaciones de la invención. En una realización de la invención, se proporciona una tabla de consulta distinta para cada número posible de coeficientes de transformación cuantificados de valor distinto a cero. Las palabras clave y los mapeos entre los pares (*racha*, *nivel*) y las palabras clave para cada una de las tablas se diseñan para proporcionar una óptima compresión de datos cuando se aplican a un conjunto de pares (*racha*, *nivel*) derivados de una matriz de valores de coeficientes de transformación cuantificados que tienen un número específico de coeficientes de valor distinto a cero. La información sobre la variación de las propiedades estadísticas de los pares (*racha*, *nivel*) con respecto al número de coeficientes de transformación cuantificados de valor distinto a cero puede obtenerse empíricamente, por ejemplo, codificando un número de secuencias de imágenes o de vídeo de prueba ("entrenamiento") por adelantado. Las palabras clave VLC y los mapeos entre las palabras clave pueden, a continuación, diseñarse y almacenarse como una o más tablas de consulta en el codificador.

En una realización preferida, para su uso en la codificación de pares (*racha*, *nivel*) derivados de bloques de imágenes 4 x 4, solo se proporcionan dos tablas de codificación VLC, una para su uso cuando el número de coeficientes de transformación cuantificados de valor distinto a cero es menor que o igual a 8 y una para usarse cuando el número de coeficientes de transformación cuantificados de valor distinto a cero es mayor que 8. Las tablas 6 y 7 ilustran las diferentes distribuciones de probabilidad de pares (*racha*, *nivel*) en esta realización de la invención. Las estadísticas se recogieron usando un codificador de vídeo implementado de acuerdo con el Modelo 8 de ensayo (TML8) de la recomendación ITU-T H.26L vigente y en relación con los bloques de imágenes INTER-codificados. En las tablas, se asigna el número 1 al par más probable, el número 2 al segundo par más probable, etc. Se enumeran los 15 primeros pares más probables.

Tabla 6: Probabilidades de pares (*racha*, *nivel*) cuando el número de coeficientes cuantificados de valor distinto a cero es menor que o igual a 8 (bloque INTER-codificado 4 x 4).

		racha									
		0	1	2	3	4	5	6	7	8	9
amplitud de nivel	1	1	2	3	4	6	7	8	11	12	15
	2	5	10	13							
	3	9									
	4	14									

Tabla 7: Probabilidades de pares (*racha*, *nivel*) cuando el número de coeficientes cuantificados de valor distinto a cero es mayor que 8 (bloque INTER-codificado 4 x 4).

		racha			
		0	1	2	3
amplitud de nivel	1	1	3	5	9
	2	2	7	12	
	3	4	10		
	4	6	14		
	5	8			
	6	11			
	7	13			
	8	15			

En una realización alternativa de la invención, la eficiencia de la codificación puede mejorarse cambiando el tipo de símbolos transmitidos. En lugar de representar los coeficientes de transformación cuantificados como pares (*racha*, *nivel*), se representan por unos pares (*racha*, *nivel*) y se asigna una palabra clave VLC a cada uno de los pares (*racha*, *nivel*). El *nivel* de un par (*nivel*, *último*) representa el valor del coeficiente, pero a diferencia de los pares (*racha*, *nivel*) descritos anteriormente, puede tomar el valor 0. El parámetro *último* se usa para indicar que no hay más coeficientes de valor distinto a cero en el bloque. Por ejemplo, puede usarse como un indicador binario, de tal manera que si el valor de *nivel* de un coeficiente específico es cero y todos los valores de coeficientes siguientes son cero, *último* se establece igual a 1. Esta representación alternativa de los coeficientes de transformación cuantificados puede proporcionar una mejora en la eficiencia de codificación (compresión de datos), especialmente para los bloques INTRA-codificados y en situaciones en las que se usan valores QP pequeños.

En otra realización alternativa de la invención, se usa un conjunto fijo de palabras clave VLC y solo cambia el mapeo entre las palabras clave y los pares (*racha*, *nivel*) a codificar en función del número de coeficientes de valor distinto a cero.

En una realización alternativa más de la invención, la tabla de codificación VLC elegida para un bloque de imágenes específico también puede depender de algún otro parámetro, tal como el valor del parámetro de cuantificación (QP) usado para cuantificar los coeficientes de transformación de un bloque de imágenes o el tipo de trama o el bloque de imágenes (INTRA/INTER). En este caso, la elección de una tabla de codificación VLC para una matriz específica de coeficientes de transformación cuantificados depende tanto del número de coeficientes de valor distinto a cero como del valor del otro parámetro.

En otra realización alternativa más de la invención, el número de coeficientes de transformación cuantificados de valor cero puede determinarse y usarse como el parámetro de acuerdo con el que se diseñan las palabras clave VLC y los mapeos entre las palabras clave y los símbolos de datos y se seleccionan las tablas de consulta VLC.

A continuación, se describirá el funcionamiento de un decodificador 700 de vídeo de acuerdo con una realización de la invención, con referencia a la figura 7. La estructura del decodificador de vídeo ilustrado en la figura 7 es esencialmente idéntica a la del decodificador de vídeo de la técnica anterior mostrado en la figura 2, con las modificaciones adecuadas para aquellas partes del decodificador que realizan operaciones de decodificación de longitud variable. Todas las partes del decodificador de vídeo que implementan funciones y que funcionan de manera idéntica al decodificador de vídeo de la técnica anterior descrito anteriormente se identifican con idénticos números de referencia. En la descripción siguiente se supone que todas las operaciones de decodificación de longitud variable se realizan en el decodificador 770 multiplexor de vídeo. Sin embargo, debe apreciarse que en las realizaciones alternativas de la invención pueden proporcionarse por separado una unidad o unidades de decodificación de longitud variable.

A continuación, se describirá en detalle el funcionamiento del decodificador de vídeo. En este caso, se supone que el decodificador de vídeo de la figura 7 se corresponde con el decodificador descrito en relación con la figura 6 y que, por lo tanto, es capaz de recibir y decodificar el flujo 635 de bits transmitido por el codificador 600. En el decodificador, el flujo de bits se recibe y se divide en sus partes constituyentes por el decodificador 770 multiplexor de vídeo. Como se ha explicado en relación con la descripción previa de la técnica anterior, los datos de vídeo comprimidos extraídos del flujo de bits se procesan macrobloque a macrobloque. Los datos de vídeo comprimidos para un macrobloque INTRA-codificado comprenden palabras clave de longitud variable que representan el valor N_c codificado con VLC y los pares (*racha*, *nivel*) para cada bloque del macrobloque, junto con una información de control codificada (por ejemplo, en relación con el parámetro QP de cuantificación). Los datos de vídeo comprimidos para un macrobloque INTER-codificado comprenden información de errores de predicción codificada con VLC para cada bloque (que comprende un valor N_c codificado con VLC y unos pares (*racha*, *nivel*) codificados con VLC), información del vector de movimiento para el macrobloque e información de control codificada.

El valor N_c y los pares (*racha*, *nivel*) codificados con VLC para cada bloque de un macrobloque INTRA-codificado y para cada bloque de datos de errores de predicción asociado al macrobloque INTER-codificado, se tratan de forma idéntica.

