



19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA

11 Número de publicación: **2 360 786**

51 Int. Cl.:

**H04N 7/24** (2006.01)

**G06T 9/00** (2006.01)

**H04N 7/26** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **02258174 .8**

96 Fecha de presentación : **27.11.2002**

97 Número de publicación de la solicitud: **1322120**

97 Fecha de publicación de la solicitud: **25.06.2003**

54

Título: **Codificación y decodificación de datos de valores clave de un modo interpolador de orientación.**

30

Prioridad: **27.11.2001 US 333130 P**  
**03.12.2001 US 334541 P**  
**26.12.2001 US 342101 P**  
**04.04.2002 US 369597 P**  
**13.11.2002 KR 10-2002-0070496**

45

Fecha de publicación de la mención BOPI:  
**09.06.2011**

45

Fecha de la publicación del folleto de la patente:  
**09.06.2011**

73

Titular/es: **SAMSUNG ELECTRONICS Co., Ltd.**  
**416, Maetan-dong**  
**Paldal-gu, Suwon-city, Kyungki-do, KR**

72

Inventor/es: **Kim, Do-Kyoon;**  
**Jung, Seok-Yoon;**  
**Jang, Euee-Seon;**  
**Woo, Sang-Oak;**  
**Lee, Shin-Jun;**  
**Han, Mahn-Jin y**  
**Jang, Gyeong-Ja**

74

Agente: **Carpintero López, Mario**

ES 2 360 786 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 y decodificación de datos de valores clave de un nodo interpolador de orientación.

5 La presente invención versa acerca de un procedimiento y un aparato para codificar y decodificar datos de valores clave de una animación y, más en particular, acerca de un procedimiento y un aparato para codificar y decodificar datos de valores clave de un nodo interpolador de orientación, que se convierten en un cuaternión codificando y decodificando diferentes entre valores convertidos por orientación aplicados a un objeto en cada trama clave.

El MPEG-4 BIFS, que es uno de los estándares multimedia internacionales, soporta una animación basada en tramas clave usando un nodo interpolador que tiene claves y valores clave de una animación.

10 Para representación una animación de una manera tan natural y fluida como sea posible usando una técnica de animación basada en tramas clave, se requiere un número considerable de claves y una cantidad considerable de datos de valores clave, y los datos de los campos entre las tramas clave se rellenan mediante interpolación. En un lenguaje para el modelado de la realidad virtual (VRLM), la interpolación implica una interpolación lineal o esférica.

15 Las claves y los valores clave se aproximan a una curva original de animación en un eje temporal. La FIG. 1 es un diagrama que ilustra trayectorias bidimensionales de datos de animación, representados por un nodo interpolador, según el transcurso de tiempo sobre la superficie de una esfera tridimensional. Tal como se muestra en la FIG. 1, el MPEG-4 BIFS convencional soporta la interpolación lineal esférica entre tramas clave, y una trayectoria de animación parece similar a un conjunto de segmentos que representa la variación de los datos de animación.

20 En un nodo interpolador de orientación proporcionado por BIFS, los datos clave indica un instante de tiempo predeterminado en un eje temporal en el que se sitúa una animación usando número discontinuos entre  $-\infty$  e  $\infty$ . Los datos de valores clave representan información sobre la rotación de un objeto en una imagen sintética en un instante de tiempo predeterminado indicada por datos clave. La información sobre la rotación del objeto en otro instante de tiempo predeterminado, que no está representada por datos clave, se obtiene usando datos clave correspondientes a dos instantes de tiempo que son los más contiguos al instante de tiempo predeterminado, mediante interpolación lineal esférica.

25 En la interpolación lineal esférica, la información de rotación se representa mediante un eje de rotación y un ángulo de rotación. El MPEG-4 BIFS, como el lenguaje para el modelado de la realidad virtual (VRLM), soporta la información de rotación representada por un eje de rotación y un ángulo de rotación usando un nodo interpolador de orientación. Cuando se genera una animación fluida usando datos de valores clave en una interpolación lineal esférica, los valores diferenciales de los datos de valores clave entre tramas clave están muy correlacionados entre sí, lo que causa redundancia entre los datos. En consecuencia, es efectivo usar un procedimiento para codificar datos de valores clave usando valores diferenciales de los datos.

30 El MPEG-4 BIFS proporciona dos procedimientos diferentes para codificar datos de los campos representados por claves y datos de valores clave de un nodo interpolador de orientación. Uno es un procedimiento para codificar datos de los campos usando modulación de impulsos codificados (MIC) y el otro es un procedimiento para codificar datos de los campos usando modulación diferencial de impulsos codificados (DPCM) y codificación por entropía.

35 En el procedimiento para codificar datos de los campos usando MIC, solo se lleva a cabo un proceso de cuantificación con los datos clave y los datos de valores clave que han de ser codificados. Dado que en este procedimiento no se consideran las características de los datos que han de ser codificados, se considera que este procedimiento es poco efectivo. En el procedimiento para codificar datos de los campos usando MIC, se introducen los datos de los campos de un nodo interpolador de orientación y se convierten los datos de valores clave de los datos de los campos en valores en un espacio de cuaterniones. A continuación, se cuantifican las claves y los datos de valores clave. Se da salida a los datos de los campos cuantificados en forma de datos binarios. Para medir el grado en el que los resultados de la transformación por cuaterniones están distorsionados visualmente en comparación con los datos originales de los campos, se restauran los datos binarios en los datos de valores clave consistentes en un eje de rotación y un ángulo de rotación. Los datos restaurados de los campos de un nodo interpolador de orientación se almacenan y, a continuación, se les da salida a la pantalla. Es posible medir el grado de distorsión visual de las imágenes causada por un error de cuaternión en el uso de los datos restaurados. La distorsión de las imágenes puede calcularse con la Ecuación (1) siguiente.

$$D = \sqrt{\left(\frac{\sum_{i=0}^{i=N} \varepsilon_i}{N}\right)^2} = \sqrt{\left(\frac{\sum_{i=0}^{i=N} Q_i - \hat{Q}_i}{N}\right)^2} \quad \dots(1)$$

40 En la Ecuación (1), N representa el número de datos de los campos, y  $\varepsilon_i$  representa un valor diferencial entre los datos de valores clave codificados  $Q_i$  y datos de valores clave  $\hat{Q}_i$  restaurados en un espacio de cuaterniones.

Por otra parte, en el procedimiento para codificar datos de los campos usando DPCM y codificación por entropía, se considera una correlación entre los datos sucesivos, y, por ello, se considera que este procedimiento es más efectivo que el procedimiento para codificar datos de los campos usando MIC en términos de eficiencia de la codificación. En este procedimiento, antes de un proceso de cuantificación, se calcula un valor diferencial entre datos de valores clave restaurados previamente y datos de valores clave que han de ser codificados, y, a continuación, se cuantifica el valor diferencial, mejorando así la eficiencia de codificación aprovechando las características de los datos mostrados en el valor diferencial.

Las FIGURAS 2A y 2B son diagramas de bloque de un codificador MPEG-4 PMFC que usa DPCM lineal y codificación por entropía, y de un decodificador MPEG-4 PMFC que usa DPCM lineal inversa y decodificación por entropía, respectivamente. Un operador de DPCM lineal mostrado en la FIG 2A calcula datos diferenciales  $\hat{Q}_i$  entre datos de valores clave actuales y datos de valores clave restaurados previamente según la Ecuación (2).

$$Q_i = Q_i - \hat{Q}_{i-1} = (q_{i,0} - \hat{q}_{i-1,0}, q_{i,1} - \hat{q}_{i-1,1}, q_{i,2} - \hat{q}_{i-1,2}, q_{i,3} - \hat{q}_{i-1,3}) \quad \dots(2)$$

En la Ecuación (2),  $Q_i$  representa datos de valores clave originales en un instante (t) de tiempo predeterminado, que están representados por cuaternión, y  $\hat{Q}_{i-1}$  representa datos de valores clave en un instante (t-1) de tiempo predeterminado, que se restauran desde un circuito de compensación de errores.

Sin embargo, el procedimiento de codificación llevado a cabo en el aparato para codificar datos de valores clave mostrado en la FIG. 2A no tiene una eficiencia de codificación elevada. Es posible imaginarse fácilmente cuáles son las desventajas del procedimiento de codificación analizando los datos de valores clave, que determinan la rotación de un objeto en un espacio de cuaterniones. Los datos de valores clave están representados por un cuaternión en la siguiente ecuación.

$$Q = \left( \cos \frac{\theta}{2}, \frac{n_x}{\|n\|} \operatorname{sen} \frac{\theta}{2}, \frac{n_y}{\|n\|} \operatorname{sen} \frac{\theta}{2}, \frac{n_z}{\|n\|} \operatorname{sen} \frac{\theta}{2} \right) \quad \dots(3)$$

Por ejemplo, cuando los componentes de un cuaternión tienen los mismos valores absolutos que sus componentes correspondientes de otro cuaternión, pero signos diferentes en el espacio de cuaterniones, como se muestra en la Ecuación (3), se considera que los dos cuaterniones son iguales. En otras palabras, los dos cuaterniones proporcionan los mismos efectos en términos de la transformación rotacional de un objeto en un espacio tridimensional, lo que significa que los factores que afectan a la transformación rotacional de un objeto son una dirección de un eje de rotación y un ángulo de rotación, no el vector del eje de rotación. Sin embargo, como en el MPEG-4 BIFS, si los datos de valores clave están representados por un cuaternión usando la Ecuación (3) y se calcula linealmente un valor diferencial calculando diferencias en los vectores entre datos de valores clave sucesivos, el valor diferencial no es 0, lo que significa que los valores diferenciales lineales no reflejan bien la redundancia en la transformación rotacional. En consecuencia, es imposible medir con precisión la calidad de las imágenes usando el procedimiento para medir el grado de distorsión de las imágenes mostrado en la Ecuación (1).

