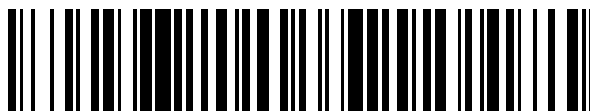


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 559 406**

51 Int. Cl.:

G06K 9/46 (2006.01)

H03M 7/40 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **23.04.2003 E 03718500 (6)**

97 Fecha y número de publicación de la concesión europea: **28.10.2015 EP 1504408**

54 Título: **Sistema y método para codificación aritmética**

30 Prioridad:

23.04.2002 US 374770 P

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

12.02.2016

73 Titular/es:

**NTT DOCOMO, INC. (100.0%)
11-1, Nagatacho 2-chome, Chiyoda-ku
Tokyo 100-6150, JP**

72 Inventor/es:

BOSSEN, FRANK JAN

74 Agente/Representante:

LLAGOSTERA SOTO, María Del Carmen

ES 2 559 406 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

Sistema y método para codificación aritmética

5 SOLICITUDES RELACIONADAS

Esta solicitud reivindica prioridad de la solicitud de patente provisional de Estados Unidos titulada "Bounding The Complexity Of An Arithmetic Coding", de Frank Bossen, presentada el 23 de abril de 2002 y a la que asignó el número S/N 60/374,770.

10 ANTECEDENTES DE LA INVENCION

Esta solicitud se refiere al proceso de datos, y más en particular, a la codificación y decodificación de secuencias de datos en un modo de límite de complejidad reducido.

15 Resulta habitual comprimir una secuencia de datos (es decir, información binaria o M-aria) para el envío desde una primera ubicación a una segunda ubicación. Por ejemplo, los datos de vídeo se pueden generar en la primera ubicación, cuando los datos representan una llamada de videoconferencia. Los datos se convierten habitualmente en una secuencia de eventos (es decir, eventos binarios). Para
20 entregar la secuencia de eventos a la segunda ubicación, es deseable comprimir o codificar la secuencia de eventos en una secuencia de información, con el fin de reducir el tiempo necesario para la transmisión de la secuencia de eventos a la segunda ubicación. Habitualmente se utiliza un codificador aritmético para comprimir la secuencia de eventos (es decir, los eventos binarios) en una secuencia de información de partes de información (es decir, bits binarios de información) para enviar a la segunda ubicación. Cuanto mayor sea la eficiencia de la compresión, o de la codificación, de la secuencia de información, mayor es la velocidad de transmisión de la secuencia de información a la segunda ubicación.

30 La compresión se consigue mediante la generación de menos de una parte de información por evento que está siendo codificado, en que las proporciones de eventos por partes de información que se están codificando pueden alcanzar 64:1 o incluso 128:1, dependiendo de la distribución de probabilidad de los eventos. En la segunda ubicación se utiliza un decodificador aritmético para descomprimir la secuencia de información a la secuencia de eventos, permitiendo de este modo que los datos de vídeo que representan la llamada de videoconferencia se presenten a un individuo en la segunda ubicación.

35 La compresión de datos se utiliza también para otras finalidades. Otra de dichas utilizaciones puede ser para aumentar la calidad de vídeo dada una cantidad de medios de almacenamiento para almacenar la secuencia de evento. En este ejemplo se pueden utilizar codificadores para almacenar vídeo (por ejemplo, que representa una película) en un disco compacto de memoria de sólo lectura (CD-ROM) o disco de vídeo digital de memoria de sólo lectura (DVD-ROM), o cualquier otro medio de almacenamiento incluyendo ópticos, magnéticos o de otro modo, para su reproducción en un momento posterior. Tal como se ha descrito anteriormente, la relación de eventos por partes de información codificada puede llegar a 64:1 o 128:1, dependiendo de la distribución de probabilidad de los eventos. En un momento posterior, se utiliza un decodificador para descomprimir la secuencia de información a la secuencia de eventos, permitiendo de ese modo que se utilice el vídeo correspondiente u otros datos.

40 Sin embargo, a medida que aumenta la eficiencia la codificación en el codificador, se requiere una mayor cantidad de procesamiento (es decir, operaciones de descodificación) en el decodificador para decodificar la secuencia de partes de información de una longitud determinada a la secuencia de evento. Dicho aumento de las necesidades de procesamiento puede ser perjudicial, especialmente cuando se desea que se proporcione la secuencia de eventos en modo de tiempo real. Por ejemplo, en el caso de un vídeo que representa una secuencia de eventos, en tiempo real indica que para la secuencia de información que representa un período de tiempo (es decir, 1 segundo) de vídeo, el decodificador es capaz de recibir la secuencia de información, y de decodificar la secuencia de información a la secuencia de evento correspondiente dentro del período de tiempo. A veces, dicho proceso en tiempo real no es posible con los esquemas de eficiencia aumentada utilizados en los codificadores convencionales.

55 Esta invención se dirige a resolver uno o más de los problemas expuestos anteriormente de una manera nueva y sencilla.

60 La patente US 6198848 describe un aparato de proceso de imágenes que implementa un sistema de control de tasa de manera que la cantidad de bits de datos para cada unidad de bloque individual sea la misma. Los datos transformados de imagen DCT se cuantifican y se codifican con una longitud variable. Se implementa el proceso iterativo para ajustar la cuantificación si se excede un presupuesto determinado. También se menciona el relleno con ceros para garantizar una cantidad fija de datos por bloque.

65 La patente WO9827734 describe un aparato de compresión y descompresión de bloques de longitud fija en que la tasa de bits por palabra comprimida se adapta sobre la base de los bits restantes en el bloque de salida.

BREVE DESCRIPCIÓN DE VARIAS VISTAS DE LOS DIBUJOS

5 La Figura 1 es un diagrama de bloques de un codificador de acuerdo con un ejemplo útil para la comprensión de la invención;
 La Figura 2 es un diagrama de flujo que ilustra la inicialización del codificador de la Figura 1 de acuerdo con un ejemplo útil para la comprensión de la invención;
 La Figura 3 es un diagrama de flujo que ilustra el funcionamiento del motor central del codificador de la Figura 1 de acuerdo con un ejemplo útil para la comprensión de la invención;
 10 La Figura 4 es un diagrama de flujo que ilustra el procedimiento de re-normalización para el codificador de la Figura 1 de acuerdo con un ejemplo útil para la comprensión de la invención;
 La Figura 5 es un diagrama de flujo que ilustra el funcionamiento del motor central del codificador de la Figura 1 al recibir una señal terminar de acuerdo con un ejemplo útil para la comprensión de la invención;
 15 La Figura 6 es un diagrama de flujo que ilustra una rutina de bit y seguir llevada a cabo por el motor central de la Figura 1 de acuerdo con un ejemplo útil para la comprensión de la invención;
 La Figura 7 es un diagrama de bloques que ilustra un decodificador de acuerdo con un ejemplo útil para la comprensión de la invención;
 La Figura 8 es un diagrama de flujo que ilustra la inicialización del decodificador de la Figura 7 de acuerdo con un ejemplo útil para la comprensión de la invención;
 20 La Figura 9 es un diagrama de flujo que ilustra el funcionamiento del motor central del decodificador de la Figura 7 de acuerdo con un ejemplo útil para la comprensión de la invención;
 La Figura 10 es un diagrama de flujo que ilustra la re-normalización del motor central del decodificador de la Figura 7 de acuerdo con un ejemplo útil para la comprensión de la invención;
 25 y
 La Figura 11 es un diagrama de bloques de un codificador de acuerdo con una realización de la invención.

DESCRIPCIÓN DETALLADA DEL INVENTO

30 De acuerdo con un ejemplo útil para la comprensión de la invención, se proporciona un codificador aritmético para convertir una secuencia de eventos que consiste en una pluralidad de eventos en una secuencia de información que consta de al menos una parte de información, incluyendo al menos un puerto de entrada para recibir eventos de la secuencia de eventos, y al menos un puerto de salida, en que
 35 un secuenciador está acoplado con el al menos un puerto de entrada y el al menos un puerto de salida e incluye un generador de contexto para la generación de información de contexto para al menos uno de los eventos recibidos. Se proporciona un estimador de probabilidad, junto con el secuenciador, para recibir un evento de la secuencia de evento y la correspondiente información de contexto desde el secuenciador, y para generar una estimación de la probabilidad de que el evento tenga un valor determinado. Un motor central está acoplado con el secuenciador y el estimador de probabilidad para recibir el evento de la secuencia de eventos desde el secuenciador, y la estimación de la probabilidad del estimador de probabilidad, y para generar 0 o más partes de la secuencia de información que responde al evento recibido y la estimación de probabilidad. El motor central puede limitar la relación de eventos por partes de información, y proporciona las 0 o más partes de información la secuencia de información al
 40 secuenciador, en que las partes de información generadas se proporcionan como la secuencia de información en el al menos un puerto de salida.

De acuerdo con otro ejemplo útil para la comprensión de la invención, también se proporciona un decodificador aritmético para convertir una secuencia de información que consta de al menos una parte de información en una secuencia de eventos compuesta de una pluralidad de eventos, incluyendo al menos un puerto de entrada para recibir partes de información de la secuencia de información, al menos un puerto de salida, y un secuenciador acoplado junto con el al menos un puerto de entrada y el al menos un puerto de salida y que incluye un generador de contexto para la generación de información de contexto para al menos un evento generado. El estimador de probabilidad está acoplado con el secuenciador para recibir un evento generado de la secuencia de evento y la información de contexto correspondiente, y para determinar una estimación de probabilidad. Un motor central está acoplado con el secuenciador y el estimador de probabilidad para el procesamiento de cero o más partes de información de la secuencia de información desde el secuenciador que responde a la estimación de probabilidad de generar un evento en base a una relación limitada de eventos por partes de información. El evento generado se proporciona al
 50 estimador de probabilidad con la información de contexto correspondiente desde el secuenciador para actualizar la estimación de probabilidad, y el evento generado se proporciona al secuenciador para su transmisión como la secuencia de eventos desde el al menos un puerto de salida.