Más específicamente, cuando se decodifican los pares (*racha*, *nivel*) codificados con VLC para un bloque de imágenes INTRA o INTER-codificado, el decodificador 770 multiplexor de vídeo determina, en primer lugar, el número de coeficientes de transformación cuantificados de valor distinto a cero en el bloque. Como se ha explicado anteriormente, la información relacionada con el número de coeficientes de transformación cuantificados de valor distinto a cero, N_c , se inserta en el flujo 635 de bits por el codificador 600. El decodificador 770 extrae esta información del flujo de bits recibido. De acuerdo con la invención, el propio valor N_c se codifica con VLC y, por lo tanto, el decodificador 770 multiplexor de vídeo realiza una operación de decodificación de longitud variable adecuada para recuperar la información relativa a N_c . Como se ha explicado anteriormente, la tabla de codificación VLC usada para codificar el valor N_c para un bloque de imágenes específico se selecciona basándose en el valor o los valores asociados con un bloque o bloques de imágenes codificados previamente. Esto significa que la palabra clave representativa del valor N_c para un bloque determinado puede, en general, decodificarse sin una indicación explícita de la tabla de codificación VLC de la que se ha elegido la palabra clave. Por lo tanto, solo es necesario que el decodificador conozca la tabla de codificación VLC a partir de la que se ha seleccionado el valor N_c para el primer bloque de imágenes codificado, el orden de codificación de los bloques de imágenes y la regla usada para determinar la tabla VLC que debe usarse en la codificación de los valores N_c posteriores. El número exacto de bloques de imágenes para los que es necesario tener un conocimiento explícito de la tabla de codificación VLC seleccionada para N_c depende de la naturaleza de la regla usada para seleccionar la tabla de codificación VLC para los valores N_c posteriores y el orden de escaneo de los bloques. Por ejemplo, en el caso de que se usen valores N_c a partir de bloques a la izquierda, arriba-izquierda, arriba y arriba-derecha para seleccionar la tabla VLC para codificar valores N_c posteriores, y el orden de escaneo (codificación) sea a partir de la parte arriba-izquierda de la imagen, fila por fila, es ventajoso para codificar valores N_c para bloques en la primera fila de la imagen, más el primer bloque de la segunda fila de acuerdo con un esquema predeterminado. A continuación, los valores N_c para todos los bloques de imágenes posteriores pueden codificarse de acuerdo con la regla descrita por la ecuación (1) anterior y pueden decodificarse en el decodificador sin ninguna indicación explícita de la tabla de codificación VLC usada.

Una vez que se ha determinado el número de coeficientes de transformación cuantificados de valor distinto a cero en el bloque de imágenes, de acuerdo con el procedimiento descrito anteriormente, el decodificador 770 multiplexor de vídeo selecciona una tabla de decodificación de longitud variable y comienza a decodificar los pares (*racha*, *nivel*) codificados con VLC para el bloque de imágenes. De acuerdo con la invención, la tabla de decodificación VLC se selecciona en función de N_c . La tabla de decodificación se selecciona a partir de un conjunto de tablas de consulta idénticas a las usadas en el codificador. Las tablas de decodificación se almacenan por adelantado en el decodificador y definen el mapeo entre las palabras clave VLC recibidas en el flujo 635 de bits y los posibles valores de pares (*racha*, *nivel*) que pueden surgir en los datos de coeficientes de transformación cuantificados para un

bloque de imágenes. Después de haber seleccionado la tabla de acuerdo con el número de coeficientes de transformación cuantificados de valor distinto a cero, el decodificador 770 multiplexor de vídeo usa los mapeos definidos en la tabla de consulta seleccionada para recuperar los valores de *racha* y *nivel*. A continuación reconstruye los valores de coeficientes de transformación cuantificados para el bloque y los pasa al cuantificador 210 inverso. El resto del procedimiento de decodificación de vídeo, por el que se forman valores de pixel reconstruidos para cada bloque de imágenes, transcurre como se ha descrito en relación con el decodificador 200 de vídeo de la técnica anterior.

Debe apreciarse que el método de acuerdo con la invención proporciona ventajas significativas en comparación con los métodos de codificación VLC de la técnica anterior descritos anteriormente. En particular, se proporciona un método de codificación y de decodificación de longitud variable que es adaptable a las características estadísticas de los símbolos de datos a codificar/decodificar, mientras que mantiene una complejidad computacional relativamente baja y una alta capacidad de recuperación de los errores de transmisión. La adaptabilidad del método radica en el uso de diferentes palabras clave de longitud variable y/o mapeos para los bloques de imágenes que tienen diferentes números de coeficientes de transformación cuantificados de valor distinto a cero. La complejidad computacional se mantiene baja proporcionando las diferentes palabras clave y/o mapeos como tablas de consulta en el codificador y el decodificador. Esto también ayuda a reducir la susceptibilidad a errores de transmisión de datos, ya que la tabla de consulta específica que se usa para un bloque de imágenes determinado se señala al decodificador por medio de un valor indicativo del número de coeficientes de transformación de valor distinto a cero en el bloque.

La figura 8 presenta un dispositivo terminal que comprende un equipo de codificación y decodificación de vídeo que puede adaptarse para funcionar de acuerdo con la presente invención. Más precisamente, la figura ilustra un terminal 80 multimedia implementado de acuerdo con la recomendación ITU-T H.324. El terminal puede considerarse como un dispositivo transceptor multimedia. Incluye elementos que capturan, codifican y multiplexan flujos de datos multimedia para la transmisión a través de una red de comunicaciones, así como elementos que reciben, desmultiplexan, decodifican y muestran los contenidos multimedia recibidos. La recomendación ITU-T H.324 define el funcionamiento general del terminal y se refiere a otras recomendaciones que regulan el funcionamiento de sus diversas partes constituyentes. Esta clase de terminal multimedia puede usarse en aplicaciones en tiempo real, tales como la videotelefonía conversacional, o aplicaciones en tiempo no real, como la recuperación y/o la reproducción de video clips, por ejemplo desde un servidor de contenidos multimedia en internet.

En el contexto de la presente invención, debe apreciarse que el terminal H.324 mostrado en la figura 8 es solo una de una serie de implementaciones de terminales multimedia alternativas adecuadas para la aplicación del método de la invención. También debe tenerse en cuenta que existen una serie de alternativas en relación con la localización y la aplicación del equipo de terminal. Como se ilustra en la figura 8, el terminal multimedia puede localizarse en el equipo de comunicaciones conectado a una red telefónica de línea fija, como una RTPC (red telefónica pública conmutada) analógica. En este caso, el terminal multimedia está equipado con un módem 91, que cumple con las recomendaciones ITU-T V.8, V.34 y, opcionalmente, V.8bis. Como alternativa, el terminal multimedia puede conectarse a un módem externo. El módem permite la conversión de las señales de datos y de control digitales multiplexadas producidas por el terminal multimedia en un formato analógico adecuado para su transmisión a través de la RTPC. Además, permite que el terminal multimedia reciba las señales de datos y de control en formato analógico desde la RTPC y las convierta en un flujo de datos digital que puede desmultiplexarse y procesarse de una manera adecuada por el terminal.

Un terminal multimedia H.324 también puede implementarse de tal manera que pueda conectarse directamente a una red de línea fija digital, tal como una RDSI (red digital de servicios integrados). En este caso, el módem 91 se sustituye por una interfaz usuario-red de la RDSI. En la figura 8, esta interfaz usuario-red de la RDSI se representa por el bloque 92 alternativo.

Los terminales multimedia H.324 también pueden adaptarse para su uso en aplicaciones de comunicaciones móviles. Si se usa con un enlace de comunicación inalámbrica, el módem 91 puede sustituirse por cualquier interfaz inalámbrica adecuada, como se representa por el bloque 93 alternativo en la figura 8. Por ejemplo, un terminal multimedia H.324/M puede incluir un transceptor de radio que permite la conexión a la red de telefonía móvil GSM de segunda generación actual, o a la UMTS de tercera generación propuesta (sistema de telefonía móvil universal).

Hay que señalar que en los terminales multimedia diseñados para la comunicación bidireccional, es decir, para la transmisión y la recepción de datos de vídeo, es ventajoso proporcionar tanto un codificador de vídeo como un decodificador de vídeo implementados de acuerdo con la presente invención. Dicho par de codificador y decodificador a menudo se implementa como una única unidad funcional combinada, denominada "códec".

A continuación, se describirá con más detalle un terminal multimedia H.324 típico con referencia a la figura 8.