El documento WO 01/41156 describe un procedimiento de procesamiento de un fichero de datos que especifica una jerarquía de nodos, para determinar un intervalo de parámetros asumidos por los nodos.

El documento US 6.075.901 da a conocer un procedimiento y un sistema para codificar de manera uniforme matrices de valores en un flujo de vídeo.

El MPEG ha presentado una solicitud de propuestas para la compresión de interpoladores y ha publicado un conjunto de condiciones experimentales para la evaluación del rendimiento en esta labor (Documentos de salida ISO/IEC JTC1/SC29/WG11 N4098 y N4364, respectivamente).

Se presenta una historia de la animación tridimensional en Jang, "3D Animation Coding: its history and framework", Proceedings of IEEE International Conference on Multimedia and Expo 2000, tomo 2, páginas 1119-1122.

Se presenta un análisis del estándar MPEG-4 en Signes, "Binary Format for Scene (BIFS): Combining MPEG-4 media to build rich multimedia services", Proceedings of the SPIE, tomo 3653, páginas 1506-1517.

La presente invención busca proporcionar un procedimiento y un aparato para codificar datos de valores clave de un interpolador de orientación calculando un valor diferencial de rotación que pueda reflejar suficientemente la redundancia de una transformación rotacional.

La presente invención también busca proporcionar un procedimiento y un aparato para decodificar datos de valores clave codificados codificando un valor diferencial de rotación.

La presente invención busca, además, proporcionar un procedimiento y un aparato para generar datos diferenciales de rotación, que se adoptan en el procedimiento y el aparato para codificar datos de valores clave según la presente invención.

5 Según un primer aspecto de la presente invención, se proporciona un aparato para codificar datos de valores clave de un interpolador de orientación que representan la transformación rotacional de un objeto en tramas clave, según la reivindicación 15.

10 Preferentemente, el generador de datos diferenciales de rotación incluye un cuantificador, que genera datos diferenciales de rotación cuantificando tres valores componentes del valor diferencial de rotación, un ajustador de datos cuantificados, que ajusta tres valores componentes de datos diferenciales de rotación introducidos en el mismo, un cuantificador inverso que cuantifica de manera inversa los valores componentes ajustados, un restaurador de valores diferenciales de rotación que restaura un valor componente, que no ha sido cuantificado, usando los tres valores componentes cuantificados de manera inversa y genera así un valor diferencial restaurado de rotación, y una unidad de medición de errores que mide un error entre un valor diferencial de rotación introducido en el cuantificador y el valor diferencial restaurado de rotación da salida a datos diferenciales de rotación que tienen valores componentes ajustados para que pueda minimizarse el error.

15 De manera alternativa, el generador de datos diferenciales de rotación puede incluir un primer multiplicador de cuaterniones, que genera el valor diferencial de rotación usando el valor de transformación rotacional de la trama clave actual y el valor restaurado de transformación rotacional de la trama clave anterior; un cuantificador, que genera datos diferenciales de rotación cuantificando el valor diferencial de rotación; un cuantificador inverso, que genera un valor diferencial restaurado de rotación cuantificando de manera inversa los datos diferenciales de rotación; y un segundo multiplicador de cuaterniones, que genera un valor diferencial restaurado de rotación de la trama clave actual mediante la multiplicación por cuaterniones del valor diferencial restaurado de rotación por un valor de transformación rotacional de la trama clave anterior.

20 Según un segundo aspecto de la presente invención, se proporciona un procedimiento para codificar datos de valores clave de un interpolador de orientación que representan la rotación de un objeto en cada trama clave, según la reivindicación 51. Preferentemente, el procedimiento incluye generar el valor diferencial de rotación usando un valor de transformación rotacional de la trama clave actual y un valor restaurado de transformación rotacional de la trama clave anterior, generando datos diferenciales de rotación cuantificando el valor diferencial de rotación, generar un valor diferencial restaurado de rotación cuantificando de manera inversa los datos diferenciales de rotación, y generar un valor restaurado de transformación rotacional de la trama clave actual mediante la multiplicación por cuaterniones del valor diferencial restaurado de rotación por un valor de transformación rotacional de la trama clave anterior.

25 Preferentemente, el procedimiento incluye generar datos diferenciales de rotación cuantificando tres valores componentes del valor diferencial de rotación, ajustando tres valores componentes de los datos diferenciales de rotación, cuantificando de manera inversa los valores componentes ajustados, generando un valor diferencial restaurado de rotación restaurando un valor componente usando los tres valores componentes cuantificados de manera inversa, y midiendo un error entre el valor diferencial de rotación y el valor diferencial restaurado de rotación y determinando datos diferenciales de rotación que tienen valores componentes ajustados para que pueda minimizarse el error como datos diferenciales de rotación que han de ser codificados por entropía.

30 Según un tercer aspecto de la presente invención, se proporciona un aparato para decodificar una corriente de datos, en el que se codifican datos de valores clave de un interpolador de orientación que representan la transformación rotacional de un objeto en tramas clave, según la reivindicación 35.

35 Según un cuarto aspecto de la presente invención, se proporciona un procedimiento para decodificar una corriente de datos en un procedimiento de animación gráfica basada en tramas clave, en el cual se codifican datos de valores clave de un interpolador de orientación que representan la rotación de un objeto en cada trama clave, según la reivindicación 71.

Según un quinto aspecto de la presente invención, se proporciona una corriente de datos, en la que se codifican datos de valores clave de un interpolador de orientación que representan la rotación de un objeto en cada trama clave, según la reivindicación 75.

40 Según un sexto aspecto de la presente invención, se proporciona un aparato para generar datos diferenciales de rotación que genera datos diferenciales de valores de transformación rotacionales que representan la rotación de un objeto en cada trama, según la reivindicación 1.

45 Según un séptimo aspecto de la presente invención, se proporciona un procedimiento para generar datos diferenciales de rotación que genera datos diferenciales de valores de transformación rotacionales que representan la rotación de un objeto en cada trama, según la reivindicación 39.

Los aspectos y las ventajas anteriores y otros de la presente invención se harán más evidentes describiendo con detalle realizaciones preferentes de la misma con referencia a los dibujos adjuntos, en los cuales:

- la FIG. 1 es un diagrama que ilustra trayectorias bidimensionales de datos de animación, representadas por un nodo interpolador de orientación, según el paso del tiempo en la superficie de una esfera tridimensional;
- 5 las FIGURAS 2A y 2B son diagramas de bloques de un codificador MPEG-4 PMFC que usa DPCM lineal y codificación por entropía y de un decodificador MPEG-4 PMFC que DPCM lineal inversa y decodificación por entropía, respectivamente;
- la FIG. 3A es un diagrama de bloques de un aparato para codificar datos de valores clave según una primera realización de la presente invención, y la FIG. 3B es un diagrama de bloques de un aparato para decodificar datos de valores clave según una primera realización de la presente invención;
- 10 la FIG. 4A es un diagrama que ilustra un ejemplo típico de una función de distribución de las probabilidades (PDF) en cada componente de un valor diferencial de rotación;
- la FIG. 4B es una curva arco tangente para la cuantificación no lineal;
- la FIG. 5A es un ejemplo de una salida de datos diferenciales de rotación de un cuantificador según una realización preferente de la presente invención, la FIG. 5B es un diagrama que ilustra los resultados de llevar a cabo una operación de DPCM lineal con los datos diferenciales mostrados en la FIG. 5A, y la FIG. 5C es un diagrama que ilustra los resultados de llevar a cabo una operación de DPCM circular con los datos diferenciales sometidos a DPCM mostrados en la FIG. 5B;
- 15 la FIG. 6A es un diagrama que ilustra un ejemplo de una función SignedAAC() usada para la codificación por entropía, y la FIG. 6B es un diagrama que ilustra un ejemplo de una función UnaryAAC() usada para la codificación por entropía;
- la FIG. 7 es un diagrama que ilustra un error de la dirección de rotación que ocurre durante la codificación de valores de transformación rotacionales del cuaternión usando un valor diferencial de rotación;
- la FIG. 8A es un diagrama de bloques de un operador de DPCM circular según una segunda realización de la presente invención, y la FIG. 8B es un diagrama de bloques de un calculador del error de la dirección de rotación mostrado en la FIG. 8A;
- 25 la FIG. 9A es un diagrama de flujo de una operación de DPCM circular según una segunda realización de la presente invención, y la FIG. 9B es un diagrama de flujo que ilustra las operaciones de un calculador del error de la dirección de rotación, un detector del error de la dirección de rotación y un corrector de la dirección de rotación mostrados en la FIG. 9A;
- la FIG. 10A es un diagrama de bloques de un cuantificador incluido en un aparato para codificar datos de valores clave según una tercera realización de la presente invención, y la FIG. 10B es un diagrama de flujo de la operación del cuantificador según la tercera realización de la presente invención;
- 30 la FIG. 11A es un diagrama de bloques de un aparato para decodificar datos de valores clave de un interpolador de orientación según una realización preferente de la presente invención, y el FIG. 11B es un diagrama de flujo de un procedimiento para decodificar datos de valores clave según una realización preferente de la presente invención;
- la FIG. 12 es un diagrama que ilustra la estructura de una corriente de datos introducida en un decodificador de entropía;
- 40 las FIGURAS 13A a 13E son diagramas que ilustran ejemplos de códigos de programa en lenguaje SDL mediante los cuales se realiza un aparato para decodificar datos de valores clave según la presente invención;
- la FIG. 14 es un diagrama que ilustra un ejemplo de un procedimiento para la medición de un error entre una trayectoria original de una animación y una trayectoria restaurada de animación; y
- 45 la FIG. 15 es un diagrama para comparar el rendimiento de un procedimiento para codificar datos de valores clave según la presente invención con el rendimiento de un procedimiento convencional para codificar datos de valores clave.