Tener el codificador aritmético que codifica la secuencia de eventos para la secuencia de información mediante la limitación de la relación de eventos por partes de información permite la compresión de una secuencia de datos (por ejemplo, eventos binarios o M-arios) de una manera tal que permite la utilización en tiempo real de la secuencia de evento. Además, la reducción de la complejidad de la secuencia de la información generada mediante la limitación de la relación de eventos por partes de información permite a
 65

los decodificadores con capacidades de procesamiento limitada (por ejemplo, decodificadores móviles) decodificar la secuencia de información de nuevo a su secuencia de evento correspondiente a la vez que manteniendo potencialmente una capacidad de procesamiento en tiempo real, o al menos sin necesidad de que un usuario de la información tenga que esperar una cantidad significativa de tiempo, mientras dichos descodificadores realizan las operaciones de descodificación.

De forma similar, proporcionar un decodificador que decodifica la secuencia de información a una secuencia de eventos en base a una relación limitada de eventos por parte de información permite dicha utilización en tiempo real de la secuencia de eventos, por ejemplo, durante una llamada de videoconferencia, o la recuperación y el proceso de la información desde los medios de almacenamiento, por ejemplo la recuperación de información de vídeo, como por ejemplo una película de un DVD-ROM y el procesamiento de los datos de vídeo para su utilización.

De acuerdo con otra realización de la invención, se proporciona un codificador de entropía para la conversión de al menos una secuencia de eventos, en que cada secuencia de eventos incluye una pluralidad de eventos, a al menos una secuencia de información, en que cada secuencia de información incluye al menos una parte de información, incluyendo un codificador aritmético para la generación de cero o más partes de información de al menos una secuencia de información que responde a al menos un evento de al menos una secuencia de eventos. El codificador de entropía incluye además un controlador acoplado con el codificador aritmético para restringir una serie de eventos en al menos una secuencia de eventos como una función de la cantidad de partes de información generadas en al menos una secuencia de información. La al menos una secuencia de eventos puede representar datos de entrada divididos en un número de segmentos, en que el controlador restringe una serie de eventos como una función del número de partes de información al restringir un número máximo de eventos en al menos una secuencia de eventos como una función de una combinación lineal que incluye un número de partes de información generadas en al menos una secuencia de información, y el número de segmentos. El codificador de entropía puede estar acoplado con un procesador para convertir datos de entrada en la al menos una secuencia de eventos, en que el codificador de entropía es capaz de enviar información al procesador con respecto a una serie de partes de información generadas en al menos una secuencia de información. Además, el procesador puede ser capaz de reducir una relación de eventos de al menos una secuencia de eventos que responde a la información recibida desde el codificador de entropía.

Tener un codificador de entropía capaz de limitar una serie de eventos en una secuencia de eventos como una función del número de partes de información generadas en una secuencia de información puede permitir que los datos de vídeo sean decodificados de forma consistente, lo que representa un número máximo de partes de información por grupo de segmentos, a la vez que se mantiene la calidad de codificación de vídeo, en comparación con los codificadores de la técnica anterior. Además, cuando el codificador recibe una indicación de que el umbral de complejidad está a punto de ser excedido, el codificador puede generar menos eventos para el siguiente segmento de los datos de entrada, por ejemplo reduciendo la posibilidad de que se supere el umbral de limitación para el siguiente segmento. Además, dado que el controlador es capaz de desencadenar la adición de bits de relleno cuando se supera el umbral de limitación, el procesador puede ser implementado de una manera tal que el procesador no tiene por qué tener en cuenta, o tratar, la restricción de la complejidad. Los codificadores y el decodificador se pueden utilizar en el procesamiento de datos de vídeo, por ejemplo, tal como son generados por un procesador de vídeo (es decir, un códec de vídeo). En el ejemplo del procesador de vídeo, una imagen de vídeo se graba, y se divide en bloques de muestra de datos que puede representar muestras de 16x16, 8x8, o 4x4 de la imagen grabada. A continuación los bloques son transformados por el procesador de vídeo (es decir, utilizando una transformación discreta del coseno), y son cuantificados para producir valores enteros que representan el bloque de muestra. Los valores enteros son convertidos en una secuencia de eventos (es decir, eventos binarios) por el procesador de vídeo, y se envían al codificador para la codificación. Como alternativa, el procesador de vídeo puede operar directamente sobre muestras individuales, transformándolas y cuantificándolas, y convirtiendo el valor entero cuantificado particular para la muestra en una secuencia de eventos.

Aunque una utilización de este tipo para el codificador y el decodificador que se describe en el presente documento es en la codificación y decodificación de datos de vídeo, un experto en la técnica se daría cuenta de que el codificador y el decodificador que se describen en el presente documento se pueden utilizar en cualquier situación en la que una secuencia de eventos se comprime a una secuencia de información en el caso del codificador, y en que una secuencia de información de este tipo se descomprime en el caso del decodificador. Además, aunque la siguiente descripción del codificador es en el contexto del procesamiento de una secuencia de eventos que comprenden una pluralidad de eventos binarios a una secuencia de información que comprende al menos una información de bit binario, y para el decodificador está en el contexto del procesamiento de una secuencia de información que comprende al menos una información de bit binario a una secuencia de eventos que comprenden una pluralidad de eventos binarios, el codificador y el decodificador podrían operar en secuencias de eventos y secuencias de información que constan de eventos que son M-arios por naturaleza (es decir, cada evento M-ario

representa > 1 bits de datos) utilizando las enseñanzas descritas en este documento, tal como sería apreciado por un experto en la técnica.

5 La Figura 1 ilustra un diagrama de bloques de un codificador aritmético 100 de acuerdo con un ejemplo útil para la comprensión de la invención. El codificador aritmético 100 incluye un secuenciador 105, un estimador de probabilidad 110, y un motor central 115, que se acoplan entre sí. Una o más líneas de datos de entrada 120 proporcionan un puerto de entrada para recibir una secuencia de eventos 125, en este caso una secuencia ordenada de eventos binarios, en el codificador 100. Los eventos binarios de la secuencia 125 pueden tener un valor "A" o "B". La secuencia de eventos es procesada por el codificador, tal como se describe a continuación, para generar una secuencia de información que es una secuencia ordenada compuesta de al menos una parte de información, en este caso bits binarios, en que el número de partes de información en la secuencia de información es menor que el número de eventos en la secuencia de eventos. Las líneas de datos de salida 130 proporcionan un puerto de salida para el envío de la secuencia de información 135 desde el codificador 100. La secuencia ordenada de bits binarios de la secuencia de información incluye uno o más bits que tienen un valor de "0" o "1".

10 Al recibir la secuencia ordenada de eventos binarios 125, el secuenciador 105 transmite secuencialmente los eventos 125 tanto al estimador de probabilidades 110 como al motor central 115 a través de líneas de transmisión de evento 140 y 145, respectivamente. Para cada evento binario de la secuencia de eventos 125, el secuenciador 105 también transmite información de contexto al estimador de probabilidad 110 para el evento binario a través de una línea de transmisión de contexto 150. El estimador de probabilidad 110, utilizando la información de contexto recibida a través de las líneas de transmisión de contexto 150, genera una estimación de la probabilidad $P(A)$ que se transmite al motor central 115 a través de la línea de transmisión de probabilidad 155. A continuación, el estimador de probabilidad 110, actualiza su estado interno basándose en el valor del evento binario recibido a través de la línea de transmisión de evento binario 140. El motor central 115 produce 0 o más bits de información utilizando el evento binario recibido a través de la línea de transmisión de eventos 145 y la correspondiente estimación de probabilidad $P(A)$ recibida a través de la línea de transmisión de probabilidad 155, a la vez que limita una relación de los eventos por bits de información, tal como se describirá con mayor detalle a continuación.

15 El secuenciador 105, tras la recepción de un primer evento binario de la secuencia de eventos 125, envía una señal de inicialización al estimador de probabilidad 110 y el motor central 115 a través de las líneas de inicialización 185 y 190, respectivamente.

20 En la producción de los cero o más bits de información para la transmisión, a través de la línea de transmisión de bits 158, el motor central 115 utiliza varios registros que incluyen un registro de intervalo 165, un registro bajo 170, un registro de bits pendientes 175, y un registro contador 180. El funcionamiento del codificador 100 se describe con respecto a los diagramas de flujo de las Figuras 2-6.

25 La Figura 2 ilustra un diagrama de flujo que describe el funcionamiento del motor central 115 cuando recibe una señal de inicialización desde el secuenciador 105. Tal como se muestra en el diagrama de flujo de la Figura 2, el motor central 115 inicializa el registro de intervalo 165, por ejemplo a un valor hexadecimal 8000 (H), tal como se muestra en el paso 205. El motor central 115 también inicializa el registro bajo 170, el registro de bits pendientes 175 y el registro contador 180 cada uno de ellos para, por ejemplo, un valor de 0, tal como se muestra en los pasos 210, 215 y 220, respectivamente.

30 Después de la inicialización, y al recibir un evento binario de la secuencia de eventos 125, el motor central opera como se ilustra en la diagrama de flujo de la Figura 3, de acuerdo con un ejemplo útil para la comprensión de la invención.

35 Tal como se muestra en el paso 305 de la Figura 3, el motor central 115 calcula un valor de sub-intervalo $R(A)$ para un valor de evento en particular, por ejemplo un valor del evento "A", utilizando un valor actual del registro de intervalos 165 y la estimación de probabilidad para ese valor de evento en particular recibida a través de la línea de transmisión de probabilidad 155. En este caso, el valor de sub-intervalo $R(A)$ se calcula multiplicando el valor del registro de intervalos "R" por la estimación de probabilidad $P(A)$. Tal como se muestra en el paso 310, un comparador del motor central 115 (que no se muestra) compara un valor "X" del evento binario recibido a través de la línea de transmisión de evento 145 con el valor del evento en particular A para el que se ha generado la probabilidad $P(A)$. Cuando el valor evento recibido X es igual a A, el registro de intervalos 165 se establece en el valor de sub-intervalo $R(A)$, tal como se muestra en el paso 315 (es decir, el valor de $R(A)$ se almacena en el registro de intervalos 165). Sin embargo, cuando el valor de evento recibido X no es igual al evento A en el paso 310, se resta el sub-intervalo $R(A)$ del valor R del registro de intervalos 165, en que el valor resultante se almacena en el registro de intervalos 165, tal como se muestra en el paso 320. Un valor L del registro bajo 170 se suma con el valor de sub-intervalo $R(A)$ y se almacena en el registro bajo 170 tal como se muestra en el paso 325. Al recalcular el valor del registro de intervalo 165 en el paso 315, o el registro bajo 170 en el paso 325, el motor central llama a una rutina de re-normalización, tal como se muestra en el paso 330 y se

describe más adelante con respecto al diagrama de flujo de la Figura 4, de acuerdo con un ejemplo útil para la comprensión de la invención.