El terminal 80 multimedia incluye diversos elementos denominados "equipo de terminal". Este incluye dispositivos de vídeo, audio y telemáticos, indicados genéricamente por los números de referencia 81, 82 y 83, respectivamente. El equipo 81 de vídeo puede incluir, por ejemplo, una cámara de vídeo para capturar imágenes de vídeo, un monitor

para visualizar los contenidos de vídeo recibidos y un equipo de procesamiento de vídeo opcional. Habitualmente, el equipo 82 de audio incluye un micrófono para capturar, por ejemplo, mensajes hablados, y un altavoz para reproducir los contenidos de audio recibidos. El equipo de audio también puede incluir unidades adicionales de procesamiento de audio. El equipo 83 telemático, puede incluir un terminal de datos, un teclado, una pizarra electrónica o un transceptor de imágenes fijas, tal como una unidad de fax.

El equipo 81 de vídeo está acoplado a un códec 85 de vídeo. El códec 85 de vídeo comprende un codificador de vídeo y un decodificador de vídeo correspondiente, ambos implementados de acuerdo con la invención. A continuación, se describirá el codificador y el decodificador mencionados. El códec 85 de vídeo es responsable de codificar los datos de vídeo capturados de una forma adecuada para su posterior transmisión a través de un enlace de comunicaciones y de decodificar los contenidos de vídeo comprimidos recibidos desde la red de comunicaciones. En el ejemplo ilustrado en la figura 8, el códec de vídeo se implementa de acuerdo con la recomendación ITU-T H.26L, con las modificaciones apropiadas para implementar el método de codificación de longitud variable adaptativo de acuerdo con la invención tanto en el codificador como en el decodificador del códec de vídeo.

El equipo de audio del terminal está acoplado a un códec de audio, indicado en la figura 8 por el número de referencia 86. Como el códec de vídeo, el códec de audio comprende un par de codificador/decodificador. Convierte los datos de audio capturados por el equipo de audio del terminal en una forma adecuada para la transmisión a través del enlace de comunicaciones y transforma los datos de audio codificados recibidos desde la red de nuevo en una forma adecuada para la reproducción, por ejemplo, en el altavoz del terminal. La salida del códec de audio se pasa a un bloque 87 de retardo. Esto compensa los retardos introducidos por el proceso de codificación de vídeo y, por lo tanto, garantiza la sincronización de los contenidos de audio y de vídeo.

El bloque 84 de control del sistema del terminal multimedia controla la señalización de extremo a extremo de la red usando un protocolo de control apropiado (bloque 88 de señalización) para establecer un modo de funcionamiento común entre un terminal de transmisión y un terminal de recepción. El bloque 88 de señalización intercambia información acerca de las capacidades de codificación y de decodificación de los terminales de transmisión y de recepción y puede usarse para permitir los diversos modos de codificación del codificador de vídeo. El bloque 84 de control del sistema también controla el uso del cifrado de datos. La información con respecto al tipo de cifrado que debe usarse en la transmisión de datos se pasa desde el bloque 89 de cifrado al multiplexor/demultiplexor 90 (unidad MUX/DMUX).

Durante la transmisión de datos desde el terminal multimedia, la unidad 90 MUX/DMUX combina los flujos de vídeo y de audio codificados y sincronizados con la entrada de datos desde el equipo 83 telemático y los posibles datos de control, para formar un único flujo de bits. La información sobre el tipo de cifrado de datos (si existe) que debe aplicarse a la corriente de bits, proporcionada por el bloque 89 de cifrado, se usa para seleccionar un modo de cifrado. En consecuencia, cuando se recibe un flujo de bits multimedia multiplexado y posiblemente cifrado, la unidad 90 MUX/DMUX es responsable de descifrar el flujo de bits, dividirlo en sus componentes multimedia constituyentes y pasar estos componentes al códec o los códec adecuados y/o al equipo del terminal para la decodificación y la reproducción.

Hay que señalar que los elementos funcionales del terminal multimedia, el codificador/decodificador de vídeo y el códec de vídeo de acuerdo con la invención pueden implementarse como un software o un hardware dedicado, o una combinación de ambos. Los métodos de codificación y decodificación de longitud variable de acuerdo con la invención son especialmente adecuados para la implementación en forma de un programa de ordenador que comprende instrucciones legibles por máquina para realizar las etapas funcionales de la invención. Como tal, el codificador y el decodificador de longitud variable de acuerdo con la invención pueden implementarse como un código de software almacenado en un medio de almacenamiento y ejecutado en un ordenador, tal como un ordenador de sobremesa personal.

Si el terminal 80 multimedia es un terminal móvil, es decir, si está equipado con un transceptor 93 de radio, se entenderá por los expertos en la materia que también puede comprender elementos adicionales. En una realización comprende una interfaz de usuario que tiene una pantalla y un teclado, que permite el funcionamiento del terminal 80 multimedia por un usuario, una unidad de procesamiento central, tal como un microprocesador, que controla los bloques responsables de diferentes funciones del terminal multimedia, una memoria de acceso aleatorio RAM, una memoria de solo lectura ROM, y una cámara digital. Las instrucciones de funcionamiento del microprocesador, que es un código de programa correspondiente a las funciones básicas del terminal 80 multimedia, se almacenan en la memoria de solo lectura ROM y pueden ejecutarse, según sea necesario, por el microprocesador, por ejemplo bajo control del usuario. De acuerdo con el código del programa, el microprocesador usa el transceptor 93 de radio para formar una conexión con una red de comunicaciones móviles, permitiendo que el terminal 80 multimedia transmita información a y reciba información de la red de comunicaciones móviles a través de un trayecto radioeléctrico.

El microprocesador monitoriza el estado de la interfaz de usuario y controla la cámara digital. En respuesta a una orden de usuario, el microprocesador da a la cámara la instrucción de grabar imágenes digitales en la memoria RAM. Una vez que se captura una imagen o, como alternativa, durante el proceso de captura, el microprocesador segmenta la imagen en segmentos de imagen (por ejemplo, macrobloques) y usa el codificador para realizar la

- 5 codificación de movimiento compensado de los segmentos con el fin de generar una secuencia de imagen comprimida, como se explica en la descripción anterior. Un usuario puede ordenar que el terminal 80 multimedia muestre las imágenes capturadas en su pantalla o que envíe la secuencia de imágenes comprimida usando el transceptor 93 de radio a otro terminal multimedia, un videoteléfono conectado a una red de línea fija (RTPC) o algún otro dispositivo de telecomunicaciones. En una realización preferida, la transmisión de datos de imagen se inicia tan pronto como el primer segmento está codificado, de tal manera que el destinatario puede iniciar un proceso de decodificación correspondiente con un mínimo retardo.