En lo que sigue, la presente invención será descrita ahora más plenamente con referencia a los dibujos adjuntos, en los cuales se muestran realizaciones preferentes de la invención.

La FIG. 3A es un diagrama de bloques de un aparato para codificar datos de valores clave de un nodo interpolador de orientación según una primera realización de la presente invención. Con referencia a la FIG. 3A, el aparato para codificar datos de valores clave de un nodo interpolador de orientación según la primera realización de la presente invención incluye un operador 300 de DPCM rotacional, que calcula valores diferenciales de rotación entre los valores de transformación rotacionales de un objeto en tramas clave sucesivas, que se obtienen aplicando datos de valores clave del cuaternión de las tramas clave a sus objetos respectivos, cuantifica los valores diferenciales de rotación, y da salida a datos diferenciales de rotación, un operador 400 de DPCM circular, que lleva a cabo de manera selecta una operación de DPCM lineal y una operación de DPCM circular con datos diferenciales de rotación cuantificados, un codificador 450 de entropía, que codifica por entropía datos de rotación sometidos a DPCM rotacional o datos diferenciales de rotación sometidos a DPCM circular, y un codificador 500 de cabeceras de valor clave, que codifica información necesaria para decodificar los datos de valores clave codificados de un nodo interpolador de orientación.

El operador 300 de DPCM rotacional incluye un primer multiplicador 310 de cuaterniones, que calcula un valor diferencial de rotación entre un valor de transformación rotacional de un objeto en una trama clave anterior y un valor de transformación rotacional del objeto en una trama clave actual mediante la multiplicación por cuaterniones del valor de transformación rotacional del objeto en la trama clave actual por el valor de transformación rotacional del objeto en la trama clave anterior, un cuantificador 340, que genera datos diferenciales de rotación cuantificados cuantificando de manera no lineal el valor diferencial de rotación introducido desde el primer multiplicador 310 de cuaterniones, un cuantificador inverso 350, que genera datos diferenciales de rotación restaurados del objeto en la trama clave actual cuantificando de manera inversa datos diferenciales de rotación cuantificados, un segundo multiplicador 370 de cuaterniones, que restaura el valor de transformación rotacional del objeto en la trama clave actual multiplicando por cuaterniones el valor diferencial de rotación del objeto en la trama clave actual por los valores de transformación rotacionales del objeto en tramas clave anteriores calculados acumulado los datos diferenciales de rotación, y un retardador 390, que da salida del valor restaurado de transformación rotacional del objeto en la trama clave actual al primer multiplicador 310 de cuaterniones cuando se introduce un valor de transformación rotacional del objeto en una trama siguiente.

En lo que sigue, se describirá con mayor detalle una operación de DPCM rotacional llevada a cabo en un operador de DPCM rotacional según la presente invención.

En un procedimiento de DPCM lineal que ha sido adoptado en un MPEG-4 PMFC convencional, se calcula un valor diferencial entre un valor de transformación rotacional por cuaterniones  $Q_1$  ( $Q_1 = (q_{1,0}, q_{1,1}, q_{1,2}, q_{1,3})$ ) que representan la transformación rotacional de un objeto en una trama clave actual (o que representan el grado en el que es transformado rotacionalmente un objeto en una trama clave actual) y un valor de transformación rotacional por cuaterniones  $Q_2$  ( $Q_2 = (q_{2,0}, q_{2,1}, q_{2,2}, q_{2,3})$ ) que representa la transformación rotacional del objeto en una trama clave siguiente según la Ecuación (4).

$$Q_{DPCMlineal} = (q_{1,0} - q_{2,0}, q_{1,1} - q_{2,1}, q_{1,2} - q_{2,2}, q_{1,3} - q_{2,3}) \quad \dots(4)$$

Sin embargo, el procedimiento de DPCM lineal solo calcula un valor diferencial entre componentes del cuaternión, lo que no refleja ningún valor diferencial de rotación significativo, es decir, un valor diferencial real de rotación. En consecuencia, el procedimiento de DPCM lineal no puede reducir de manera efectiva la redundancia entre los datos de valores clave sucesivos que han de ser codificados. Además, en el procedimiento de DPCM lineal, se codifican cuatro componentes de un cuaternión, salvo el componente que tiene el mayor valor. Por lo tanto, es necesario codificar adicionalmente información con dos bits de longitud en el componente que no ha sido codificado, y transmitir la información con dos bits de longitud a un decodificador desde un codificador.

Para resolver el problema anterior con el procedimiento de DPCM lineal convencional reduciendo la redundancia entre datos de valores clave sucesivos, el aparato para codificar datos de valores clave según la primera realización de la presente invención proporciona una operación de DPCM rotacional en la que se consideran valores diferenciales de rotación reales.

Cuando  $\hat{x}$  representa un vector de referencia que representa una posición de referencia de un objeto en una trama clave actual,  $(\hat{n}_{i-1}, \theta_{i-1})$  representa datos de valores clave cuando los datos clave son iguales a  $k_{i-1}$ , e  $\hat{y}_{i-1}$  es un vector de desplazamiento de  $\hat{x}$  obtenido transformando rotacionalmente el objeto en la trama clave actual, la transformación rotacional en un espacio de cuaterniones puede expresarse por medio de la siguiente ecuación.

$$Y_{i-1} = Q_{i-1} \times X_0 \times Q_{i-1}^* \quad \dots(5)$$

En la Ecuación (5),  $X_0$ ,  $Y_{i-1}$ ,  $Q_{i-1}$ , y  $Q_{i-1}^*$  representan cuaterniones unitarios de  $\hat{x}$ ,  $\hat{y}_{i-1}$ ,  $(\hat{n}_{i-1}, \theta_{i-1})$  y  $(\hat{n}_{i-1}, \theta_{i-1})^*$ , respectivamente. Además,  $Q_{i-1}^*$  representa un conjugado complejo de cuaternión de  $Q_{i-1}$ , y H representa una multiplicación de cuaterniones.

De esta manera, cuando los datos clave son iguales a  $k_i$ , la información rotacional en un espacio de cuaterniones puede expresarse por medio de la siguiente ecuación.

$$Y_i = Q_i \times X_0 \times Q_i^* \quad \dots(6)$$

Un valor diferencial de rotación entre valores de transformación rotacionales sucesivos de datos de valores clave sucesivos puede calcularse siguiendo la Ecuación (7).

$$Y_i = Q_i \times X_0 \times Q_i^* = Q_i \times Q_{i-1}^* \times Y_{i-1} \times Q_{i-1} \times Q_i^* = Q_i \times Y_{i-1} \times Q_i^* \quad \dots(7)$$

5 En consecuencia, un valor diferencial de rotación puede definirse por medio de la siguiente ecuación, que se deriva de las Ecuaciones (5) y (6).

$$Q_i = Q_i \times Q_{i-1}^* \quad \dots(8)$$

Para evitar que un error de cuantificación afecte a un siguiente valor diferencial de rotación, el aparato para codificar datos de valores clave según la primera realización de la presente invención redefine el valor diferencial de rotación definido por la Ecuación (8) usando un valor de transformación rotacional en una trama clave siguiente y un valor

10 restaurado de transformación rotacional  $\hat{Q}_{i-1}^*$  en una trama clave anterior, que se muestra en la Ecuación (9).

$$Q_i = Q_i \times \hat{Q}_{i-1}^* \quad \dots(9)$$

En lo que sigue, se describirá un procedimiento para codificar datos de valores clave según una realización preferente de la presente invención con referencia a la FIG. 3B. La FIG. 3B es un diagrama de flujo de un procedimiento para codificar datos de valores clave según una realización preferente de la presente invención.