5 Tal como se muestra en la Figura 4, la re-normalización se inicia cuando un valor del registro de intervalos 165 se compara con un valor de intervalo predeterminado, en este caso 4000H tal como se muestra en el paso 405. Cuando el valor R del registro de intervalos 165 es $> 4000H$, el registro contador se incrementa en 1, un valor de incremento predeterminado, tal como se muestra aquí en el paso 410. Cuando se determina en el motor central 115 que el valor de R es menor o igual a 4000H en el paso 405, el valor L del registro bajo 170 y el valor R del registro de intervalos 165 se suman, y se compara el resultado con el valor 8000H tal como se muestra en el paso 415. Cuando la suma es menor o igual a 8000H, se llama a una rutina de Bit y seguir que tiene un parámetro B igual a "0" tal como se muestra en el paso 420 y tal como se describirá más adelante. A continuación, el valor del registro bajo se dobla tal como se muestra en el paso 425, y el valor C del registro contador 180 se compara con 0 tal como se muestra en el paso 430. Cuando el valor C del registro contador 180 no es menor que 0, un valor de decremento, en este caso 4H, se resta del registro contador 180, tal como se muestra en el paso 435, y el flujo vuelve al paso 405 y funciona tal como se ha descrito anteriormente. Sin embargo, cuando se determina en el paso 430 que el valor C del registro contador es menor que 0, el registro de intervalos se reajustará doblando el valor R del registro de intervalos 165, tal como se muestra en el paso 440, y el flujo del programa continúa a la etapa 435 tal como se ha descrito anteriormente.

20 Cuando se determina en el paso 415 que la suma de los valores L y R no es menor que o igual a 8000H, el valor L del registro bajo 170 se compara con 8000H tal como se muestra en el paso 445. Cuando el valor L es mayor que o igual a 8000H, se llama a la rutina de Bit y seguir con un parámetro B igual a "1" tal como se muestra en el paso 450 y como se describe a continuación. Se resta un valor 8000H del registro bajo 170 tal como se muestra en el paso 455, y el flujo continúa al paso 425 tal como se ha descrito anteriormente. No obstante, cuando se determina que L no es mayor que o igual a 8000H en el paso 445, el registro de bits pendientes 175 se incrementa en 1, tal como se muestra en el paso 460, y se resta un valor 4000H del registro bajo 170 tal como se muestra en el paso 465. A continuación, el flujo continúa al paso 425 tal como se ha descrito anteriormente.

30 Tras la recepción de un evento final de la secuencia de eventos 125, el secuenciador 105 envía una señal de finalización al motor central 115 a través de la línea de transmisión de finalización 195. El funcionamiento del motor central 115 cuando se recibe una señal de finalización se muestra en el diagrama de flujo de la Figura 5, de acuerdo con un ejemplo útil para la comprensión de la invención.

35 Tal como se muestra en la Figura 5, se llama a una rutina de Bit y seguir, el paso 505, con un argumento B que tiene el valor del bit más significativo (MSB) del registro bajo 170. Los bits menos significativos (LSB) del registro bajo 170 son enviados a continuación al secuenciador 105 tal como se muestra en el paso 510. La rutina de Bit y seguir se muestra con más detalle en el diagrama de flujo de la Figura 6, de acuerdo con un ejemplo útil para la comprensión de la invención.

40 Tal como se muestra en la Figura 6, se envía un valor B como un bit de la secuencia de información 135 al secuenciador 105 a través de la línea de transmisión de bits 158, tal como se muestra en el paso 605. Un valor BO del registro de bits pendientes 175 se compara con 0, en el paso 610. Cuando un valor BO del registro de bits pendientes 175 no es mayor que 0, el flujo vuelve a cualquier segmento de programa / rutina que llamó a la rutina de Bit y seguir (es decir, la caja 420 o la caja 450 de la Figura 4 o la caja 505 de la Figura 5). Sin embargo, cuando el valor BO es mayor que 0 en el paso 610, un bit de la secuencia de información 135 que tiene un valor de 1 menos el valor B se envía al secuenciador 105 a través de la línea de transmisión de bits 158, tal como se muestra en el paso 615. El valor BO del registro de bits pendientes se decrementa en 1 tal como se muestra en el paso 620, y el flujo vuelve al paso 610, y continúa tal como se ha descrito anteriormente.

45 Por lo tanto, proporcionar al codificador 100 que tiene un motor central 115 el contador 180 permite el seguimiento de la relación de eventos por partes de información, permitiendo de esta manera que dicha relación esté limitada con el fin de reducir el límite de complejidad de la secuencia de información proporcionada por el codificador. El valor de incremento del contador descrito con respecto al paso 410, el valor del contador de decremento descrito con respecto al paso 435 y el valor con el que se compara el registro contador 180 en el paso 430 se utilizan para regir la relación media de los eventos por partes de información. Por ejemplo, con los valores descritos anteriormente con respecto a la Figura 4, que tiene un valor de decremento 4 en el paso 435 los resultados en la relación media de eventos por partes de información está limitada a 4. La alteración del valor de decremento altera la proporción media de los eventos por partes de información. En consecuencia, el valor de incremento, el valor de decremento y el valor con el que se compara el contador son meramente ejemplares, ya que dichos valores, junto con el valor al que se ha inicializado el registro contador 180 en la caja 220 de la Figura 2, no necesitan ser tal como se ha descrito anteriormente, sino que pueden ser cualquier valor entero, en el que se seleccionarán los valores correspondientes para lograr la relación media limitada deseada de eventos por partes de información, tal como sería apreciado por un experto en la técnica.

Las Figuras 7-10 ilustran un decodificador aritmético y el funcionamiento del mismo que es capaz de decodificar una secuencia de información que consta de al menos una parte de información en una secuencia de eventos que comprende una pluralidad de eventos, que representa una relación limitada de eventos por partes de información en la secuencia de información. La Figura 7 es un diagrama de bloques del decodificador, en que las Figuras 8-10 son diagramas de flujo que ilustran el funcionamiento del decodificador.

La Figura 7 ilustra un decodificador aritmético 700 de acuerdo con un ejemplo para la comprensión de la invención. El decodificador 700 incluye un secuenciador 705, un estimador de probabilidad 710, y un motor central 715 acoplados entre sí. Una o más líneas de datos de entrada de secuencia de información 720 proporcionan un puerto de importación para una secuencia de información 725, en este caso en una secuencia ordenada de bits binarios, al decodificador 700. Los bits binarios de la secuencia de 725 pueden tener un valor "0" o "1". El decodificador procesa la secuencia de información para generar una secuencia de eventos teniendo en cuenta la relación limitada de eventos por partes de información en la secuencia de información, tal como se describe más adelante. La secuencia de eventos generada es una secuencia de eventos ordenados que comprende una pluralidad de eventos, en este caso eventos binarios, que pueden tener un valor "A" o "B". La secuencia de eventos se proporciona a las líneas de datos de secuencias de eventos de salida 730, que proporciona al menos un puerto de salida desde el decodificador 700.

Al recibir la secuencia ordenada de bits binarios 725, el secuenciador 705 transmite secuencialmente los uno o más bits al motor central 715 a través de la línea de transmisión de bits 740. Tras la recepción de los uno o más bits, el motor central 715 inicia la generación de eventos, en este caso eventos binarios, que se transmiten al secuenciador 705 y al estimador de probabilidad 710 a través de líneas de transmisión de evento 745 y 750 respectivamente. Tal como se describe más adelante, el estimador de probabilidad envía una estimación de probabilidad inicial al motor central para generar el primer evento binario. A continuación, para cada evento binario generado por el motor central 715 y enviado al secuenciador 705, el secuenciador 705 transmite un contexto correspondiente al estimador de probabilidades 710 sobre la línea de transmisión de contexto 755. Basándose en el valor del contexto recibido a través de la línea de transmisión de contexto 755, el estimador de probabilidad 710 genera una estimación de probabilidad correspondiente $P(A)$, que se envía al motor central 715 sobre la línea de transmisión de probabilidad 760, y es utilizada por el motor central 715 en la generación de nuevos eventos. Después de transmitir la estimación de probabilidad $P(A)$, el estimador de probabilidad 710 actualiza su estado interno basándose en el valor del evento binario recibido desde el motor central 715 sobre la línea de transmisión de evento 750. El motor central 715 consume cero o más bits de información para cada evento binario generado. El motor central 715 utiliza varios registros en la generación de los eventos de la secuencia de eventos 735, incluyendo un registro de intervalos 765, un registro de valores 770 y un registro contador 775. Se proporciona una señal de inicialización al motor central 715 y al estimador de probabilidad 710 a través de líneas de transmisión de inicialización 780 y 785, respectivamente. Se proporciona una señal de finalización al motor central 715 a través de una línea de finalización de señal 790. El funcionamiento del decodificador 700 se muestra en los diagramas de flujo de las Figuras 8-10.

La Figura 8 ilustra el funcionamiento del decodificador 700 al recibir una señal de inicialización a través de la línea de señal de inicialización 780 de acuerdo con un ejemplo para la comprensión de la invención. Tal como se muestra en la Figura 8, el motor central 715 inicializa el registro de intervalos 765 a un valor de 8000H tal como se muestra en el paso 800, e inicializa el valor del registro 770 a un valor particular representado por 16 bits (que se describe más adelante) tal como se muestra en el paso 805. Además, el motor central 715 inicializa el registro contador 775 a un valor de cero tal como se muestra en el paso 810.

El registro de valores 770 se inicializa en el paso 805 mediante la recopilación de 16 bits en el registro de valores 770, por ejemplo las primeras 16 partes de información de la secuencia de información, tal como se describe en Arithmetic Coding Revisited, de Moffat et al., ACM Transactions on Information Systems, VOL. 16, NO. 3, julio de 1998, páginas 256-294, que se incorpora como referencia en el presente documento, y tal como sería apreciado por un experto en la técnica.