REIVINDICACIONES

1. Un método de codificación de símbolos de datos de un bloque de imágenes que comprende símbolos de datos que tienen un primer valor y símbolos de datos que tienen valores distintos de dicho primer valor, y aplicándose a dichos símbolos de datos una codificación de longitud variable, **caracterizado por que** el método comprende además:
 - determinar un promedio de al menos la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en un bloque de imágenes codificado previamente y la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en otro bloque de imágenes codificado previamente;
 - aplicar la codificación de longitud variable a dichos símbolos de datos de dicho bloque de imágenes;
 - determinar la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en dicho bloque de imágenes;
 - codificar una indicación de dicha cantidad, que comprende
 - proporcionar al menos dos conjuntos alternativos de palabras clave de longitud variable;
 - usar dicho promedio para seleccionar un conjunto de palabras clave de longitud variable de dichos al menos dos conjuntos alternativos de palabras clave de longitud variable; y
 - usar dicha cantidad como un índice del conjunto seleccionado de palabras clave de longitud variable para seleccionar una palabra clave de longitud variable a partir de dicho conjunto seleccionado de palabras clave de longitud variable para representar a la indicación; y
 - adaptar la codificación de longitud variable de dichos símbolos de datos basándose en dicha cantidad.
2. Un método de acuerdo con la reivindicación 1, **caracterizado por que** dicha indicación de dicha cantidad se transmite en un flujo de bits a un dispositivo de decodificación.
3. Un método de acuerdo con cualquiera de las reivindicaciones 1 o 2, **caracterizado por que** comprende la representación de cada uno de dichos símbolos de datos que tienen valores distintos de dicho primer valor mediante un par de valores que comprenden un primer valor indicativo del valor del símbolo de datos y un segundo valor indicativo de un número de símbolos de datos de valor cero que preceden o que siguen a dicho símbolo de datos, formando de este modo un conjunto de pares de valores primero y segundo que representan dichos símbolos a codificar con longitud variable.
4. Un método de acuerdo con cualquiera de las reivindicaciones 1 a 3, **caracterizado por que** comprende realizar una operación de mapeo en la que cada uno de dichos símbolos a codificar con longitud variable se asigna a una palabra clave, formando de este modo un conjunto de valores codificados representativos de dichos símbolos a codificar con longitud variable, seleccionándose la palabra clave a partir de un conjunto de palabras clave, habiéndose diseñado dicho conjunto de palabras clave para proporcionar una reducción en la cantidad de información requerida para representar dichos símbolos a codificar con longitud variable cuando se usan en dicha operación de mapeo.
5. Un método de acuerdo con la reivindicación 4, **caracterizado por que** dicha operación de mapeo depende de dicha cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en dicho bloque de imágenes.
6. Un método de acuerdo con la reivindicación 5, **caracterizado por que** dicha operación de mapeo se realiza seleccionando una tabla de un conjunto de tablas de asignación seleccionables, en donde cada tabla de dicho conjunto de tablas de asignación seleccionables define una asignación de símbolos a codificar con longitud variable a palabras clave, y asignando cada uno de dichos símbolos a codificar con longitud variable a una palabra clave de acuerdo con la asignación definida por dicha tabla de asignación seleccionada.
7. Un método de acuerdo con cualquiera de las reivindicaciones 1 a 6, **caracterizado por que** dichos símbolos de datos que tienen un primer valor son símbolos de datos de valor cero y dichos símbolos de datos que tienen valores distintos de dicho primer valor son símbolos de datos de valor distinto a cero.
8. Un método de acuerdo con cualquiera de las reivindicaciones 1 a 7, **caracterizado por que** dichos símbolos de datos son valores de coeficientes de transformación cuantificados.
9. Un método de decodificación de palabras clave de longitud variable que representan símbolos de datos de un bloque de imágenes, comprendiendo los símbolos de datos unos símbolos de datos que tienen un primer valor y unos símbolos de datos que tienen valores distintos de dicho primer valor, y habiéndose formado dichas palabras clave de longitud variable mediante la aplicación de una codificación de longitud variable a dichos símbolos de datos, **caracterizado por que** el método comprende además:
 - recibir una palabra clave de longitud variable que representa la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor;

- determinar un promedio de, al menos, la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en un bloque de imágenes decodificado previamente y la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en otro bloque de imágenes decodificado previamente;
 - proporcionar al menos dos conjuntos alternativos de palabras clave de longitud variable;
 - usar dicho promedio para seleccionar un conjunto de palabras clave de longitud variable de dichos al menos dos conjuntos alternativos de palabras clave de longitud variable;
 - usar el conjunto seleccionado de palabras clave de longitud variable para decodificar dicha palabra clave de longitud variable recibida que representa dicha cantidad para obtener dicha cantidad; y
 - adaptar al menos una característica de la decodificación aplicada a las palabras clave de longitud variable que representan dichos símbolos de datos de acuerdo con dicha cantidad.
10. Un método de acuerdo con la reivindicación 9, **caracterizado por que** dicha palabra clave codificada de longitud variable que representa dicha cantidad se recibe desde un dispositivo de codificación.
11. Un método de acuerdo con las reivindicaciones 9 o 10, **caracterizado por que** comprende decodificar la palabra clave de longitud variable representativa de dicha cantidad realizando una operación de mapeo inversa que define una asignación de palabras clave a los valores.
12. Un método de acuerdo con la reivindicación 11, **caracterizado por que** dicha operación de mapeo inversa se realiza por medio de una tabla de mapeo que es una de un conjunto de tablas de mapeo y por que dicha tabla de mapeo se selecciona basándose en la cantidad de al menos un conjunto de símbolos de datos codificado previamente.
13. Un método de acuerdo con cualquiera de las reivindicaciones 9 a 12, **caracterizado por que** cada uno de dicho número de símbolos que se han codificado con longitud variable se representa por un par de valores que comprenden un primer valor indicativo del valor de un símbolo de datos y un segundo valor indicativo de un número de símbolos de datos de valor cero que preceden o que siguen a dicho símbolo de datos, representando dichos pares de valores primero y segundo dichos símbolos que se han codificado con longitud variable.
14. Un método de acuerdo con cualquiera de las reivindicaciones 9 a 13, **caracterizado por que** comprende realizar una operación de mapeo inversa en la que cada palabra clave de longitud variable se usa para identificar un símbolo específico de dicho conjunto de símbolos que se han codificado con longitud variable.
15. Un método de acuerdo con la reivindicación 14, **caracterizado por que** dicha operación de mapeo inversa depende de dicha indicación de la cantidad.
16. Un método de acuerdo con la reivindicación 15, **caracterizado por que** dicha operación de mapeo inversa se realiza seleccionando una tabla de un conjunto de tablas de asignación seleccionables, en el que cada tabla de dicho conjunto de tablas de asignación seleccionables define una asignación de símbolos a codificar a palabras clave, e identificando un símbolo específico de dichos símbolos que se han codificado con longitud variable de acuerdo con la asignación definida por dicha tabla de asignación seleccionada.
17. Un método de acuerdo con cualquiera de las reivindicaciones 9 a 16, **caracterizado por que** dichos símbolos de datos que tienen un primer valor son símbolos de datos de valor cero y dichos símbolos de datos que tienen un segundo valor son símbolos de datos de valor distinto a cero.
18. Un codificador para codificar símbolos de datos de un bloque de imágenes que comprende símbolos de datos que tienen un primer valor y símbolos de datos que tienen valores distintos de dicho primer valor, y comprendiendo el codificador medios de codificación para la codificación de longitud variable de dichos símbolos de datos, **caracterizado por que** el codificador está dispuesto para
- determinar un promedio de, al menos, la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en un bloque de imágenes codificado previamente y la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en otro bloque de imágenes codificado previamente;
 - determinar la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en dicho bloque de imágenes;
 - codificar una indicación de dicha cantidad
 - proporcionando al menos dos conjuntos alternativos de palabras clave de longitud variable;
 - usando dicho promedio para seleccionar un conjunto de palabras clave de longitud variable de dichos al menos dos conjuntos alternativos de palabras clave de longitud variable; y
 - usando dicha cantidad como un índice del conjunto seleccionado de palabras clave de longitud variable para seleccionar una palabra clave de longitud variable a partir de dicho conjunto seleccionado de palabras clave de longitud variable para representar la indicación; y
 - adaptar la codificación de longitud variable de dichos símbolos de datos basándose en dicha cantidad.

19. Un codificador de acuerdo con la reivindicación 18, **caracterizado por que** está dispuesto para formar un flujo de bits que comprende dicha indicación de la cantidad.

5 20. Un codificador de acuerdo con las reivindicaciones 18 o 19, **caracterizado por que** está dispuesto para transmitir dicho flujo de bits a un dispositivo de decodificación.

10 21. Un codificador de acuerdo con cualquiera de las reivindicaciones 18 a 20, **caracterizado por que** está dispuesto para seleccionar una palabra clave de longitud variable para representar dicha cantidad realizando una operación de mapeo que define una asignación de valores a palabras clave.