15 El aparato para codificar datos de valores clave según la primera realización de la presente invención recibir el orden de la DPCM, un modo de codificación por entropía y datos de valores clave en la etapa S300.

Acto seguido, el primer multiplicador 310 de cuaterniones recibe datos de valores clave  $Q_i$  representados por un cuaternión y comprueba si los datos de valores clave  $Q_i$  introducidos son o no primeros datos de valores clave  $Q_0$  en la etapa S305. Si los datos de valores clave  $Q_i$  introducidos son los primeros datos de valores clave  $Q_0$ , ello significa que no se usa ningún valor acumulado de transformación por cuaterniones para la multiplicación de cuaterniones.

20 En consecuencia, el primer multiplicador 310 de cuaterniones da salida a los datos de valores clave  $Q_i$  introducidos al cuantificador 340. Si los datos de valores clave  $Q_i$  introducidos no son los primeros datos de valores clave  $Q_0$ , el primer multiplicador 310 de cuaterniones calcula un valor diferencial de cuaternión  $\dot{Q}_i$ , que es un valor diferencial entre los datos de valores clave de la trama clave actual y los datos de valores clave restaurados de la trama clave anterior, en la etapa S310 multiplicando por cuaterniones ( $Q_i$  H  $\hat{Q}_{i-1}^*$ ) el valor de transformación restaurado del

25 cuaternión  $\hat{Q}_{i-1}^*$  (conjugado complejo de  $\hat{Q}_{i-1}$ ) en la trama clave anterior por los datos de valores clave  $Q_i$  introducidos representados por un valor de transformación de un cuaternión.

El cuantificador 340 recibe del primer multiplicador 310 de cuaterniones los primeros datos de valores clave  $Q_0$  o el valor diferencial de rotación del cuaternión  $Q_i$  y cuantifica la entrada usando un número predeterminado de bits de cuantificación en la etapa S330.

30 Dado que todos los valores diferenciales de cuaterniones rotacionales están representados cada uno por un cuaternión unitario, una norma de un cuaternión que representa un valor diferencial de rotación es siempre 1. Por lo tanto, un componente entre cuatro componentes de un cuaternión que no haya sido codificado puede ser decodificado usando los otros tres componentes. El aparato para codificar datos de valores clave de un nodo interpolador de orientación según la primera realización de la presente invención codifica solo tres componentes entre cuatro componentes de un cuaternión para reducir la cantidad de datos que han de ser codificados. En consecuencia el cuantificador 340 cuantifica solamente los tres componentes que serán codificados.

El cuantificador 340 según la presente invención lleva a cabo una cuantificación no lineal en vez de una cuantificación lineal general, y la razón se describirá más abajo con referencia a la FIG. 4A.

35 La FIG. 4A es un diagrama que ilustra un ejemplo típico de una función de distribución de las probabilidades (PDF) en cada componente de un valor diferencial de rotación. Tal como se muestra en la FIG. 4A, generalmente los valores diferenciales de los valores componentes de rotación en torno a 0, lo que se denomina compactación de la energía y significa que es posible reducir de manera efectiva la redundancia entre informaciones de la rotación. En consecuencia, para cuantificar valores diferenciales de rotación, reflejando más suficientemente los valores componentes inferiores de cada uno de los valores diferenciales de rotación, es preciso que la cuantificación se lleve a cabo de manera más elaborada con los valores componentes inferiores, y por eso el cuantificador 340 según la

presente invención lleva a cabo una cuantificación no lineal. Aquí, el cuantificador 340 usa una curva de arco tangente para asignar un factor de escala no lineal a cada uno de los valores diferenciales de rotación.

La FIG. 4B es una curva de arco tangente usada para la cuantificación según la presente invención. Tal como se muestra en la FIG. 4B, la curva de arco tangente proporciona una resolución apropiada a los valores superiores introducidos y proporciona una resolución mucho mayor a los valores inferiores introducidos. El cuantificador 340 cuantifica un valor diferencial de rotación en la etapa S330 usando una función de escala no lineal, lo que se muestra en la ecuación siguiente.

$$\begin{aligned} \tilde{q}_i &= \text{piso}\left(q'_i \cdot \left(2^{nQBits-1} - 1\right) + 0,5\right) \quad (q'_i \geq 0) \\ &= -\text{piso}\left(-q'_i \cdot \left(2^{nQBits-1} - 1\right) + 0,5\right) \quad (q'_i < 0) \quad \dots(10) \\ &\left(q'_i = \frac{4}{\pi} \cdot \tan^{-1}(q_i), i = 1, 2, 3, -1 \leq q_i \leq 1\right) \end{aligned}$$

Aquí,  $q$  representa cada componente del valor diferencial de rotación introducido en el cuantificador 340,  $q'_i$  representa un valor escalado de  $q_i$ ,  $nQBits$  representa un número predeterminado de bits de cuantificación usados para la cuantificación y  $\text{piso}(x)$  representa una función para convertir un valor  $x$  introducido en un entero máximo no mayor que  $x$ .

El cuantificador 340 lleva a cabo la cuantificación con el valor diferencial de rotación introducido y después comprueba si los datos diferenciales de rotación cuantificados corresponden o no a los últimos datos de valores clave, el cuantificador 340 da salida a los datos diferenciales de rotación cuantificados al operador de DPCM circular 400. Si los datos diferenciales de rotación cuantificados no corresponden a los últimos datos de valores clave, el cuantificador 340 da salida a los datos diferenciales de rotación cuantificados al cuantificador inverso 350.

En la etapa S340, el cuantificador inverso 350 cuantifica de manera inversa los datos diferenciales de rotación cuantificados introducidos desde el cuantificador 340 y da salida a un valor diferencial restaurado de rotación en la etapa S350.

Tal como se ha descrito más arriba, el aparato para codificar datos de valores clave según la primera realización de la presente invención codifica solo tres componentes de un cuaternión, lo que representa un valor diferencial de rotación cuantificando los tres componentes que no son el primer componente. Se supone que un cuantificador inverso 1130 de un aparato para decodificar datos de valores clave, que se muestra en la FIG. 11A, y el cuantificador inverso 350, que ha sido descrito más arriba, restauran el otro, que no ha sido codificado, entre los cuatro componentes del valor diferencial de rotación usando los tres componentes codificados. Dado que todos los valores diferenciales de rotación están representados cada uno por un cuaternión unitario, la norma de que un cuaternión represente un valor diferencial de rotación es siempre 1. Por lo tanto, es posible restaurar el primer componente entre los cuatro componentes del valor diferencial de rotación usando la ecuación siguiente.

$$\hat{q}_0 = \sqrt{1 - (\hat{q}_1^2 + \hat{q}_2^2 + \hat{q}_3^2)} \quad (\hat{q}_0 \geq 0) \quad \dots(11)$$

En la Ecuación (11),  $\hat{q}_1$ ,  $\hat{q}_2$  y  $\hat{q}_3$  representan tres componentes almacenados de un valor diferencial de rotación, y  $\hat{q}_0$  representa un primer componente restaurado que usa los tres componentes restaurados  $\hat{q}_1$ ,  $\hat{q}_2$  y  $\hat{q}_3$ .

Para restaurar el primer componente  $\hat{q}_0$  siguiendo la Ecuación (11), el primer componente  $\hat{q}_0$  debe tener un valor positivo, y esta condición puede ser satisfecha aprovechando las características de un cuaternión, que aparecen cuando se aplica un cuaternión a la transformación rotacional de un objeto en un espacio tridimensional y se muestran en la Ecuación (12).

$$Y = Q \times X \times Q^* = (-Q) \times X \times (-Q)^* \quad \dots(12)$$

La Ecuación (12) muestra que los valores de transformación rotacionales  $Q$  y  $-Q$  son iguales en términos de sus funciones cuando se aplica la transformación rotacional a un objeto en un espacio tridimensional. En consecuencia, si el primer componente de un valor diferencial de rotación en el aparato para codificar datos de valores clave según la primera realización de la presente invención tiene un valor negativo, el valor diferencial de rotación puede convertirse fácilmente en un número positivo multiplicando cada uno de los componentes del valor diferencial de rotación por  $-1$ . En este caso, sin embargo, la suma del cuadrado de cada uno de los componentes de un valor diferencial restaurado de rotación, salvo el primer componente, puede superar 1 debido a un error de cuantificación. En este caso,  $\hat{q}_0$  no puede ser determinado por la Ecuación (11) y se considera un valor que es cercano a 0 y es

menor que un valor mínimo que puede ser cuantificado por el cuantificador 340. El hecho de que  $\hat{q}_0$  tenga tal valor significa que el objeto ha sido transformado rotacionalmente en hasta aproximadamente 180 grados. En consecuencia, un aparato para decodificar datos de valores clave precisa un procedimiento para determinar  $\hat{q}_0$  a la vez que para minimizar el efecto del primer valor componente restaurado  $\hat{q}_0$  sobre los tres valores componentes

5 restaurados  $\hat{q}_1$ ,  $\hat{q}_2$  y  $\hat{q}_3$ , y el procedimiento también será aplicado al cuantificador inverso 350 del aparato para codificar datos de valores clave según la primera realización de la presente invención. Por ejemplo, el valor mínimo que puede ser cuantificado por el cuantificador 340 y el valor mínimo multiplicado por un entero predeterminado puede ser determinado como  $\hat{q}_0$ . Puede usarse la siguiente ecuación para determinar  $\hat{q}_0$ .

$$\hat{q}_0 = a \cdot 2^{-m} \quad \dots(13)$$

10 En la Ecuación (13), 'a' representa una constante arbitraria y m representa un número predeterminado de bits de cuantificación.