La Figura 9 es un diagrama de flujo que ilustra el funcionamiento del motor central 715 en la generación de la secuencia de evento 725 de acuerdo con un ejemplo para la comprensión de la invención. Tal como se muestra en el paso 900, un valor de subintervalo $R(A)$ es calculado por el motor central 715 para un valor de evento "A". El valor de subintervalo $R(A)$ se calcula basándose en el valor del registro de intervalo 765 y la estimación de probabilidad determinada $P(A)$ recibida a través de la línea de transmisión de estimación de probabilidad 760 del estimador de probabilidad 710, multiplicando el valor R del registro de intervalo 765 por la estimación de probabilidad $P(A)$. Cuando se calcula el valor de sub-intervalo $R(A)$ por primera vez para una secuencia de información particular 735, el estimador de probabilidad 710 determina que la estimación de la probabilidad $P(A)$ es alguna estimación de probabilidad inicial /

predeterminada (es decir, acordada por anticipado). A partir de entonces, a medida que los eventos se emiten desde el motor central 715, el estimador de probabilidad 710 utiliza los eventos emitidos junto con información de contexto correspondiente recibida a través de línea de la información de contexto 755 en la actualización de la estimación de la probabilidad $P(A)$ proporcionada a través de línea de transmisión de estimación de probabilidad 760, tal como se describe a continuación.

El valor de subintervalo $R(A)$ se compara entonces con un valor V del registro de valores 770, tal como se muestra en el paso 905. Cuando el valor de sub-intervalo $R(A)$ es menor que el valor V del registro de valores 765, el registro de intervalos 165 se establece en el valor de sub-intervalo $R(A)$ tal como se muestra en el paso 910, y un evento "A" se emite al secuenciador 705 a través de la línea de transmisión de eventos 745 tal como se muestra en el paso 915. A continuación se llama a un proceso de re-normalización, tal como se muestra en el paso 920, y que se describirá con más detalle con respecto al diagrama de flujo de la Figura 10.

Cuando el valor de sub-intervalo $R(A)$ no es menor que el valor V del registro de valores 770 en el paso 905, el valor de sub-intervalo $R(A)$ se resta del registro de intervalos 765, tal como se muestra en el paso 925. El valor de sub-intervalo $R(A)$ se resta del registro de valores 770, tal como se muestra en el paso 930, y un evento "B" se emite desde el motor central al secuenciador 705 a través de línea de transmisión de evento 745, tal como se muestra en el paso 935. El flujo continúa con el procedimiento de re-normalización en el paso 920.

Tal como se ha descrito anteriormente, después de la emisión de un evento a través de la línea de transmisión de evento 745, el secuenciador 705 transmite un contexto a través de la línea de transmisión de contexto 755 al estimador de probabilidad 210, en que el estimador de probabilidad 710 actualiza su estado interno basándose en el valor del evento binario emitido por el motor central 715.

La Figura 10 es un diagrama de flujo que ilustra el funcionamiento del motor central 115 al realizar la re-normalización tal como se ha descrito anteriormente con respecto al paso 920 de acuerdo con un ejemplo para la comprensión de la invención. Tal como se muestra en el paso 1000, el valor R del registro de intervalo 765 se compara con un valor de intervalo predeterminado, en este caso $4000H$. Cuando el valor predeterminado R no es menor que o igual a $4000H$, el registro contador 775 se incrementa en un valor de incremento predeterminado, en este caso 1, tal como se muestra en el paso 1005, y la re-normalización es completa. Sin embargo, cuando el valor R del registro de intervalos 765 es menor o igual a $4000H$ un valor C del registro contador 775 se compara con 0 tal como se muestra en el paso 1010. Cuando el valor C del registro contador 775 no es menor que 0, el valor V del registro de valores 770 se duplica, y se incrementa con un valor del bit más reciente recibido en el motor central 715 a través de la línea de transmisión de bits 740, en que la suma resultante se almacena en el registro de valores 770, tal como se muestra en el paso 1015. Tal como se muestra en el paso 1020, un valor C del registro contador 775 se decrementa en un valor de decremento predeterminado, en este caso $4H$, y el flujo vuelve al paso 1000. Cuando un valor C del registro contador es menor que 0 en el paso 1010, el registro de intervalo 765 se reajustará duplicando el valor R del registro de intervalo tal como se muestra en el paso 1025, y el flujo continúa al paso 1015 tal como se ha descrito anteriormente.

Después de que una parte final de la información de la secuencia de información 725 se transmita a, y sea procesada por el motor central 715, el secuenciador 705 transmite una señal de terminar al motor central 715 a través de líneas de transmisión de señal de terminar 790 que indican al motor central 715 que la decodificación de la secuencia de información 725 es completa. El motor central 715 no necesita realizar ninguna operación en respuesta a la señal de terminación.

El codificador 100 con un límite de complejidad reducido que es capaz de codificar una secuencia de eventos a una secuencia de información limitando una relación de eventos por partes de información resulta ventajoso ya que proporciona una secuencia de información que puede ser decodificada y utilizada para soportar la prestación de la información para su uso en tiempo real, sin afectar significativamente a la eficiencia de codificación de la secuencia de información. Además, una codificación de complejidad reducida de este tipo puede ser beneficiosa cuando el decodificador utilizado en la decodificación de la secuencia de información posee una capacidad de procesamiento reducida (es decir, se encuentra en un dispositivo móvil) ya que un decodificador de este tipo sería capaz de decodificar la secuencia de información sin ningún retraso indebido en espera de que se lleve a cabo la decodificación. Además, el decodificador 700 capaz de decodificar una secuencia de información a una secuencia de eventos a través de una relación limitada de eventos por partes de información en la secuencia de información, puede resultar ventajoso al permitir una decodificación eficiente de la secuencia de información para su utilización en, por ejemplo, el uso en tiempo real de la secuencia de evento.

La Figura 11 ilustra un codificador 1100 de acuerdo con otra realización de la invención. Tal como se muestra en la Figura 11, el codificador incluye un procesador 1105, acoplado con un codificador de entropía 1110. El codificador de entropía incluye un codificador aritmético 1115 acoplado con un

controlador 1120. Los datos de entrada 1125 son recibidos en el codificador, y se codifican a una o más secuencias de partes de información que se muestran en 1130.

5 Los datos de entrada 1125 pueden ser cualquier dato que vaya a ser codificado por el codificador 1100, incluyendo, por ejemplo, datos de vídeo. El procesador 1105 recibe los datos de entrada, y convierte los datos de entrada en una o más secuencias de eventos 1135. El procesador 1105 puede particionar los datos de entrada en uno o más segmentos de información (que no se muestran). Por ejemplo, cuando los datos de entrada 1125 son datos de vídeo, el procesador puede particionar los datos de vídeo en bloques de datos, por ejemplo representando muestras de 16x16 de la imagen de vídeo correspondiente. Una
10 secuencia de eventos 1135 puede representar uno o más segmentos (bloques de vídeo).

El codificador de entropía 1110 recibe la secuencia de eventos 1135 que va a ser codificada aritméticamente, en el que el codificador aritmético 1115 genera 0 o más bits de la secuencia de información 1130 para cada caso de la secuencia de eventos 1135. El controlador 1120 es capaz de
15 regular el codificador de entropía 1110 al limitar el número de partes de información de la secuencia de información 1130 como una función del número de eventos de la secuencia de eventos 1135, de tal manera que el número de eventos no exceda un umbral con respecto al número de partes de información generadas. Cuando se supera el umbral, el controlador 1120 puede controlar el codificador de entropía para añadir una o más partes de información de relleno a la secuencia de información 1130 con el fin de
20 alcanzar el umbral.

El codificador de entropía puede codificar la secuencia de eventos 1135 de una manera similar a la descrita anteriormente con respecto al codificador aritmético 100 de la Figura 1. Por ejemplo, el
25 codificador aritmético 1115 puede incluir, en su caso, un secuenciador, y un estimador de probabilidad tal como se ha descrito anteriormente con respecto a la Figura 1. El codificador aritmético 1115 puede incluir además un motor central con Registros de Intervalo, Bajo y de Bits Pendientes. El controlador 1120 puede incluir uno o más contadores para el seguimiento de una serie de eventos de la secuencia de eventos 1135 procesada y una serie de partes de información de la secuencia de información generada 1130. Alternativamente, los uno o más contadores pueden estar presentes en el codificador de entropía 1110, donde el controlador 1120 es capaz de monitorizar el/los contador (es) para asegurar que se mantiene
30 sustancialmente el umbral entre eventos procesados y partes de información generadas. El controlador 1120 puede limitar el número de partes de información, por ejemplo, a lo largo de cada secuencia individual de eventos que están siendo codificados. Por ejemplo, el controlador puede garantizar que (# partes de información generadas) x (umbral) ≥ (# eventos procesados). Esto se puede lograr, por ejemplo, de una manera similar a como se ha descrito anteriormente con respecto a las Figuras 1-6, y, específicamente, la Figura 4, en que el controlador incrementa un contador para cada evento procesado, y resta un número del contador que representa el valor umbral, por ejemplo, "4" cuando se procesa un bit. A continuación, el controlador puede comparar el valor del contador, por ejemplo, con el valor 0, y desencadenar la inserción de un bit de relleno si el valor del contador es mayor que 0. De esta manera, el
40 controlador es capaz de asegurar que el codificador aritmético 1115 está limitado sustancialmente a una relación de 4:1 de eventos procesados por partes de información generadas.

Además, o como alternativa, el controlador puede limitar el número de eventos de la secuencia de eventos 1135 como una función del número de partes de información en la secuencia de partes de información, y un número de segmentos de los datos de entrada 1125 representado en la secuencia de eventos. Por ejemplo, la restricción puede tomar la forma de una combinación lineal:
45

$$e \leq \alpha B + \beta S,$$

50 donde
e es el número de eventos representados en la secuencia de partes de información,
B es un número de partes de información en la secuencia de partes de información,
S es un número de segmentos representados en la secuencia de eventos, y
α y β representan un valor de decremento para un contador utilizado por el controlador para mantener
55 sustancialmente una limitación del número de eventos de la secuencia de eventos con respecto a un número de partes de información generadas.