15 22. Un codificador de acuerdo con la reivindicación 21, **caracterizado por que** está dispuesto para realizar dicha operación de mapeo por medio de una tabla de mapeo que es una de un conjunto de tablas de mapeo y por que está dispuesto para seleccionar dicha tabla de mapeo basándose en la cantidad de al menos un conjunto de símbolos de datos codificado previamente.

20 23. Un codificador de acuerdo con cualquiera de las reivindicaciones 18 a 22, **caracterizado por que** está dispuesto para representar cada uno de dichos segundos números de símbolos de datos por un par de valores que comprenden un primer valor indicativo del valor del símbolo de datos y un segundo valor indicativo de un número de símbolos de datos de valor cero que preceden o que siguen a dicho símbolo de datos, formando de este modo un conjunto de pares de valores primero y segundo que representan dichos símbolos a codificar con longitud variable.

25 24. Un codificador de acuerdo con cualquiera de las reivindicaciones 18 a 23, **caracterizado por que** está dispuesto para realizar una operación de mapeo en la que cada uno de dichos símbolos a codificar con longitud variable se asigna a una palabra clave, formando de este modo un conjunto de valores codificados representativos de dichos símbolos a codificar con longitud variable, seleccionándose la palabra clave a partir de un conjunto de palabras clave, habiéndose diseñado dicho conjunto de palabras clave para proporcionar una reducción en la cantidad de información requerida para representar dichos símbolos a codificar con longitud variable cuando se usan en dicha operación de mapeo.

30 25. Un codificador de acuerdo con la reivindicación 24, **caracterizado por que** dicha operación de mapeo está dispuesta para depender de dicha cantidad.

35 26. Un codificador de acuerdo con la reivindicación 25, **caracterizado por que** dicha operación de mapeo está dispuesta para realizarse seleccionando una de un conjunto de tablas de asignación seleccionables, en el que cada tabla de dicho conjunto de tablas de asignación seleccionables define una asignación de símbolos a codificar con longitud variable a palabras clave, y asignándose cada uno de los símbolos de datos que tienen valores distintos de dicho primer valor a una palabra clave de acuerdo con la asignación definida por dicha tabla de asignación seleccionada.

40 27. Un codificador de acuerdo con cualquiera de las reivindicaciones 18 a 26, **caracterizado por que** dichos símbolos de datos que tienen un primer valor son símbolos de datos de valor cero y dichos símbolos de datos que tienen un segundo valor son símbolos de datos de valor distinto a cero.

45 28. Un codificador de acuerdo con cualquiera de las reivindicaciones 18 a 27, proporcionado en un codificador de vídeo.

50 29. Un codificador de acuerdo con cualquiera de las reivindicaciones 18 a 28, implementado como un código ejecutable por máquina almacenado en un medio de almacenamiento legible por ordenador.

55 30. Un decodificador para decodificar palabras clave de longitud variable que representan símbolos de datos de un bloque de imágenes, comprendiendo dichos símbolos de datos unos símbolos de datos que tienen un primer valor y unos símbolos de datos que tienen valores distintos de dicho primer valor, y formándose dichas palabras clave de longitud variable mediante la aplicación de una codificación de longitud variable a dichos símbolos de datos, **caracterizado por que** el decodificador está dispuesto para

- recibir una palabra clave de longitud variable que representa la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor;
- determinar un promedio de, al menos, la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en un bloque de imágenes decodificado previamente y la cantidad de símbolos de datos que tienen valores distintos de dicho primer valor en otro bloque de imágenes decodificado previamente;
- usar dicho promedio para seleccionar un conjunto de palabras clave de longitud variable de al menos dos conjuntos alternativos de palabras clave de longitud variable;
- usar el conjunto seleccionado de palabras clave de longitud variable para decodificar dicha palabra clave de longitud variable recibida que representa dicha cantidad para obtener dicho segundo número; y
- adaptar al menos una característica de la decodificación aplicada a las palabras clave de longitud variable que

representan dichos símbolos de datos de acuerdo con dicha cantidad.

- 5 31. Un decodificador de acuerdo con la reivindicación 30, **caracterizado por que** dicha palabra clave codificada de longitud variable que representa dicha cantidad se recibe desde un dispositivo de codificación.
32. Un decodificador de acuerdo con las reivindicaciones 30 o 31, **caracterizado por que** está dispuesto para decodificar la palabra clave de longitud variable representativa de dicha cantidad realizando una operación de mapeo inversa que define una asignación de palabras clave a los valores.
- 10 33. Un decodificador de acuerdo con la reivindicación 32, **caracterizado por que** está dispuesto para realizar dicha operación de mapeo inversa por medio de una tabla de mapeo que es una de un conjunto de tablas de mapeo y por que está dispuesto para seleccionar dicha tabla de mapeo basándose en el número de símbolos codificados con longitud variable en al menos un conjunto de símbolos de datos codificado previamente.
- 15 34. Un decodificador de acuerdo con cualquiera de las reivindicaciones 30 a 33, **caracterizado por que** está dispuesto para decodificar palabras clave de longitud variable que representan símbolos de datos que incluyen un par de valores, que comprenden un primer valor indicativo del valor de un símbolo de datos y un segundo valor indicativo de un número de símbolos de datos de valor cero que preceden o que siguen a dicho símbolo de datos.
- 20 35. Un decodificador de acuerdo con cualquiera de las reivindicaciones 30 a 34, **caracterizado por que** está dispuesto para decodificar dichas palabras clave de longitud variable realizando una operación de mapeo inversa en la que cada palabra clave de longitud variable se usa para identificar un símbolo específico de dicho conjunto de símbolos que se han codificado con longitud variable.
- 25 36. Un decodificador de acuerdo con la reivindicación 35, **caracterizado por que** está dispuesto para realizar dicha operación de mapeo inversa en función de dicha indicación de la cantidad.
- 30 37. Un decodificador de acuerdo con la reivindicación 36, **caracterizado por que** está dispuesto para realizar dicha operación de mapeo inversa seleccionando una tabla de un conjunto de tablas de asignación seleccionables, en el que cada tabla de dicho conjunto de tablas de asignación seleccionables define una asignación de símbolos que se han de codificar con longitud variable a palabras clave, e identificando un símbolo específico de dichos símbolos que se han codificado con longitud variable de acuerdo con la asignación definida por dicha tabla de asignación seleccionada.
- 35 38. Un decodificador de acuerdo con cualquiera de las reivindicaciones 30 a 37, **caracterizado por que** dichos símbolos de datos que tienen un primer valor son símbolos de datos de valor cero y dichos símbolos de datos que tienen un segundo valor son símbolos de datos de valor distinto a cero.
- 40 39. Un decodificador de acuerdo con cualquiera de las reivindicaciones 30 a 38, proporcionado en un decodificador de vídeo.
- 40 40. Un decodificador de acuerdo con cualquiera de las reivindicaciones 30 a 38, implementado como un código ejecutable por máquina almacenado en un medio de almacenamiento legible por ordenador.
- 45 41. Un terminal multimedia que comprende un codificador de acuerdo con cualquiera de las reivindicaciones 18 a 29.
42. Un terminal multimedia que comprende un decodificador de acuerdo con cualquiera de las reivindicaciones 30 a 40.
- 50 43. Terminal multimedia de acuerdo con las reivindicaciones 41 o 42, **caracterizado por que** es un terminal multimedia móvil dispuesto para comunicar con una red de telecomunicaciones móviles por medio de una conexión de radio.