Aunque se refiere una información de 2 bits de longitud en cada uno de los datos de valores clave para la decodificación en el procedimiento MPEG-4 BIFS PMFC convencional, la información de 2 bits de longitud no es necesaria en la presente invención y, por ello, el número de bits que han de ser codificados puede disminuir en hasta 2N cuando se codifican N datos de valores clave.

15 El valor diferencial restaurado de rotación al que se da salida desde el cuantificador inverso 350 es introducido en el segundo multiplicador 370 de cuaterniones, y el segundo multiplicador 370 de cuaterniones comprueba si una trama clave introducida es una primera trama clave que ha de ser codificada en la etapa S360. Si la trama clave introducida es la primera trama clave que ha de ser codificada, el segundo multiplicador 370 de cuaterniones acumula el valor restaurado de transformación rotacional de la primera trama clave introducida desde el

20 cuantificador inverso 350 ( $\hat{Q}_0 = \hat{Q}_0$ ) and da salida al valor acumulado al retardador 390 en la etapa S380.

Si la trama clave introducida es la primera trama clave que ha de ser codificada, el segundo multiplicador 370 de cuaterniones restaura un valor de transformación rotacional  $\hat{Q}_i$  de una trama clave actual en la etapa S370 multiplicando por cuaterniones un valor diferencial de rotación  $\hat{Q}_i$  de la trama clave actual y un valor restaurado de transformación rotacional  $\hat{Q}_{i-1}$  de una trama clave anterior ( $\hat{Q}_i = \hat{Q}_i \times \hat{Q}_{i-1}$ ).

25 El segundo multiplicador 370 de cuaterniones da salida al valor restaurado de transformación rotacional  $\hat{Q}_i$  de la trama clave actual al retardador 390, y luego el retardador 390 guarda  $\hat{Q}_i$  hasta que se introduce un valor de transformación rotacional de una trama clave siguiente, de modo que  $\hat{Q}_{i-1}$  puede ser igual que  $\hat{Q}_{i-1}$  y luego da salida a  $\hat{Q}_{i-1}$  al primer multiplicador 310 de cuaterniones.

30 Cuando se introduce en el primer multiplicador 310 de cuaterniones un valor de transformación rotacional  $Q_i$  de una trama clave actual, el primer multiplicador 310 de cuaterniones también recibe el valor restaurado de transformación rotacional  $\hat{Q}_{i-1}$  de una trama clave anterior desde el retardador 390 y genera un valor diferencial de rotación  $\hat{Q}_i$  entre el valor de transformación rotacional de la trama clave actual y el valor restaurado de transformación rotacional de la trama clave anterior en la etapa S310 multiplicando por cuaterniones el valor restaurado de transformación del cuaternión  $\hat{Q}_{i-1}^*$  en la trama clave anterior por  $Q_i$ . ( $Q_i \cdot \hat{Q}_{i-1}^*$ ).

35 Se da salida al valor diferencial de rotación generado al cuantificador 340 y después el cuantificador 340 lo cuantifica en la etapa S330, tal como se ha descrito más arriba. El cuantificador 340 da salida a datos diferenciales de rotación generados cuantificando un valor diferencial de rotación al operador de DPCM circular 400 en la etapa S340 si el valor diferencial de rotación introducido corresponde a los últimos datos de valores clave.

40 El operador de DPCM circular 400 comprueba si el orden de DPCM de los datos diferenciales de rotación cuantificados introducidos desde el cuantificador 340 es 0. Si el orden de DPCM es 0, el operador de DPCM circular 400 da salida a los datos diferenciales de rotación cuantificados al codificador 450 de entropía en la etapa S400 sin

llevar a cabo una operación de DPCM lineal y una operación de DPCM circular. Si el orden de DPCM no es 0, el operador de DPCM circular 400 lleva a cabo una operación de DPCM lineal y una operación de DPCM circular con los datos diferenciales de rotación cuantificados en la etapa S410.

5 La FIG. 5A es un diagrama que ilustra un ejemplo de datos diferenciales a los que se da salida desde el cuantificador 340, y la FIG. 5B es un diagrama que ilustra los resultados de llevar a cabo una operación de DPCM lineal con los datos diferenciales a los que se da salida desde el cuantificador 340.

10 Tal como se muestra en la FIG. 5B, como consecuencia de una operación de DPCM lineal, el intervalo de datos diferenciales que ha de ser codificado puede aumentarse el doble de lo que se acostumbraba. El propósito del desempeño de la operación de DPCM circular es mantener el intervalo de datos diferenciales dentro del intervalo de datos diferenciales cuantificados.

15 La operación de DPCM circular se lleva a cabo con la suposición de que un valor máximo y un valor mínimo en un intervalo de cuantificación están conectados de forma circular entre sí. En consecuencia, si los datos diferenciales, que son el resultado de llevar a cabo una DPCM lineal en dos datos cuantificados consecutivos, son mayores que la mitad del valor máximo en el intervalo de cuantificación, pueden ser representados por valores menores restando el valor máximo de los datos diferenciales.

Si los datos diferenciales son menores que la mitad del valor mínimo en el intervalo de cuantificación, pueden ser representados por valores menores aún más añadiendo el valor máximo en el intervalo de cuantificación a los datos diferenciales.

20 Cuando  $\tilde{Q}_i$  y  $\tilde{Q}_{i-1}$  representan datos diferenciales de rotación cuantificados en dos instantes de tiempo sucesivos  $t_i$  y  $t_{i-1}$ , respectivamente, se lleva a cabo una operación de DPCM lineal con los dos datos diferenciales de rotación cuantificados sucesivos  $\tilde{Q}_i$  y  $\tilde{Q}_{i-1}$  siguiendo la Ecuación (14).

$$X_i = \tilde{Q}_i - \tilde{Q}_{i-1} = (\tilde{q}_{i,1} - \tilde{q}_{i-1,1}, \tilde{q}_{i,2} - \tilde{q}_{i-1,2}, \tilde{q}_{i,3} - \tilde{q}_{i-1,3})^T \quad \dots(14)$$

Además, se lleva a cabo una operación de DPCM circular con los datos diferenciales obtenidos siguiendo la Ecuación (14), siguiendo la Ecuación (15).

$$\begin{aligned} \tilde{Q}_i &= \min(|X_i|, X_i') \\ X_i' &= X_i - (2^{n_{QBits}} - 1) \quad (si \ X_i \geq 0) \\ X_i' &= X_i + (2^{n_{QBits}} - 1) \quad (en \ los \ demás \ casos) \end{aligned} \quad \dots(15)$$

25 En la Ecuación (15),  $n_{QBits}$  representa un número predeterminado de bits de cuantificación. La FIG.5C es un diagrama que ilustra los resultados de llevar a cabo una operación de DPCM circular con los datos diferenciales sometidos a DPCM mostrados en la FIG. 5B. tal como se muestra en la FIG. 5C, el intervalo de los datos diferenciales sometidos a DPCM circular es mucho menor que el intervalo de los datos diferenciales sometidos a DPCM lineal.

30 El codificador 450 de entropía recibe los datos diferenciales de rotación o los datos diferenciales de rotación sometidos a DPCM circular del operador de DPCM circular 400 dependiendo del orden de DPCM de los datos diferenciales de rotación y luego codifica los datos diferenciales introducidos eliminando la redundancia de los bits.

Con referencia a la FIG. 3B nuevamente, el codificador 450 de entropía comprueba un modo de codificación por entropía en la etapa S450.

35 Si el modo de codificación por entropía es un modo binario de codificación por entropía, el codificador 450 de entropía codifica los datos diferenciales introducidos usando una función SignedAAC() en la etapa S460. Si el modo de codificación por entropía es un modo unario de codificación por entropía, el codificador 450 de entropía codifica los datos diferenciales introducidos usando una función UnaryAAC() en la etapa S470.

La función SignedAAC() se usa para codificar datos diferenciales usando un codificador aritmético binario adaptable, que codifica el signo y la magnitud de los datos diferenciales en cada plano de bits.