Los valores de α y β se proporcionan normalmente al controlador 1120, y la derivación de α y β se describirán más adelante. El valor de α puede representar un valor de decremento, por ejemplo, un
60 contador tras la generación de una parte de información en el codificador, en que el valor β puede representar el valor de decremento, por ejemplo, un contador tras la finalización de procesamiento un segmento. Como alternativa, el valor β puede ser decrementado desde un valor de contador en el inicio del proceso de un segmento, o en cualquier otro momento durante el proceso de un segmento, tal como resultaría evidente para un experto en la técnica.
65

Se puede utilizar una línea de control de relación 1140 para limitar el número de eventos de la secuencia de eventos 1135 como una función del número de partes de información en la secuencia de partes de información, y un número de segmentos de los datos de entrada 1125 representado en la secuencia de partes de información. Por ejemplo, la línea de control de velocidad 1140 se puede utilizar para transportar información relativa a la restricción de la complejidad (MEPS y / o PMEPS, ver más abajo) desde el codificador de entropía 1110 al procesador 1105. El codificador de entropía puede indicar al procesador que el umbral de limitación está a punto de ser, o ya ha sido excedido, por ejemplo, mediante la transmisión de información sobre el valor del/de los contador (es). De esta manera, el procesador 1105 puede reducir la relación de eventos (es decir, reducir el número de eventos por segmento / bloque.), por ejemplo, para reducir la probabilidad de que el controlador 1120 active la inserción de bits de relleno en la secuencia de información 1130.

En algunas circunstancias, el número determinado de segmentos de datos de los datos de entrada 1125 que serán procesados puede ser conocido en el codificador 1100, por ejemplo, cuando la secuencia de partes de información no tiene por qué limitarse sustancialmente para la transmisión a través de un medio de comunicación en particular. En estas circunstancias, la secuencia de partes de información generada en el codificador 1100 puede representar una imagen codificada completa para la transmisión generada a partir de un número conocido de segmentos de los datos de entrada 1125. Cuando el número específico de los segmentos de datos de los datos de entrada que serán procesados es conocido, el controlador puede, por ejemplo, limitar el número de eventos en la secuencia de eventos 1135 como:

$$e - \beta S \leq \alpha B,$$

Dado que el número total de segmentos, S, y el valor β son conocidos, el producto de $\beta \times S$ puede ser restado del número de eventos, e, para la secuencia de los eventos 1135 durante o después del proceso de uno o más segmentos de los datos de entrada 1125. Por ejemplo, cuando se utiliza un contador para limitar el número de eventos que responden a la cantidad de bits que se han generado, el contador puede inicialmente ser decrementado en un valor de $\beta \times S$, y puede ser decrementado en un valor α para cada parte de información generada, mientras que el contador se incrementa en "1" para cada evento de la secuencia de eventos 1135 procesados por el codificador de entropía 1110. El valor de β puede ser cualquier valor, habitualmente en el intervalo de 1 a 100, y puede ser determinado, por ejemplo, tal como se describe más adelante. El valor de α puede ser cualquier valor, habitualmente en el intervalo de 1 a 10, y puede ser determinado, por ejemplo, tal como se describe más adelante.

En algunas circunstancias, el número de segmentos de los datos de entrada 1125 que se van a procesar no se conoce por adelantado, por ejemplo, cuando el medio de comunicación limita el número de partes de información que pueden ser proporcionadas en la secuencia de información 1130. Esto puede ocurrir, por ejemplo, cuando la secuencia de información 1130 va a ser transmitida a través de Internet, como un paquete de Protocolo de Internet (IP), en que el paquete de IP tiene una limitación de tamaño máximo. En estas circunstancias, dependiendo de la complejidad de una imagen en particular, pueden ser necesarias una o más secuencias de partes de información para representar una sola imagen de los datos de entrada 1125. Sin embargo, el número de segmentos utilizados para la generación de una secuencia de partes de información no se puede saber de antemano, ya que no se puede saber después de cuántos segmentos procesados se alcanza el tamaño máximo de una secuencia de partes de información. Cuando el número de segmentos de los datos de entrada 1125 que se van a procesar no se conoce de antemano, el controlador puede dar cuenta de la pluralidad de secuencias de eventos como el uno o más segmentos que representan una secuencia particular de los eventos que se codifican. Por ejemplo, cuando se utiliza un contador para limitar el número de eventos que responde a la cantidad de bits que han sido generados, el contador puede ser decrementado en un valor β para cada segmento procesado, y puede ser decrementado en un valor α para cada parte de información generada, mientras que el contador puede ser incrementado en "1" para cada caso de la secuencia de eventos 1135 procesados por el codificador de entropía 1110.

Los valores para α y β se pueden determinar, por ejemplo, teniendo en cuenta las limitaciones impuestas por un estándar particular (es decir, un estándar que rige la codificación de vídeo), y / o limitaciones con respecto a un decodificador que se utiliza para decodificar la secuencia de información codificada por el codificador 1100. Estas limitaciones impuestas por el estándar particular puede incluir información con respecto a un número máximo de bits por segundo (Mbps), un número máximo de segmentos (bloques) por segundo (MSPS), y un número máximo pico de bits por segundo (PMBPS). El PMBPS puede representar, por ejemplo, un número máximo de bits de una imagen dividido por la distancia temporal entre dos imágenes a codificar.

Los MBPS pueden indicar la capacidad del canal, por ejemplo, aproximadamente 64 000 bits por segundo para aplicaciones de comunicaciones móviles, 500 000 bits por segundo para la línea de abonado digital (DSL), y 4 000 000 bits por segundo para aplicaciones de difusión como la televisión (TV). Los MSPS pueden indicar el estándar de formato de vídeo, por ejemplo, 176x144 píxeles a 10 o 15 fotogramas por

ES 2 559 406 T3

segundo para aplicaciones móviles (es decir, telefonía celular), 320x240 píxeles a 24 fotogramas por segundo para aplicaciones de DSL, 720x480 píxeles a 30 fotogramas por segundo para aplicaciones de televisión estándar, y 1920x1088 píxeles a 30 fotogramas por segundo para TV de Alta Definición. PMBPS puede ser cualquier múltiplo de los MBPS, habitualmente varias veces MBPS, por ejemplo, 5 PMBPS \approx 10 x MBPS.

Dependiendo de la implementación particular, los recursos de un decodificador (es decir, las capacidades de procesamiento y / o la memoria intermedia disponible) pueden ser limitados, y pueden también tomarse en cuenta al determinar los valores de α y β . Estas limitaciones pueden incluir valores reales o 10 supuestos con un número máximo de eventos por segundo (MEPS) que pueden ser procesados por el decodificador, y / o un número máximo pico de eventos por segundo (PMEPS) que pueden ser procesados por el decodificador (es decir, un número máximo de eventos en una imagen dividida por la distancia temporal entre dos imágenes a descodificar).

15 La explicación de una o más de las anteriores limitaciones del decodificador, de α y β puede ser representada en un sistema lineal como:

$$\text{PMEPS} = \alpha \cdot \text{PMBPS} + \beta \cdot \text{MSPS},$$

20 y

$$\text{MEPS} = \alpha \cdot \text{MBPS} + \beta \cdot \text{MSPS}.$$

25 Los valores de α y β pueden determinarse resolviendo el sistema lineal para α y β , dadas por ejemplo, las limitaciones de PMBPS y de MSPS fijadas por el estándar particular, y una o las dos limitaciones de PMEPS y MEPS del decodificador. Los valores de α y β no necesitan ser valores enteros, sino que más bien pueden ser cualquier valor para satisfacer sustancialmente el sistema lineal.

30 Los valores de α y β pueden ser determinados de antemano por un diseñador de sistema del codificador 1100 que representan una o más de las limitaciones mencionadas anteriormente, y que se proporcionan al controlador 1120. Alternativamente, o además, los valores de α y β pueden ser determinados por el controlador 1120, o por cualquier otro componente del codificador 1100, de acuerdo con una o más de las limitaciones descritas anteriormente, o como valores por defecto del codificador 1100. Cuando el controlador 1120 determina los valores para α y β utilizando una o las dos limitaciones impuestas por el estándar o por un dispositivo de decodificación, la información relativa a una o más de las limitaciones se 35 puede almacenar en una memoria (que no se muestra) del controlador 1120, y es utilizada por el controlador 1120 en la determinación de los valores de α y β . Además, o como alternativa, se puede proporcionar información relativa a las limitaciones al controlador 1120, por ejemplo, mediante algún dispositivo externo, como una memoria externa (es decir, un Disco de Vídeo Digital (DVD)), un dispositivo 40 reproductor de DVD, o a través de un ingeniero de sistemas que, por ejemplo, manipule algunas de las funciones en lo que se refiere a la codificación de los datos de entrada particulares 1125. En este último caso, el ingeniero de sistemas puede introducir en una consola u otro dispositivo de entrada (que no se muestra), o de alguna otra manera especificar, información con respecto a las limitaciones impuestas como resultado de un estándar de codificación y / o un dispositivo de decodificación, tal como sería 45 apreciado por un experto en la técnica.

Cuando no se conocen una o las dos limitaciones PMEPS y MEPS del decodificador, se pueden suponer valores para las PMEPS y MEPS, como por ejemplo los requisitos operativos mínimos que debe poseer un decodificador para ser capaz de decodificar la secuencia de información 1130 generada por el 50 codificador 1100. Además, cuando se asumen los valores de las PMEPS y / o MEPS, pueden tenerse en cuenta las consideraciones en cuanto a, por ejemplo, el porcentaje de tiempo que utilizará el decodificador para la decodificación aritmética de los datos. Por ejemplo, se puede suponer que el decodificador será capaz de gastar no más de 30% de su tiempo / capacidades de procesamiento para llevar a cabo la decodificación aritmética de una secuencia de información recibida en el decodificador. 55

Además, aunque los valores de α y β se pueden determinar utilizando el sistema lineal anterior, dichos valores pueden ser ajustados para lograr un rendimiento de codificación deseado, en que los valores ajustados para α y β pueden ser utilizados posteriormente por el codificador 1100 para codificar los datos de entrada 1125 en la secuencia de información 1130. Por ejemplo, los valores de α y β se pueden ajustar 60 con las consideraciones en cuanto a una posible pérdida de la calidad de vídeo debido a las partes de información de relleno en la secuencia de información 1130. Puede ser deseable, por ejemplo, reducir al mínimo el número de partes de información de relleno con el fin de mantener una óptima calidad de vídeo (es decir, de vídeo con poca o ninguna distorsión), a la vez que se limita la complejidad de decodificación. Una compensación de este tipo entre la complejidad de decodificación y la calidad de vídeo puede 65 determinarse experimentalmente, y puede tener en cuenta el algoritmo de codificación de vídeo específico utilizado por el codificador 1100.