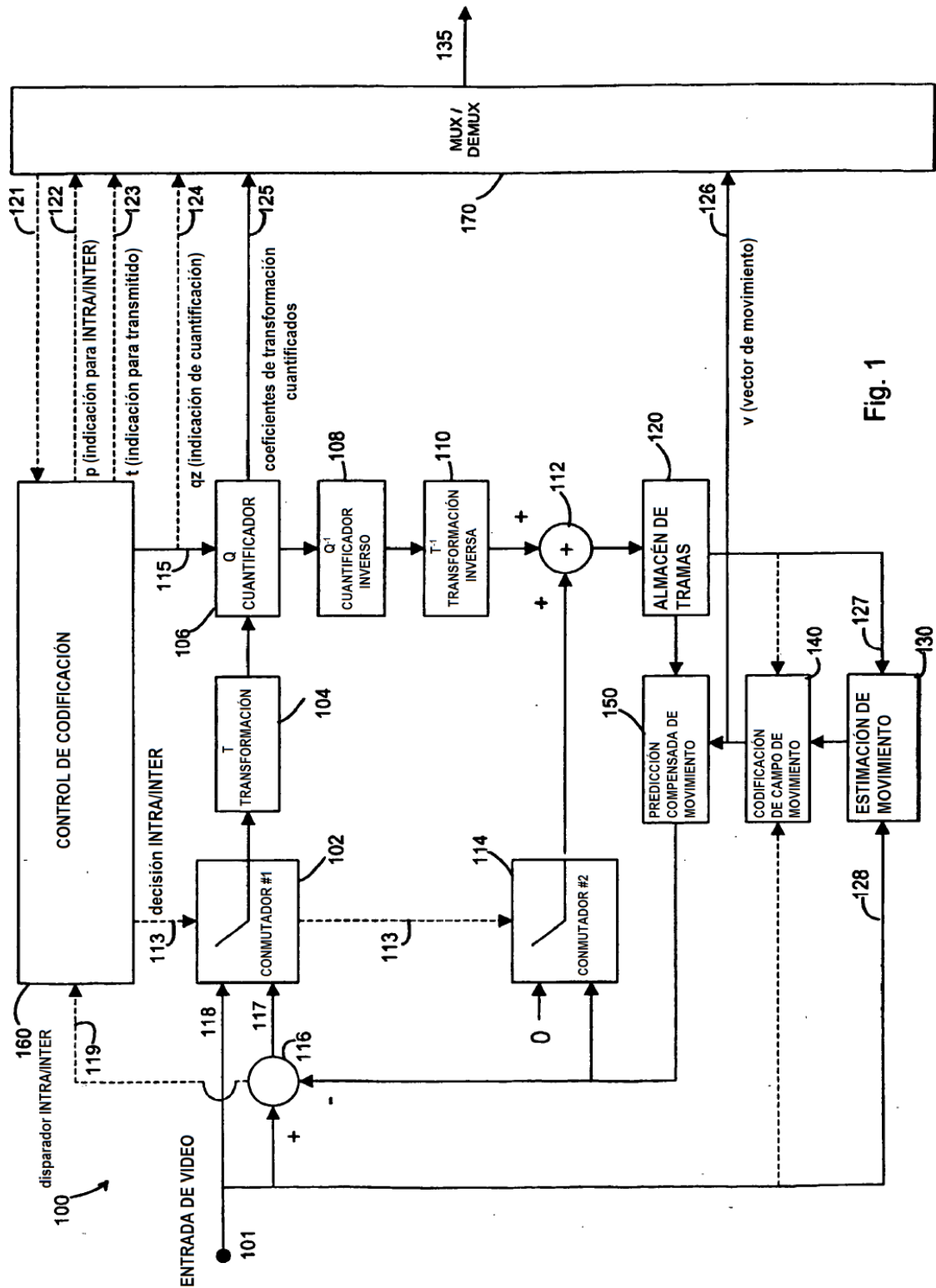


Fig. 1

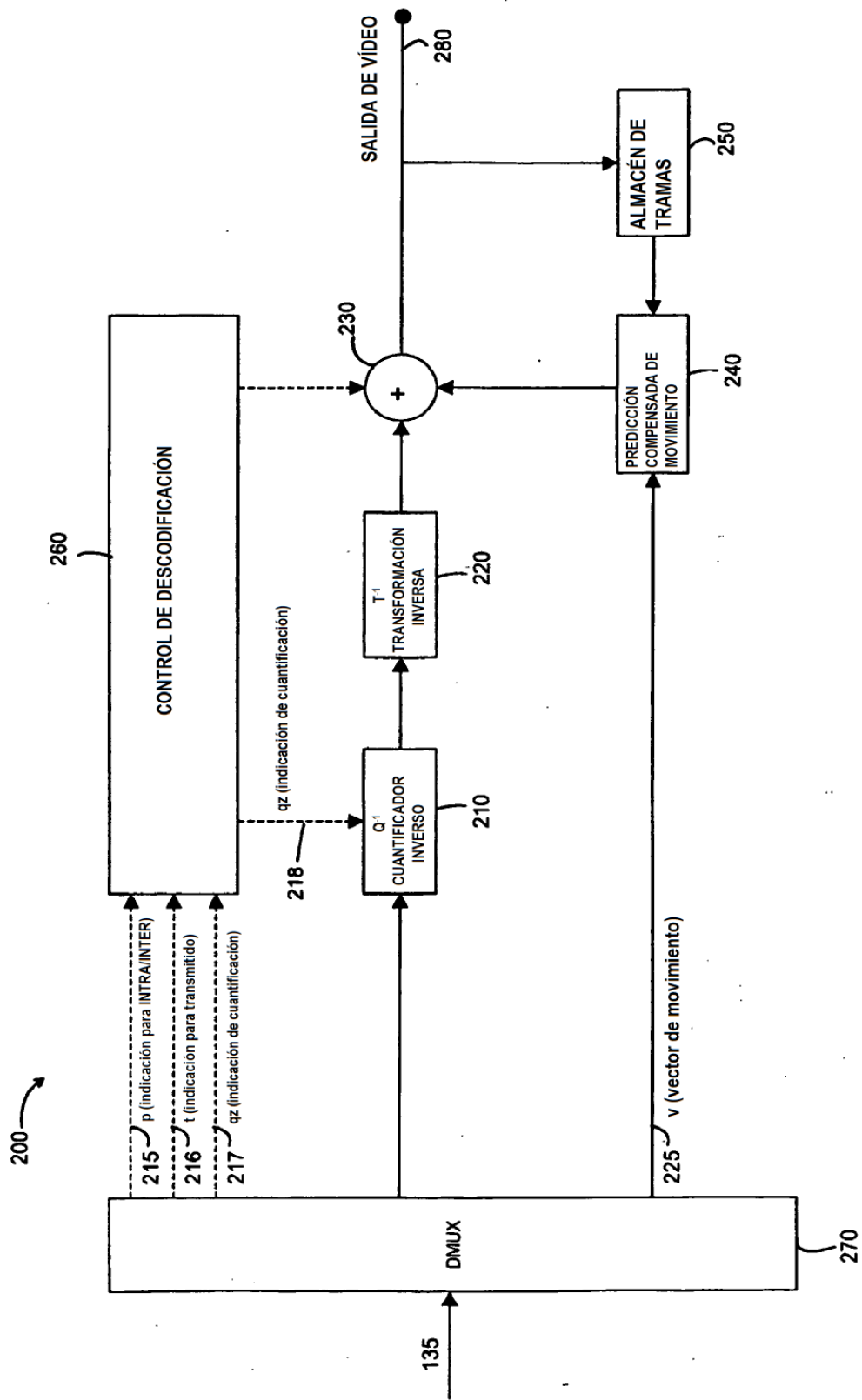
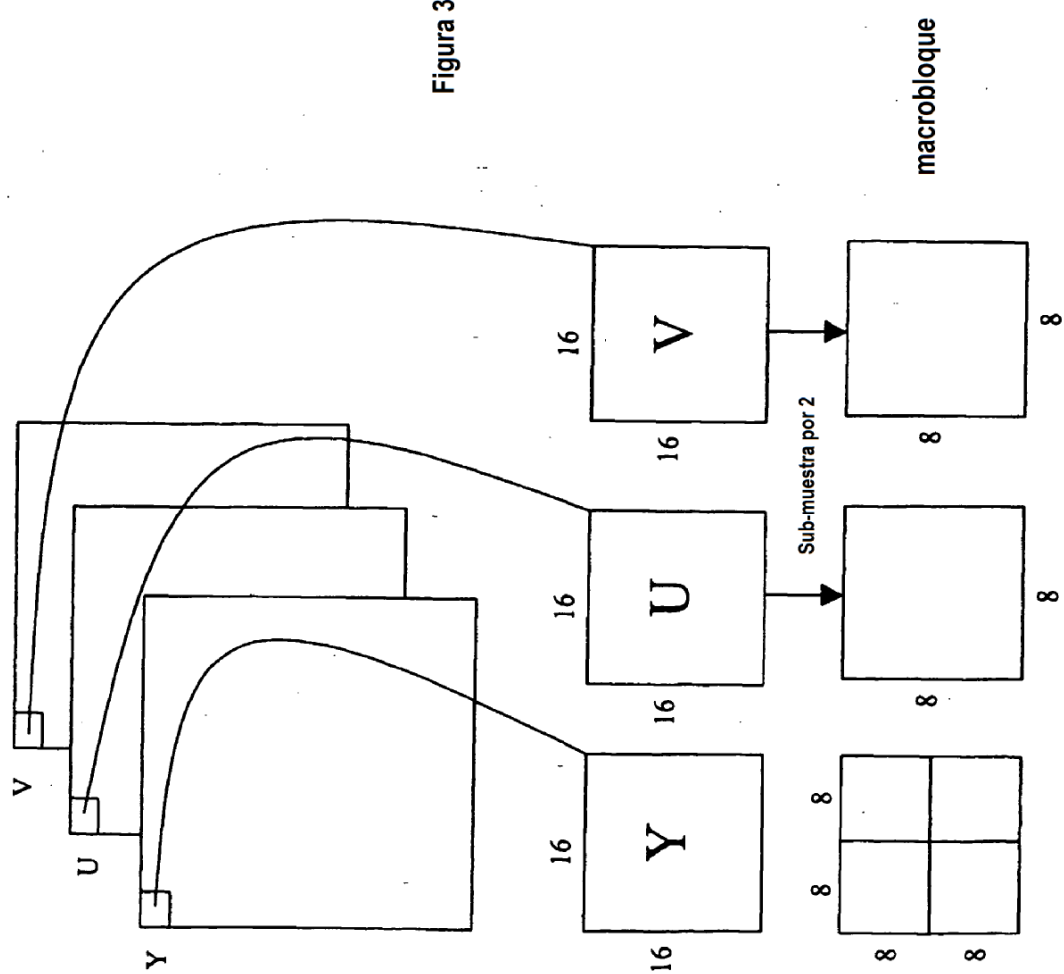