40 La FIG. 6A es un diagrama que ilustra un ejemplo de la función SignedAAC(). Con referencia a la FIG. 6A, cuando un valor introducido es 74 y el número de bits requeridos para codificar el valor introducido es 8, el signo del valor introducido es 0 y un número binario que representa el número introducido es 1001010. Los signos en dos los planos de bits se codifican de la manera siguiente:

Primera etapa: se codifica un número binario en cada plano de bits en un orden que va de su bit más significativo (MSB) a su bit menos significativo (LSB);

Segunda etapa: se comprueba si el bit que se está codificando actualmente es 0;

5 Tercera etapa: si el bit que se está codificando actualmente no es 0, se codifica a continuación el signo del número binario; y

Cuarta etapa: se codifican los bits restantes del número binario.

La función UnaryAAC() convierte un valor que ha de ser codificado en un número predeterminado de bits, lo que consiste en una serie de ceros, indicando un bit 1 de bandera que termina la serie de ceros, y representando un bit el signo del valor. Aquí, el número de ceros corresponde a la magnitud del valor.

10 La función UnaryAAC() será descrita más plenamente en lo que sigue con referencia a la FIG. 6B. Por ejemplo, 256 se codifica en una serie de bits que consiste en doscientos cincuenta y seis ceros, 1 que es un bit de bandera que indica indicando que termina la serie de ceros, y 0 que representa el signo de 256, es decir, un signo más por la función UnaryAAC(). Debido a UnaryAAC(), aumenta la redundancia entre los bits que representan el símbolo que ha de ser codificado, lo que mejora la eficiencia de la codificación del símbolo.

15 En lo que sigue, se describirá un aparato para codificar datos de valores clave según una segunda realización de la presente invención. El aparato para codificar datos de valores clave según la segunda realización de la presente invención incluye un medio para corregir un error de la dirección de rotación que ocurre durante la cuantificación, así como todos los elementos del aparato para codificar datos de valores clave según la primera realización de la presente invención.

20 La FIG. 7 es un diagrama que ilustra un error de la dirección de rotación generado durante la codificación de valores de transformación rotacionales del cuaternión usando valores diferenciales de rotación. El error de la dirección de rotación ocurre porque el procedimiento de codificación de cuaterniones llevado a cabo en el aparato para codificar datos de valores clave según la presente invención es un procedimiento de codificación con pérdida.

25 En la FIG. 7, cuando  $Q_i$ ,  $Q_{i-1}$ ,  $\hat{Q}_i$  y  $\hat{Q}_{i-1}$  representan un valor de transformación rotacional que se está aplicando en la actualidad da un objeto en una trama clave iésima, un valor de transformación rotacional aplicado al objeto en una trama clave anterior, un valor restaurado de  $Q_i$  y un valor restaurado de  $Q_{i-1}$ , respectivamente, las ubicaciones del objeto rotado por las transformaciones rotaciones respectivamente aplicadas por  $Q_i$ ,  $Q_{i-1}$ ,  $\hat{Q}_i$  y  $\hat{Q}_{i-1}$  pueden estar representadas por cuatro áreas diferentes, como se muestra en la FIG. 7.

30 En otras palabras, si el objeto rota de  $Q_{i-1}$  a  $Q_i$  a lo largo del arco más corto, de modo que el objeto esté ubicado en el área 1 y en el área 3 después de rotar en  $Q_i$  y  $Q_{i-1}$ , se considera que el objeto ha rotado de  $Q_{i-1}$  a  $Q_i$  en el sentido contrario a las agujas del reloj. En cambio, si el objeto rota de  $Q_{i-1}$  a  $Q_i$  a lo largo del arco más corto, de modo que el objeto esté ubicado en el área 2 y en el área 4 después de rotar en  $Q_{i-1}$  y  $Q_i$ , se considera que el objeto ha rotado de  $Q_{i-1}$  a  $Q_i$  en el sentido de las agujas del reloj.

35 Por otra parte, si el objeto rota siguiendo la información de rotación que ha sido codificada y después decodificada, un aparato para decodificar datos de valores clave rota el objeto usando  $\hat{Q}_i$  y  $\hat{Q}_{i-1}$ , correspondientes a  $Q_i$  y  $Q_{i-1}$ , respectivamente. En consecuencia, si la ubicación relativa del objeto rotado en  $\hat{Q}_i$  con respecto a la ubicación del objeto rotado en  $\hat{Q}_{i-1}$  es el área 2 o 3  $\hat{Q}_{i-1}$ , el objeto rota en el sentido contrario a las agujas del reloj. Si la ubicación relativa del objeto rotado en  $\hat{Q}_i$  con respecto a la ubicación del objeto rotado en  $\hat{Q}_{i-1}$  es el área 1 o 4, el objeto rota en el sentido de las agujas del reloj. En las áreas 1 y 2, la dirección de rotación del objeto cuando se usa un valor de transformación rotacional original puede ser opuesta a la dirección del rotación del objeto cuando se usa un valor de transformación rotacional decodificado, porque  $Q_i$  y  $\hat{Q}_i$  son diferentes debido a la codificación con pérdida, que se lleva a cabo para codificar valores de transformación rotacionales del cuaternión. Para resolver este problema, se requiere minimizar el grado hasta el que rota el objeto en una dirección equivocada, que es opuesta a la dirección deseada, o corregir la dirección de rotación del objeto cuando el objeto rota en una dirección equivocada, para que el objeto rote en una dirección deseada. En la presente invención, se adopta el procedimiento para corregir la dirección de rotación del objeto para hacer que el objeto rote en una dirección deseada.

40

45

Con referencia nuevamente a la FIG. 7, se describirá en lo que sigue el concepto de corrección de un error de la dirección de rotación según la presente invención. Si se detecta un error de la dirección de rotación, como el fenómeno que ocurre en las áreas 1 y 2, se controlan los valores diferenciales de rotación del cuaternión que han de

codificarse para rotar en una dirección acertada, en cuyo caso la incoherencia en las direcciones de rotación sigue ocurriendo en el área 2. Sin embargo, en el área 2, a diferencia del área 1, las diferencias entre los valores originales del cuaternión y los valores restaurados del cuaternión son relativamente pequeñas. En consecuencia, la corrección de la dirección de rotación según la segunda realización de la presente invención se lleva a cabo únicamente en el área 1.

En lo que sigue, se describirá el aparato para codificar datos de valores clave, que lleva a cabo la corrección de la dirección de rotación, según la segunda realización de la presente invención con referencia a las FIGURAS 8A a 9B. El aparato para codificar datos de valores clave según la segunda realización de la presente invención tiene casi la misma estructura que el aparato para codificar datos de valores clave según la primera realización de la presente invención. La única diferencia entre ellos es la estructura de un operador de DPCM rotacional, y así en lo que sigue solo se describirá la estructura de un operador de DPCM rotacional en el aparato para codificar datos de valores clave según la segunda realización de la presente invención.

La FIG. 8A es un diagrama de bloques de un operador 800 de DPCM rotacional según la segunda realización de la presente invención, y la FIG. 8B es un diagrama de bloques de un calculador del error de la dirección de rotación 820 mostrado en la FIG. 8A.

Con referencia a la FIG. 8A, el operador 800 de DPCM rotacional incluye un calculador 820 del error de la dirección de rotación, que recibe un valor de transformación rotacional de un objeto en una trama clave actual y un valor restaurado de transformación rotacional del objeto en una trama clave anterior y calcula errores de la dirección de rotación, un detector 830 del error de la dirección de rotación, que detecta, en base a errores de la dirección de rotación introducidos desde el calculador 820 del error de la dirección de rotación si ha ocurrido o no durante la decodificación un error suficiente para cambiar la dirección de rotación del objeto, un corrector 815 de la dirección de rotación, que corrige y da salida a un valor diferencial de rotación introducido desde el primer multiplicador 810 de cuaterniones para que el objeto, que está ubicado ya girado siguiendo a un valor de transformación rotacional decodificado de la trama clave anterior, pueda girar adicionalmente en hasta 180 grados en una dirección de rotación original, y un selector 835 de la dirección de rotación, que selecciona el valor diferencial de rotación introducido desde el corrector 815 de la dirección de rotación o el valor diferencial de rotación introducido desde el primer multiplicador 810 de cuaterniones, dependiendo de un valor introducido desde el detector 830 del error de la dirección de rotación, y da salida al valor seleccionado al cuantificador 840.