- Además, cuando se determinan los valores de α y β se pueden realizar consideraciones en cuanto a si la restricción de la complejidad es demasiado ajustada, por ejemplo, si los valores de α y β son demasiado bajos. Una alta proporción partes de información de relleno en la secuencia de partes de información (es decir, una serie de partes de relleno superior a aproximadamente un 1% o un 2% de las partes de información de la secuencia de información) puede indicar que la limitación es demasiado ajustada. Un experto se daría cuenta de que otras proporciones pueden indicar una alta proporción de partes de información de relleno, por ejemplo, teniendo en cuenta el estándar y / o el decodificador específicos que pueden ser utilizados.
- 10 Cuando se determina, por ejemplo, que los valores de α y β son demasiado ajustados, se puede aumentar los valores para α y β con el fin de reducir la probabilidad que se añadan bits de relleno (es decir, reducir la probabilidad de una afectación de la calidad en la secuencia de información codificada). Al aumentar los valores para α y β se pueden hacer consideraciones en cuanto al efecto sobre los límites de la complejidad resultante (es decir, MEPS y / o PMEPS) con respecto a un decodificador que se utilizará para decodificar la secuencia de información codificada. Estas consideraciones pueden incluir el costo de implementar el decodificador. Si el límite de complejidad es mayor, más potencia de procesamiento puede ser necesaria en el decodificador. Un aumento de la potencia de procesamiento requerida probablemente podría tener como resultado un mayor costo de implementación.
- 15 Como alternativa, se puede determinar α y β experimentalmente, utilizando técnicas de regresión lineal. Se puede codificar una serie de secuencias de eventos, cada uno en representación de los segmentos S, sin imponer ninguna limitación de complejidad. Para cada secuencia z de eventos, se conoce el número de partes de información generadas resultante B (z) para el número de eventos e (z). Se puede determinar, utilizando regresión lineal, una línea $e = c*B + d$ que se aproxima a los pares de datos (e (z), B (z)). Un valor inicial de α puede ser indicado por c, y un valor inicial de β puede ser indicado por d / S. Los valores de α y β pueden entonces ser aumentados con el fin de minimizar el número de pares de datos (e (z), B (z)) que se encuentran por encima de la línea $e = \alpha*B + \beta*S$. La cantidad en la cual α y β se incrementan probablemente también tendrá en cuenta los valores resultantes de MEPS y / o PMEPS, por ejemplo, para controlar los costos de implementación de un decodificador.
- 20 Utilizando los valores de α y β de acuerdo con lo determinado por una o más de las diversas técnicas descritas anteriormente, el codificador 1100 puede dar cuenta de un valor de α (es decir, disminuir un contador en el valor de α) para cada parte de información generada, y puede dar cuenta de un valor de β (es decir, disminuir un contador en el valor de β) a la finalización de un segmento de los datos de entrada 1125. Por ejemplo, cuando α y β son valores enteros, dicha cuenta (es decir, los decrementos en uno o más contadores) puede llevarse a cabo directamente.
- 25 Cuando, por ejemplo, uno de ellos o tanto α como β son valores fraccionarios, se puede determinar un denominador común para proporcionar valores no fraccionarios para α y β . En esta circunstancia, los valores nuevos, no fraccionarios para α y β pueden ser contabilizados tal como se ha descrito anteriormente, por ejemplo, decrementando un contador por los valores de α y β en cada generación de pieza de información y finalización de segmento de procesamiento, respectivamente. El denominador común determinado puede explicarse, por ejemplo, mediante la adición del valor del denominador común al valor de contador en el procesamiento de cada evento de la secuencia de eventos 1135. Por ejemplo, cuando los valores para A y 3 se determinan en 4/3 y 25, respectivamente, se puede determinar un denominador común como 3. Por lo tanto, los valores no-fraccionarios para α y β se pueden determinar como 4 y 75 respectivamente, utilizando el denominador común. Así pues, cuando se utiliza un contador para dar cuenta de los valores de α y β , el contador puede ser decrementado en 4 para cada pieza de información generada, decrementado en 75 tras la finalización del procesamiento de cada segmento, e incrementado en 3 para cada evento procesado.
- 30 Tener un codificador capaz de limitar un número de partes de información a través de uno o más segmentos de datos de entrada puede permitir que los datos de vídeo sean decodificados de forma consistente, lo que representa un número máximo de partes de información por grupo de segmentos, manteniendo la calidad de la codificación de vídeo en comparación con los codificadores de la técnica anterior. Además, cuando a través de la línea de control de la velocidad el codificador recibe una señal de que el umbral de complejidad está a punto de ser superado, el codificador puede generar menos eventos, por ejemplo β o menos eventos a partir del siguiente segmento de los datos de entrada, con el fin de evitar que se sobrepase ese umbral para ese próximo segmento / bloque. Por lo tanto, tener un valor de β distinto de cero puede ayudar al procesador 1105 en el ajuste de calidad de vídeo. Además, dado que el controlador normalmente desencadena la adición de bits de relleno cuando se supera el umbral de limitación, el procesador puede ser implementado de una manera en la que el procesador no tiene por qué tener en cuenta, o tratar, la restricción de la complejidad. Por lo tanto, si el procesador cumple los límites impuestos por el estándar en particular, por ejemplo, los límites en relación con uno o más entre MSPS, MBPS y PMBPS, a la vez también cumplirá con el límite de complejidad MEPS y / o PMEPS (es decir, derivados de α y β tal como se ha descrito anteriormente).
- 35
- 40
- 45
- 50
- 55
- 60
- 65

Aunque la limitación del número de eventos que responden a la cantidad de partes de información generadas se ha descrito con respecto al codificador 1100, un experto se dará cuenta de que una restricción similar del número de eventos de la secuencia de eventos que responde al número de partes de información en la secuencia de información puede tener su explicación en el codificador 100 descrito anteriormente con respecto a la Figura 1. Además, aunque el controlador 1120 se ha descrito como acoplado al codificador aritmético 1115, un experto se dará cuenta de que dicho acoplamiento puede ser directo, o a través de otros dispositivos intermedios. Dicho acoplamiento permite la comunicación directa y / o indirecta entre el controlador 1120 y el codificador aritmético 1115, de tal manera que el controlador 1120 es capaz de regular el codificador de entropía 1110 al limitar el número de partes de información de la secuencia de información 1130 como una función del número de los eventos de la secuencia de eventos 1135.

Los codificadores 100 y 1100, y el decodificador 700, descritos en este documento pueden estar compuestos completamente de hardware, como software que se ejecuta en un microprocesador adecuado, o como una combinación de hardware o software. La funcionalidad de los motores centrales 115 y 715, así como el codificador aritmético 1115, el controlador 1120 y el procesador 1105, puede llevarse a cabo como un programa de ordenador que se ejecuta en un microprocesador adecuado, y los registros contadores 180 y 775 pueden ser un registro que reside dentro del microprocesador, o que puede residir en algún medio de almacenamiento externo en comunicación con el microprocesador, por ejemplo, RAM, EEPROM, o cualquier medio de almacenamiento magnético, óptico o de otro tipo regrabable tal como sería apreciado por un experto en la técnica. Tal como se ha descrito anteriormente, aunque los codificadores 100 y 1100, y el decodificador 700, han sido descritos como que procesan una secuencia de eventos binaria a una secuencia de información binaria en el caso de los codificadores, y una secuencia de información binaria a una secuencia de evento binario en el caso del decodificador, un experto en la técnica se daría cuenta de que, utilizando las enseñanzas descritas en el presente documento, la secuencia de eventos M-arios puede ser procesada y producida sin dejar de lograr las ventajas descritas en el presente documento.

Además, la señal de los registros contadores 180 y 775 puede invertirse sin dejar de lograr las ventajas descritas en el presente documento, en que los incrementos en el registro contador se describen en los pasos 410 y 1005 serían cambiados a decrementos, los decrementos en el registro contador que se describen en los pasos 435 y 1020 serían cambiados a incrementos, y las comparaciones con el registro contador en los pasos 430 y 1010 se cambiarían para determinar si el valor del respectivo registro contador es > cero.

Adicionalmente, se pueden utilizar dos contadores en lugar de uno. El primer contador C1 puede contar eventos, y el segundo contador C2 puede contar bits. En tal caso, una comparación entre los dos contadores puede ser si $C1 < 4 * C2$ para determinar si (o explicar que) la relación predeterminada de los eventos por bits se mantiene sustancialmente.

Además, aunque el codificador 100 y el decodificador 700 se han descrito como hardware separado, un experto en la técnica se daría cuenta de que la funcionalidad llevada a cabo por cada uno de ellos tal como se describe en el presente documento podría integrarse en una sola unidad, por ejemplo, que puede ser útil en el caso de un ordenador personal (PC) que se utiliza para una conferencia de vídeo, en que se utilizaría el procesador y el almacenamiento de medios del PC para llevar a cabo la funcionalidad del codificador y el decodificador descritos en el presente documento .

La relación limitada de eventos por partes de información y la explicación de la relación limitada de eventos por partes de información realizada por los motores centrales descritos en el presente documento puede llevarse a cabo en otros diseños de motores centrales, incluyendo el de un codificador Q, un codificador MQ, un codificador Z, así como los descritos en "Low-Complexity Arithmetic Implementation", René van der Vleuten, Documento JVT- B033, reunión JVT en Ginebra, Suiza, enero de 2002, y "Low-Complexity Arithmetic Coding Engine", Winger, Documento JVT-B036, JVT", reunión en Ginebra, Suiza, enero de 2002, ambos incorporados como referencia en el presente documento.