Fig. 2



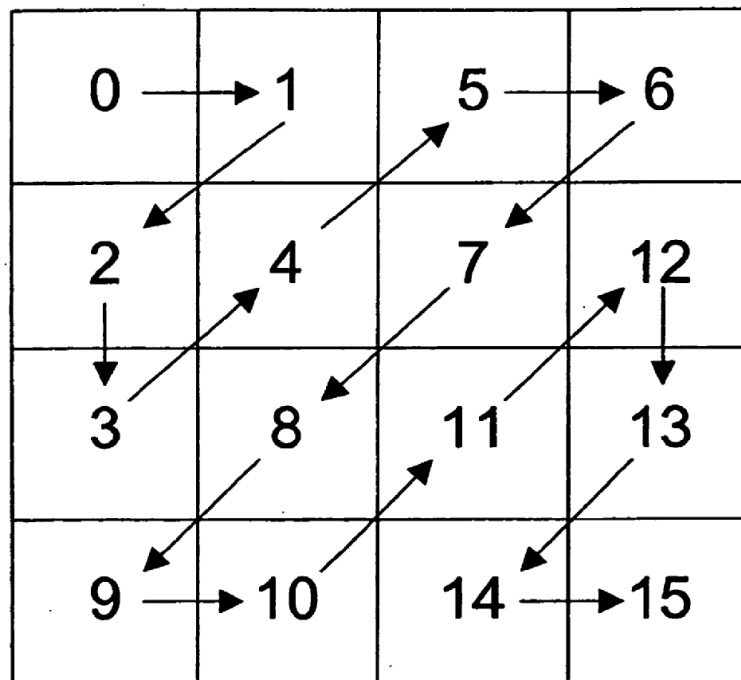


Figura 4

Arriba- izquierda Nc_TL	Arriba Nc_T	Arriba- derecha Nc_TR
Izquierda Nc_L	Bloque actual	