Con referencia a la FIG. 8B, EL calculador 820 del error de la dirección de rotación mostrado en la FIG. 8A incluye un retardador 822, que almacena un valor de transformación rotacional introducido hasta que se introduce un valor de transformación rotacional de una trama clave siguiente, un tercer multiplicador 824 de cuaterniones, que recibe el valor de transformación rotacional introducido y un valor de transformación rotacional de una trama clave anterior al que da salida el retardador 822, y calcula un valor diferencial de rotación entre el valor de transformación rotacional introducido y el valor de transformación rotacional de la trama clave anterior mediante multiplicación de cuaterniones, un cuarto multiplicador 826 de cuaterniones, que calcula un valor diferencial de rotación entre el valor de transformación rotacional de la trama clave anterior al que da salida el retardador 822 y el valor restaurado de transformación rotacional de la trama clave anterior, y un quinto multiplicador 828 de cuaterniones, que calcula un valor diferencial de rotación entre el valor de transformación rotacional introducido y el valor restaurado de transformación rotacional de la trama clave anterior.

En lo que sigue, se describirá una operación de DPCM rotacional según una segunda realización de la presente invención con mayor detalle con referencia a la FIG. 9A.

Se introducen datos de valores clave  $Q_i$  de un nodo interpolador de orientación, que están a punto de ser codificados, en el primer multiplicador 810 de cuaterniones y el calculador 820 del error de la dirección de rotación del operador 800 de DPCM rotacional en la etapa S900.

El primer multiplicador 810 de cuaterniones, como el primer multiplicador de cuaterniones del aparato para codificar datos de valores clave según la primera realización de la presente invención, genera un valor diferencial de rotación  $\dot{Q}_i$  multiplicando por cuaterniones  $(Q_i \times \hat{Q}_{i-1}^*)$  un valor de transformación rotacional introducido de una trama clave actual por un valor restaurado de transformación rotacional de una trama clave anterior introducido desde el segundo multiplicador 870 de cuaterniones y da salida al valor diferencial de rotación generado  $\dot{Q}_i$  al corrector 815 de la dirección de rotación y al selector 835 de la dirección de rotación en la etapa S910.

El corrector 815 de la dirección de rotación corrige un valor diferencial de rotación introducido en el mismo siguiendo la Ecuación (16) y da salida al valor diferencial de rotación corregido  $Q_s$  al selector 835 de la dirección de rotación. El calculador 820 del error de la dirección de rotación recibe el valor de transformación rotacional  $Q_i$  de la trama clave actual y el valor restaurado de transformación rotacional  $\hat{Q}_{i-1}$  de la trama clave anterior introducido desde el segundo multiplicador 870 de cuaterniones y calcula valores de transformación rotacionales  $Q_A$ ,  $Q_B$  y  $Q_C$ , que serán

descritos más tarde. El detector 830 del error de la dirección de rotación detecta, usando el valor diferencial de rotación introducido desde el calculador 820 del error de la dirección de rotación, si ha ocurrido o no un error de la dirección de rotación y da salida al resultado de la detección al selector 835 de la dirección de rotación en la etapa S920.

$$Q_s = \begin{pmatrix} \frac{|\delta_T|}{\sqrt{(q_{R,1})^2 + (q_{R,2})^2 + (q_{R,3})^2}} q_{R,1} \\ \frac{1}{\sqrt{(q_{R,1})^2 + (q_{R,2})^2 + (q_{R,3})^2}} q_{R,2} \\ \frac{1}{\sqrt{(q_{R,1})^2 + (q_{R,2})^2 + (q_{R,3})^2}} q_{R,3} \end{pmatrix} \quad \dots(16)$$

5 En la Ecuación (16),  $\delta_T$  representa una constante muy cercana a 0, y  $(\dot{q}_{R,0}, \dot{q}_{R,1}, \dot{q}_{R,2}, \dot{q}_{R,3})^T$  representa el valor diferencial de rotación  $\dot{Q}_i$  al que da salida el primer multiplicador 810 de cuaterniones. El calculador 820 del error de la dirección de rotación, el detector 830 del error de la dirección de rotación, y el corrector 815 de la dirección de rotación serán descritos más tarde con referencia a la FIG. 9B.

10 El selector 835 de la dirección de rotación comprueba si ha ocurrido o no un error de la dirección de rotación, de modo que el objeto rote en la dirección opuesta a una dirección deseada cuando se decodifican datos de valores clave codificados usando un valor lógico introducido desde el detector 830 del error de la dirección de rotación. Si no ha ocurrido un error de la dirección de rotación, el selector 835 de la dirección de rotación da salida al valor diferencial de rotación introducido desde el primer multiplicador 810 de cuaterniones al cuantificador 840 en la etapa S930. Si ha ocurrido un error de la dirección de rotación, el selector 835 de la dirección de rotación da salida al valor diferencial de rotación corregido introducido desde el corrector 815 de la dirección de rotación en la etapa S930.

15 El cuantificador 840 cuantifica los datos diferenciales de rotación originales  $\dot{Q}_i$  o los datos diferenciales de rotación corregidos  $Q_s$  en las etapas S940 y 950 en el mismo procedimiento de cuantificación que en la primera realización de la presente invención.

20 El cuantificador 840 comprueba si los datos diferenciales de rotación cuantificados  $\tilde{Q}_i$  pertenecen o no a los últimos datos de valores clave en la etapa S960. Si los datos diferenciales de rotación cuantificados  $\tilde{Q}_i$  corresponden a los últimos datos de valores clave, el cuantificador 840 da salida a los datos diferenciales de rotación cuantificados  $\tilde{Q}_i$  a un operador 400 de DPCM circular en la etapa S970. Si los datos diferenciales de rotación cuantificados  $\tilde{Q}_i$  no corresponden a los últimos datos de valores clave, el cuantificador 840 da salida a los datos diferenciales de rotación cuantificados  $\tilde{Q}_i$  al cuantificador inverso 850.

25 El cuantificador inverso 850 cuantifica de manera inversa los datos diferenciales de rotación cuantificados  $\tilde{Q}_i$  en el mismo procedimiento de cuantificación inversa que en la primera realización de la presente invención y da salida a los datos diferenciales de rotación restaurados  $\hat{Q}_i$  al segundo multiplicador 870 de cuaterniones en la etapa S980.

30 El segundo multiplicador 870 de cuaterniones genera un valor de conversión rotacional restaurado  $\hat{Q}_i$  de una trama clave actual mediante la multiplicación por cuaterniones del valor diferencial restaurado de rotación  $\hat{Q}_i$  por un valor de transformación rotacional  $\hat{Q}_{i-1}$  de una trama clave anterior. El segundo multiplicador 870 de cuaterniones da salida a un valor de transformación rotacional de una trama clave actual al primer multiplicador 810 de cuaterniones y al calculador 820 del error de la dirección de rotación por medio del retardador 890 en la etapa S990.

En lo que sigue, se describirán las operaciones del calculador 820 del error de la dirección de rotación, el detector 830 del error de la dirección de rotación y el corrector 815 de la dirección de rotación con referencia a la FIG. 9B.

El calculador 820 del error de la dirección de rotación calcula valores diferenciales de rotación correspondientes a las áreas 2 a 4, que han sido descritas más arriba con referencia a la FIG. 7.

5 El calculador 820 del error de la dirección de rotación recibe un valor de transformación de rotación, que corresponde a los datos de valores clave de un trama clave que han de ser codificados, recibe un valor restaurado de transformación rotacional de una trama clave anterior antes de la trama clave actual, y calcula un error de la dirección de rotación en la etapa S922.

10 El valor de transformación rotacional correspondiente a los datos de valores clave de un interpolador de orientación de la trama clave actual se introduce directamente en el tercer multiplicador 824 de cuaterniones y el quinto multiplicador 828 de cuaterniones y se introduce en el cuarto multiplicador 826 de cuaterniones por medio del retardador 822 cuando se introduce un valor de transformación rotacional de una trama clave siguiente. Además, el valor restaurado de transformación rotacional de la trama clave anterior al que da salida el segundo multiplicador 870 de cuaterniones es introducido en el cuarto multiplicador 826 de cuaterniones y el quinto multiplicador 828 de cuaterniones.

15 El tercer multiplicador 824 de cuaterniones genera el valor diferencial de rotación  $Q_A$  multiplicando por cuaterniones  $(Q_A = Q_i \times \hat{Q}_{i-1}^*)$  el valor de transformación rotacional de la trama clave actual por el valor de transformación rotacional de la trama clave anterior y da salida a  $Q_A$  al detector 830 del error de la dirección de rotación. El valor diferencial de rotación  $Q_A$  representa una dirección de rotación original de un objeto durante un intervalo de tiempo  $[t_{i-1}, t_i]$ .

20 El cuarto multiplicador 826 de cuaterniones genera el valor diferencial de rotación  $Q_B$  multiplicando por cuaterniones  $(Q_B = Q_{i-1} \times \hat{Q}_{i-1}^*)$  el valor de transformación rotacional de la trama clave anterior por el valor restaurado de transformación rotacional de la trama clave anterior y da salida a  $Q_B$  al detector 830 del error de la dirección de rotación. El valor diferencial de rotación  $Q_B$  calculado por el cuarto multiplicador 826 de cuaterniones representa el error de detección de rotación y la dirección de rotación de un objeto, que son afectados por un error de cuantificación, en un instante de tiempo predeterminado  $t_{i-1}$  y corresponde al área 2 mostrada en la FIG. 7.