Por lo tanto, se ha descrito un codificador con un límite de complejidad reducido que es capaz de codificar una secuencia de eventos a una secuencia de información limitando la proporción de eventos por partes de información. Dicho codificador resulta ventajoso ya que proporciona una secuencia de información que puede ser decodificada y utilizada para soportar la prestación de la información para su uso en tiempo real, sin afectar significativamente la eficiencia de codificación de la secuencia de información. Además, una codificación de complejidad reducida de este tipo puede ser beneficiosa cuando el decodificador utilizado en la decodificación de la secuencia de la información posee unas capacidades de procesamiento reducidas (es decir, se encuentra en un dispositivo móvil) ya que un decodificador de este tipo sería capaz de decodificar la secuencia de información sin ningún retraso indebido a la espera de que se lleve a cabo la decodificación. Además, se ha descrito un codificador capaz de limitar un número de partes de información en uno o más segmentos de los datos de entrada, que puede permitir que los datos de vídeo sean decodificados de forma consistente, lo que representa un número máximo de partes de

5 información por grupo de segmentos, a la vez que se mantiene la calidad de codificación de vídeo, en comparación con los codificadores de la técnica anterior. Cuando el codificador recibe una señal a través de la línea de control de velocidad de que el umbral de complejidad está a punto de ser superado, el codificador puede generar menos eventos, con el fin de evitar superar el umbral para este próximo segmento de los datos de entrada. Además, dado que el controlador normalmente desencadena la adición de bits de relleno cuando se supera el umbral de limitación, el procesador puede ser implementado de una manera tal que el procesador no tenga por qué tener en cuenta o tratar la restricción de la complejidad. Además, se describe un decodificador capaz de decodificar una secuencia de información a una secuencia de eventos explicando una relación limitada de eventos por partes de información en la secuencia de información, permitiendo de esta manera una decodificación eficiente de la secuencia de información para su utilización en, por ejemplo, el uso en tiempo real de la secuencia de eventos.

10 Aunque esta invención se ha mostrado y descrito en conexión con realizaciones particulares, es evidente que, además de los mencionados anteriormente, pueden llevarse a cabo ciertos cambios y modificaciones de las características básicas de esta invención. Además, pueden existir muchos tipos diferentes de software y hardware que se pueden utilizar para la práctica de la invención, y la invención no se limita a los ejemplos descritos anteriormente. Por consiguiente, las realizaciones descritas han de ser consideradas en todos los aspectos sólo como ilustrativas y no restrictivas, y por lo tanto el alcance de la invención está indicado por las reivindicaciones adjuntas más que por la descripción anterior.

Reivindicaciones

- 5 1. Un codificador de entropía (1100) para convertir al menos una secuencia de eventos, en que cada secuencia de eventos incluye una pluralidad de eventos, en al menos una secuencia de información, en que cada secuencia de información incluye al menos una parte de información, en que la al menos una secuencia de eventos representa datos de entrada particionados en un número de segmentos, que comprende:
- 10 • un codificador aritmético (1115) configurado para generar cero o más partes de información de al menos una secuencia de información codificando al menos un evento de la al menos una secuencia de eventos; y
- 15 • un controlador (1120) acoplado con el codificador aritmético (1115) configurado para mantener una limitación del número de eventos de la secuencia de eventos con respecto a un número de partes de información generadas en que la limitación incluye el número de eventos en la al menos una secuencia de eventos para que sea igual o menor que un número determinado por una función de una combinación lineal que incluye el número de partes de información generadas en al menos una secuencia de información y el número de segmentos
- 20 2. El codificador de entropía (1100) de la reivindicación 1, en el que el controlador (1120) también está configurado para limitar el número de eventos de al menos una secuencia de eventos al número de partes de información de al menos una secuencia de información generando cero o más partes de relleno en al menos una secuencia de información.
- 25 3. El codificador de entropía (1100) de la reivindicación 1 en que los datos de entrada son datos de vídeo.
- 30 4. El codificador de entropía (1100) de la reivindicación 3, en que cada segmento comprende un bloque de muestras de 16 por 16 de los datos de entrada.
- 35 5. El codificador de entropía (1100) de la reivindicación 1 en que al menos una de las partes de información y los eventos son M-arios en su naturaleza.
- 40 6. Un aparato de codificación que comprende el codificador de entropía (1100) de la reivindicación 1, que comprende además:
- un procesador configurado para convertir los datos de entrada en la al menos una secuencia de eventos.
- 45 7. El aparato de codificación de la reivindicación 6 en que el controlador también está configurado para limitar el número de eventos de la al menos una secuencia de eventos al número de partes de información de la al menos una secuencia de información generando cero o más partes de relleno en la al menos una secuencia de información.
- 50 8. El aparato de codificación de la reivindicación 6, en que el procesador también está configurado para particionar los datos de entrada en el número de segmentos.
- 55 9. El aparato de codificación de la reivindicación 8, en que el codificador aritmético y el controlador comprenden un codificador de entropía, en que el codificador de entropía está acoplado con el procesador, y está configurado para enviar información relativa al número de partes de información generadas en la al menos una secuencia de información al procesador.
- 60 10. El aparato de codificación de la reivindicación 9, en que el procesador está configurado para reducir una relación de eventos de la al menos una secuencia de eventos que responde a la información recibida del codificador de entropía.
- 65 11. El aparato de codificación de la reivindicación 9, en que los datos de entrada son datos de video.
12. El aparato de codificación de la reivindicación 11 en que cada segmento comprende un bloque de muestras de 16 x 16.
13. El aparato de codificación de la reivindicación 8, en que el controlador también está configurado para limitar el número máximo de eventos generando cero o más partes de relleno en la al menos una secuencia de información.
14. El aparato de codificación de la reivindicación 6, en que al menos una de las partes de información y los eventos son M-arios en su naturaleza.

- 5 15. Un decodificador de entropía para convertir al menos una secuencia de eventos, en que cada secuencia de eventos incluye una pluralidad de eventos, en al menos una secuencia de información, en que cada secuencia de información incluye al menos una parte de información, en que la al menos una secuencia de eventos representa datos de entrada particionados en un número de segmentos, que comprenden:
- 10 • pasos implementados por ordenador realizados por un procesador de un sistema de ordenador para implementar:
 - 15 • generar cero o más partes de información de al menos una secuencia de información codificando al menos un evento de al menos una secuencia de eventos de acuerdo con una codificación aritmética; y
 - mantener una limitación del número de eventos de la secuencia de eventos con respecto a un número de partes de información generadas en que la limitación incluye el número de eventos en al menos una secuencia de eventos para que sea igual o inferior a un número determinado por una función de una combinación lineal que incluye el número de partes de información generadas en al menos una secuencia de información y el número de segmentos.