Fig. 5

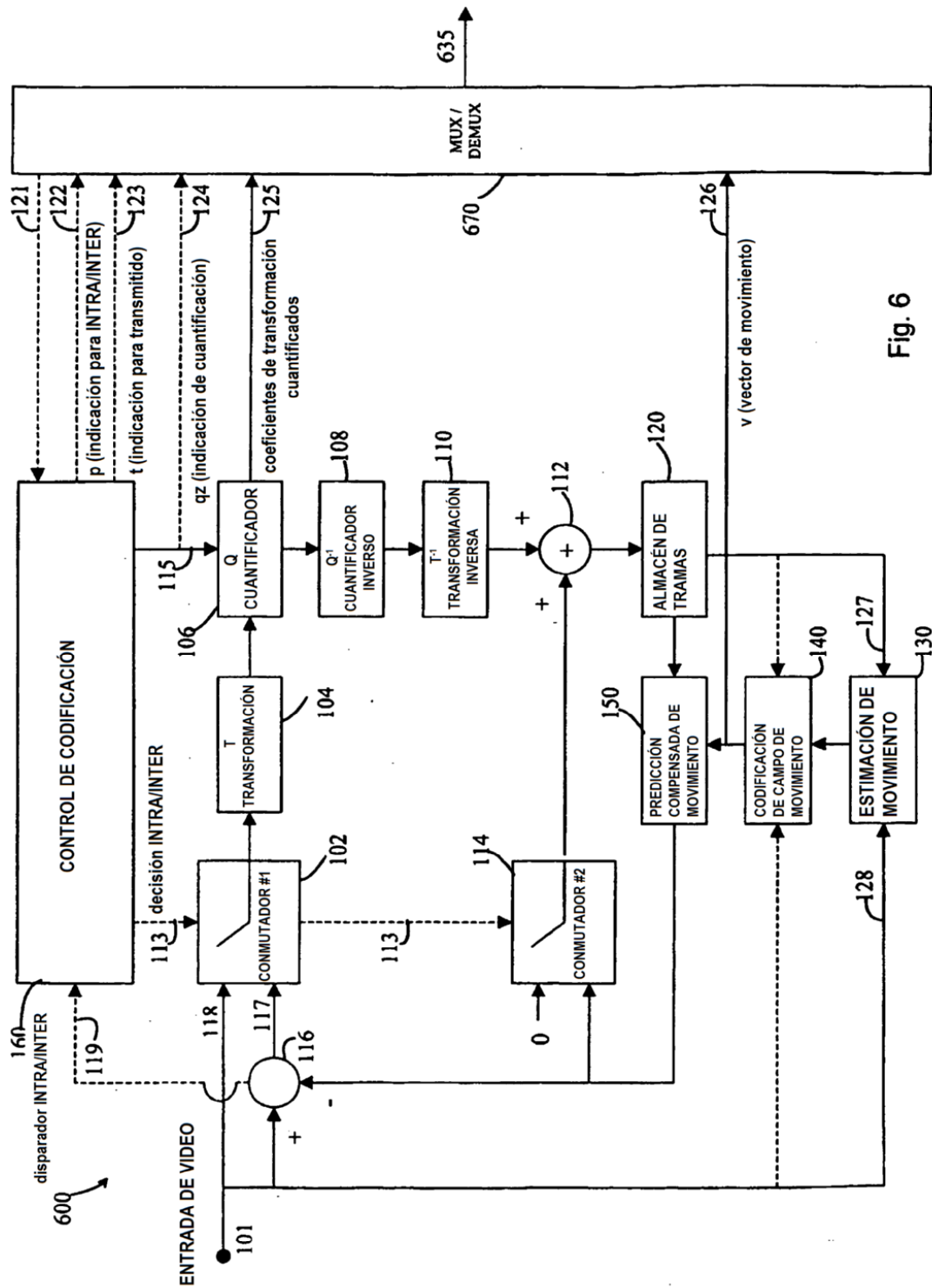


Fig. 6

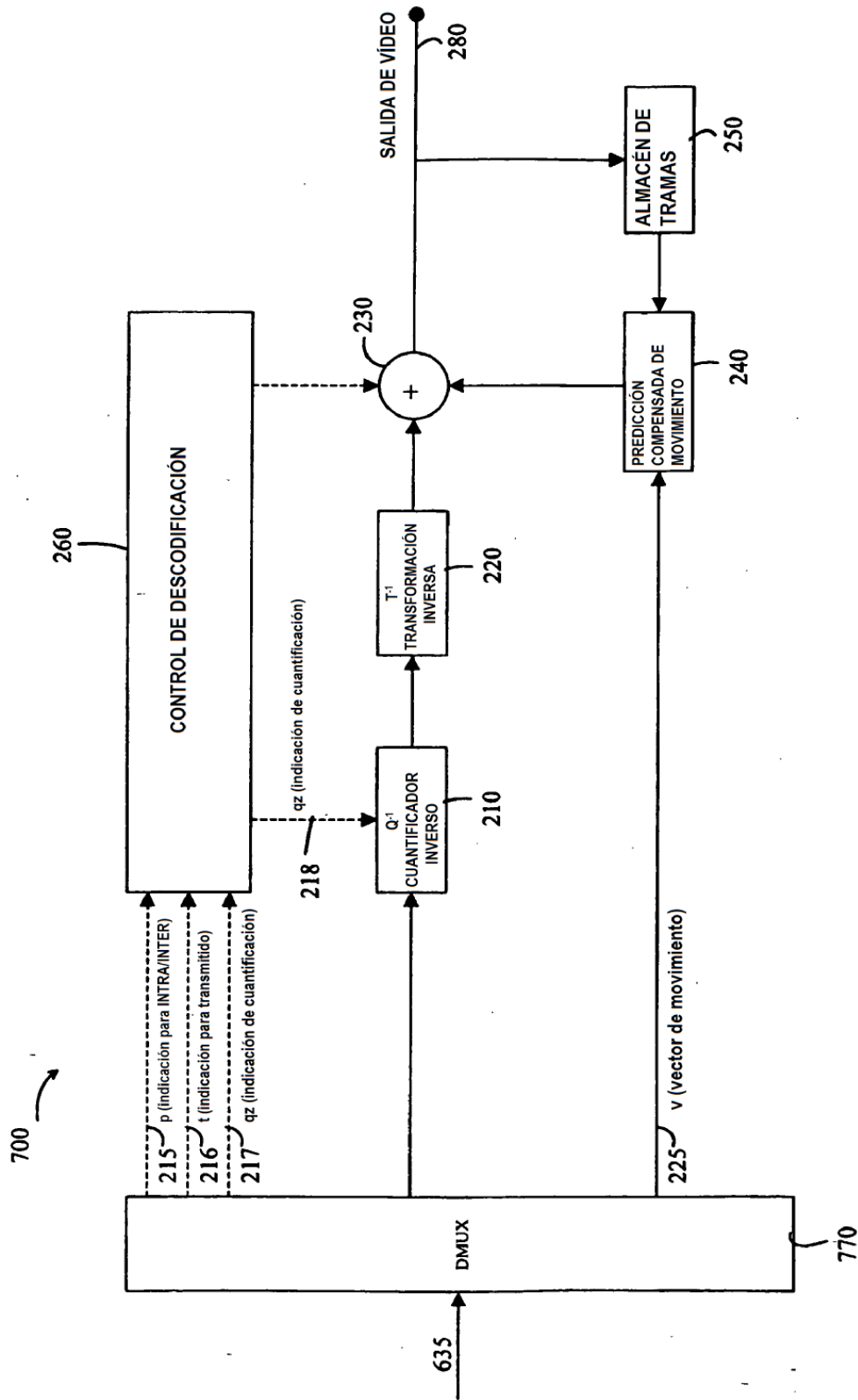


Fig. 7

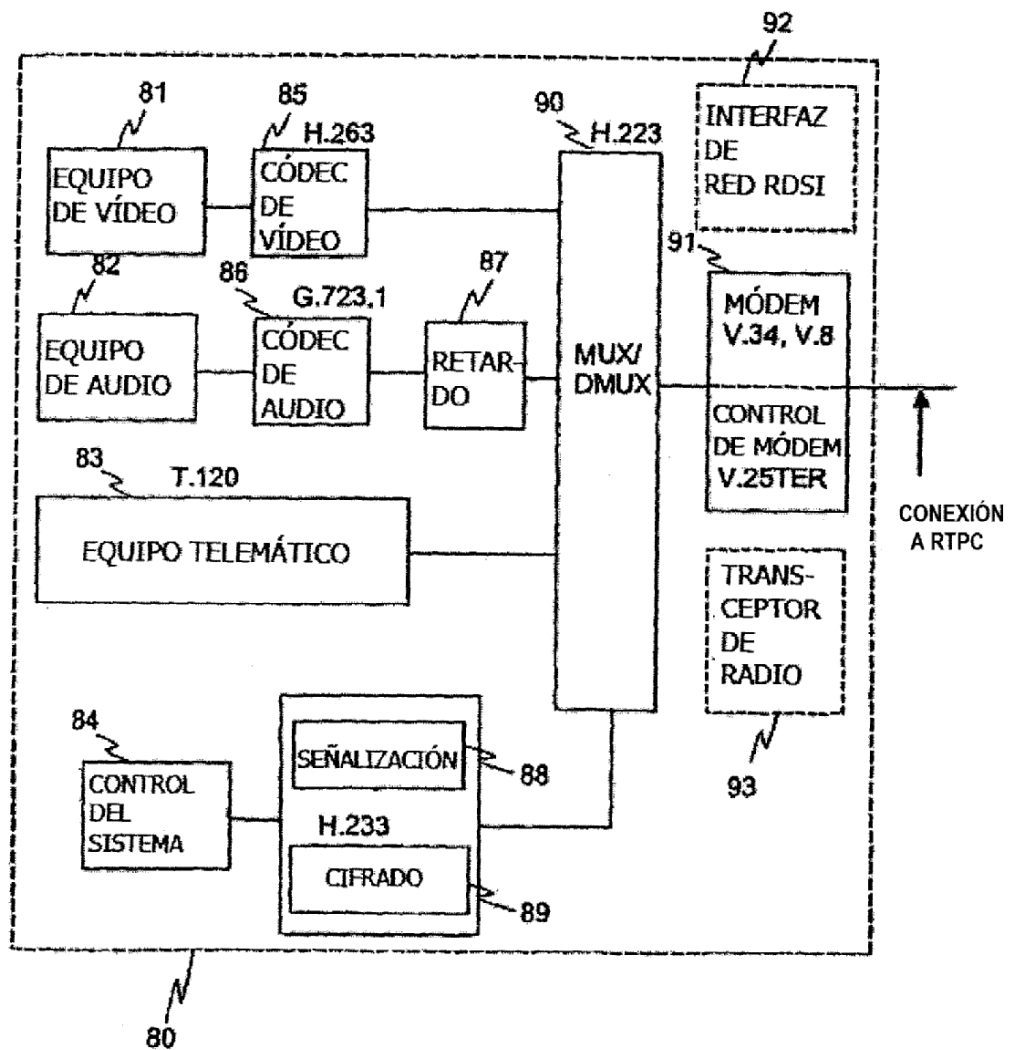


Fig. 8