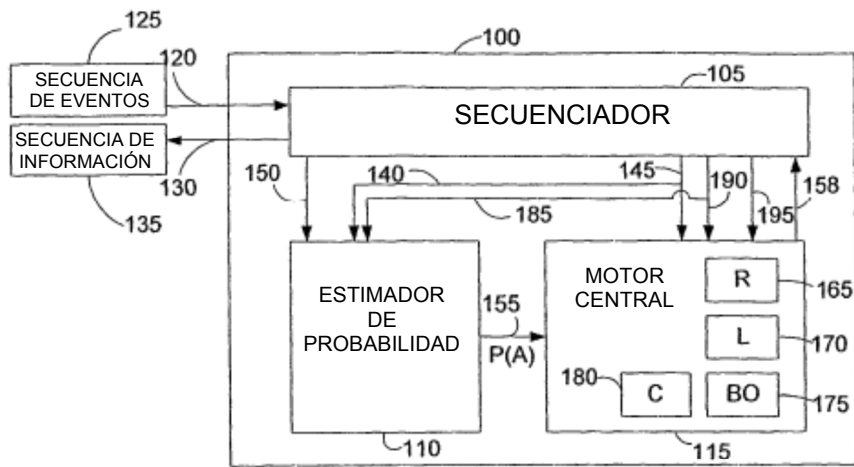


FIGURA 1

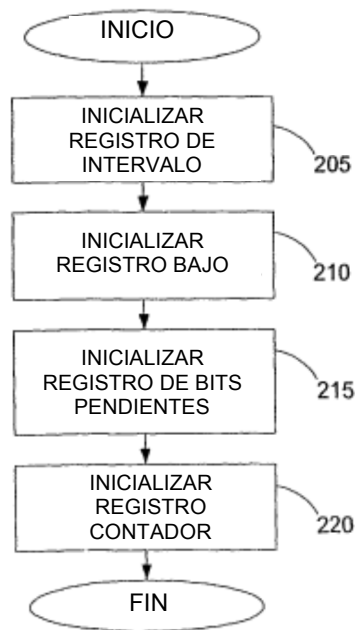


FIGURA 2

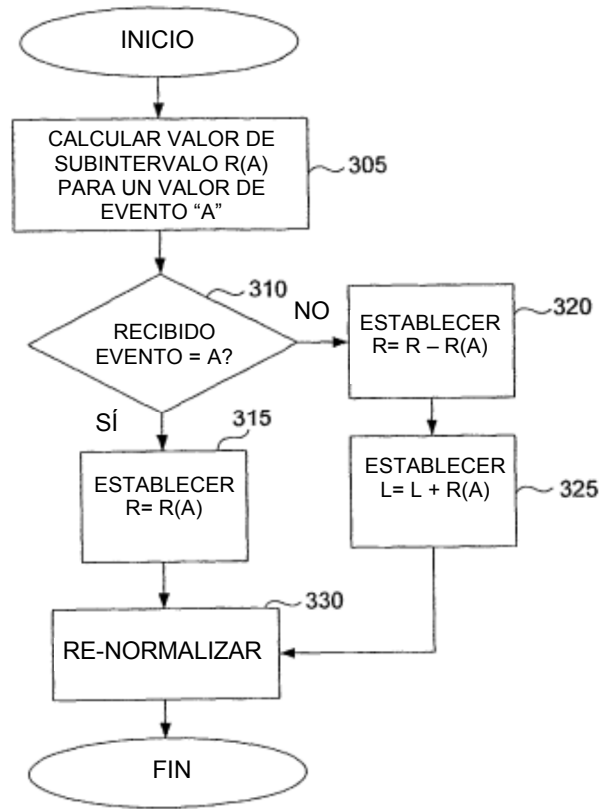


FIGURA 3

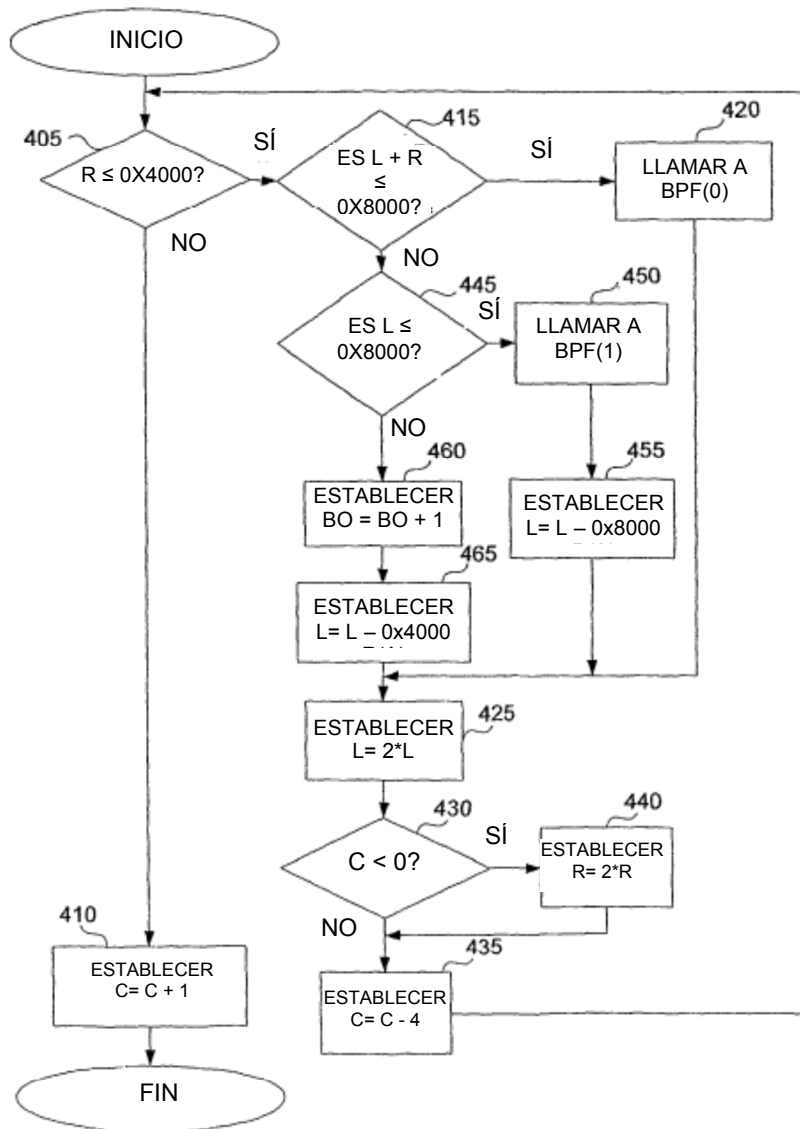


FIGURA 4

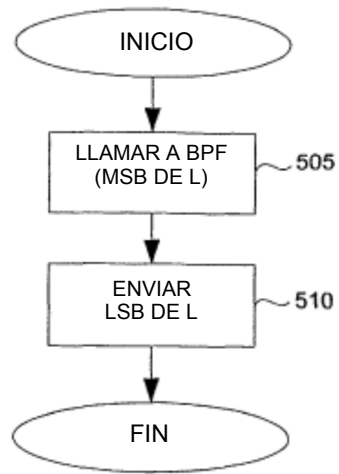


FIGURA 5

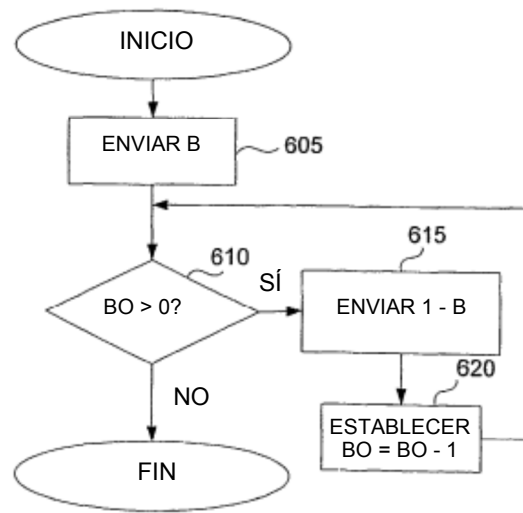


FIGURA 6

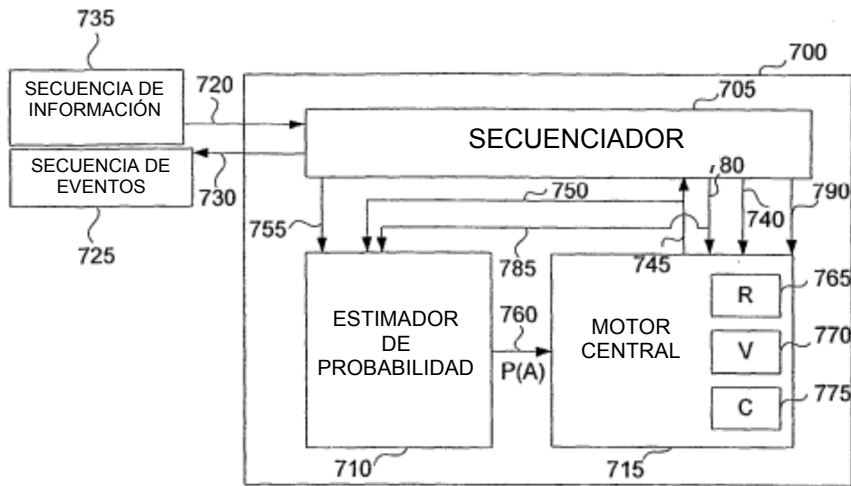


FIGURA 7

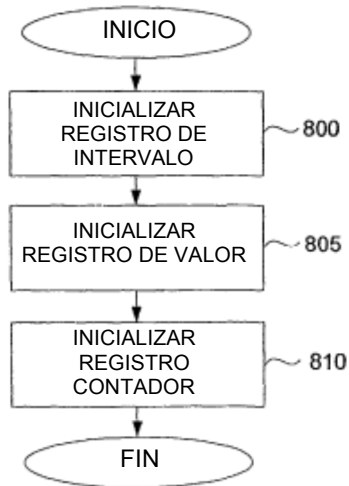


FIGURA 8

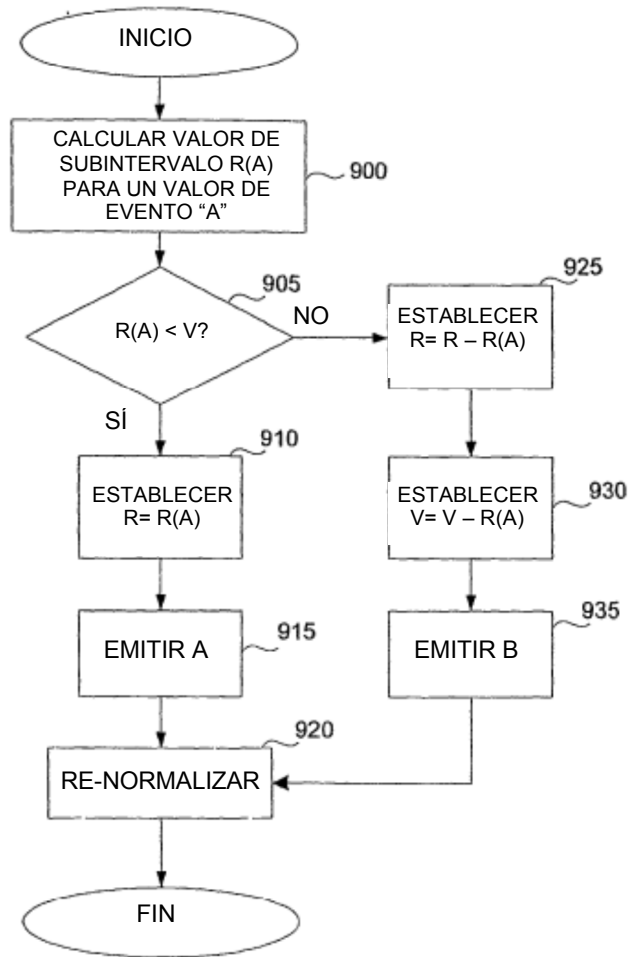


FIGURA 9

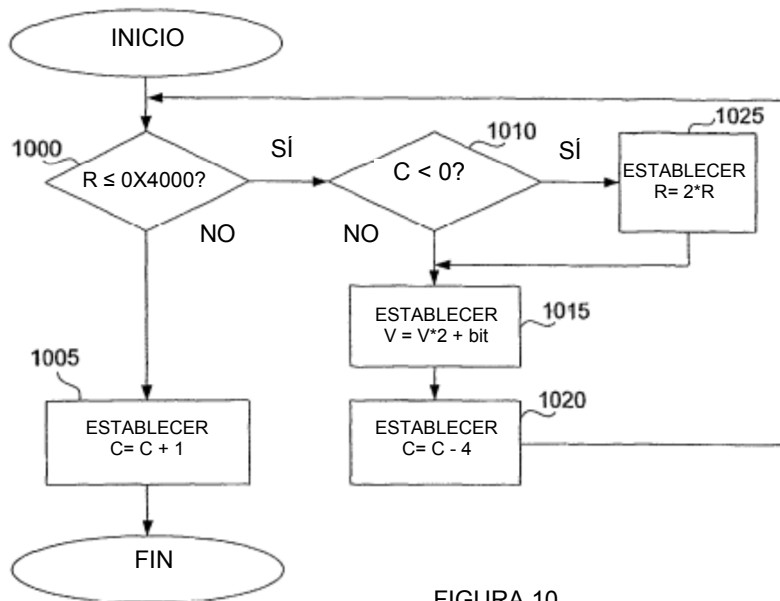


FIGURA 10

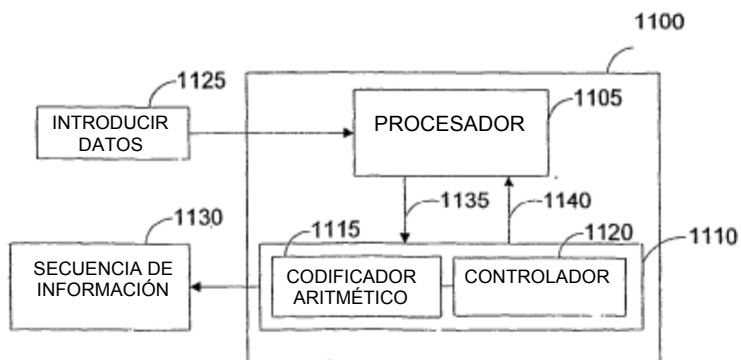


FIGURA 11