25 El quinto multiplicador 828 de cuaterniones genera el valor diferencial de rotación  $Q_C$  multiplicando por cuaterniones  $(Q_C = Q_i \times \hat{Q}_{i-1}^*)$  el valor de transformación rotacional de la trama clave actual por el valor restaurado de transformación rotacional de la trama clave anterior y da salida a  $Q_C$  al detector 830 del error de la dirección de rotación. El valor diferencial de rotación  $Q_C$  calculado por el quinto multiplicador 828 de cuaterniones representa un valor diferencial de rotación que ha de ser codificado en el instante de tiempo predeterminado  $t_i$  y corresponde al área 4 mostrada en la FIG. 7.

30 El detector 830 del error de la dirección de rotación comprueba, usando los valores diferenciales de rotación  $Q_A$ ,  $Q_B$  y  $Q_C$  introducidos desde el calculador 820 del error de la dirección de rotación, si el objeto rota o no en una dirección opuesta a una dirección original en el proceso de decodificación de los datos de valores clave codificados en la etapa S924. Para hacer esto, el detector 830 del error de la dirección de rotación comprueba si los valores diferenciales de rotación  $Q_A$ ,  $Q_B$ , y  $Q_C$  satisfacen o no las Desigualdades (17) a (21). En primer lugar, el detector 830 del error de la dirección de rotación comprueba si los valores diferenciales de rotación  $Q_A$  y  $Q_C$  satisfacen la Desigualdad (17).

$$\begin{pmatrix} q_{A,1} \\ q_{A,2} \\ q_{A,3} \end{pmatrix} \cdot \begin{pmatrix} q_{C,1} \\ q_{C,2} \\ q_{C,3} \end{pmatrix} < 0 \quad \dots(17)$$

40 En la Desigualdad (17), cuando el valor diferencial de rotación  $Q_A$  introducido desde el calculador 820 del error de la

dirección de rotación es expresado por  $Q_A = (q_{A,0}, q_{A,1}, q_{A,2}, q_{A,3})^T$ ,  $\begin{pmatrix} q_{A,1} \\ q_{A,2} \\ q_{A,3} \end{pmatrix}$  representa un vector tridimensional  $(q_{A,1}, q_{A,2}, q_{A,3})^T$  que consiste en los cuatro componentes  $q_{A,0}$ ,  $q_{A,1}$ ,  $q_{A,2}$  y  $q_{A,3}$  de  $Q_A$ , salvo el primer componente  $q_{A,0}$ .

5  $\begin{pmatrix} q_{C,1} \\ q_{C,2} \\ q_{C,3} \end{pmatrix}$  representa un vector tridimensional  $(q_{C,1}, q_{C,2}, q_{C,3})^T$  que consiste en los cuatro componentes  $q_{C,0}, q_{C,1}, q_{C,2}$  y

$q_{C,3}$  de  $Q_C$ , salvo el primer componente  $q_{C,0}$ . La Desigualdad (17) muestra una condición en la que un producto interno de los dos vectores tridimensionales  $(q_{A,1}, q_{A,2}, q_{A,3})^T$  y  $(q_{C,1}, q_{C,2}, q_{C,3})^T$  es menor que 0.

Si el producto interno en la Desigualdad (17) es menor que 0, la dirección de rotación de un objeto cuando se usa  $Q_A$  es opuesta a la dirección de rotación del objeto cuando se usa  $Q_C$ . Cuando el producto interno es menor que 0, el valor de la expresión lógica mostrada en la Desigualdad (17) se fija a 'verdadero'. Cuando el producto interno es mayor que 0, el valor de la expresión lógica se fija a 'falso'.

$$\begin{pmatrix} q_{B,1} \\ q_{B,2} \\ q_{B,3} \end{pmatrix} \cdot \begin{pmatrix} q_{C,1} \\ q_{C,2} \\ q_{C,3} \end{pmatrix} < 0 \quad \dots(18)$$

En la Desigualdad (18), cuando el valor diferencial de rotación  $Q_B$  introducido desde el calculador 820 del error de la

dirección de rotación es expresado por  $Q_B = (q_{B,0}, q_{B,1}, q_{B,2}, q_{B,3})^T$ ,  $\begin{pmatrix} q_{B,1} \\ q_{B,2} \\ q_{B,3} \end{pmatrix}$  representa un vector tridimensional  $(q_{B,1},$

10  $q_{B,2}, q_{B,3})^T$  que consiste en los cuatro componentes  $q_{B,0}, q_{B,1}, q_{B,2}$  y  $q_{B,3}$  de  $Q_B$ , salvo el primer componente  $q_{B,0}$ .

$\begin{pmatrix} q_{C,1} \\ q_{C,2} \\ q_{C,3} \end{pmatrix}$  representa un vector tridimensional  $(q_{C,1}, q_{C,2}, q_{C,3})^T$  que consiste en los cuatro componentes  $q_{C,0}, q_{C,1}, q_{C,2}$  y

$q_{C,3}$  de  $Q_C$ , salvo el primer componente  $q_{C,0}$ . La Desigualdad (18) muestra una condición en la que un producto interno de los dos vectores tridimensionales  $(q_{B,1}, q_{B,2}, q_{B,3})^T$  y  $(q_{C,1}, q_{C,2}, q_{C,3})^T$  es menor que 0.

15 Si el producto interno en la Desigualdad (18) es menor que 0, la dirección de rotación de un objeto cuando se usa  $Q_B$  es opuesta a la dirección de rotación del objeto cuando se usa  $Q_C$ . Cuando el producto interno es menor que 0, el valor de la expresión lógica mostrada en la Desigualdad (18) se fija a 'verdadero'. Cuando el producto interno es mayor que 0, el valor de la expresión lógica se fija a 'falso'.

$$A_{TH} < 2 \cos^{-1} |q_{A,0}| \quad \dots(19)$$

En la Desigualdad (19),  $q_{A,0}$  representa el primer componente del valor diferencial de rotación  $Q_A$ , y  $A_{TH}$  se fija a una constante predeterminada cercana a 0. Cuando, en la Desigualdad (19),  $2 \cos^{-1} |q_{A,0}|$  es mayor que  $A_{TH}$ , el valor de la expresión lógica mostrada en la Desigualdad (19) se define como 'verdadero'. Cuando  $2 \cos^{-1} |q_{A,0}|$  no es mayor que  $A_{TH}$ , el valor de la expresión lógica se define como 'falso'. Cuando la Desigualdad (19) se fija a 'falso', significa que puede ignorarse un ángulo  $\theta (=2 \cos^{-1} |q_{A,0}|)$  de rotación menor que una constante  $A_{TH}$  predeterminada cuando se realiza un aparato para codificar datos de valores clave según la presente invención. Aunque exista un error de la dirección de rotación de hasta el ángulo  $\theta$  de rotación, el error de la dirección de rotación no provoca imágenes muy distorsionadas para los ojos humanos, especialmente cuando se usa un procedimiento de medición de errores propuesto por la presente invención.

$$A_{TH} < 2 \cos^{-1} |q_{B,0}| \quad \dots(20)$$

En la Desigualdad (20),  $q_{B,0}$  representa el primer componente del valor diferencial de rotación  $Q_B$ , y  $A_{TH}$  es lo mismo que en el parámetro correspondiente en la Desigualdad (19). Cuando, en la Desigualdad (20),  $2 \cos^{-1} |q_{B,0}|$  es mayor que  $A_{TH}$ , el valor de la expresión lógica mostrada en la Desigualdad (20) se define como 'verdadero'. Cuando  $2 \cos^{-1} |q_{B,0}|$  no es mayor que  $A_{TH}$ , el valor de la expresión lógica se define como 'falso'.

$$A_{TH} < 2 \cos^{-1} |q_{C,0}| \quad \dots(21)$$

En la Desigualdad (21),  $q_{C,0}$  representa el primer componente del valor diferencial de rotación  $Q_C$ , y  $A_{TH}$  es lo mismo que en el parámetro correspondiente en la Desigualdad (19). Cuando, en la Desigualdad (21),  $2 \cos^{-1} |q_{C,0}|$  es mayor que  $A_{TH}$ , el valor de la expresión lógica mostrada en la Desigualdad (21) se define como 'verdadero'. Cuando  $2 \cos^{-1} |q_{C,0}|$  no es mayor que  $A_{TH}$ , el valor de la expresión lógica se define como 'falso'.

El detector 830 del error de la dirección de rotación lleva a cabo una operación AND con los valores lógicos de las Desigualdades (17) a (21) y da salida al resultado de la operación AND al selector 835 de la dirección de rotación.

5 El primer multiplicador 810 de cuaterniones tiene dos valores  $Q_i$  y  $\hat{Q}_{i-1}$  que son introducidos en un instante de tiempo predeterminado  $t$ . El primer multiplicador 810 de cuaterniones da salida a un valor diferencial de rotación usando los dos valores introducidos  $Q_i$  y  $\hat{Q}_{i-1}$ . Tal como se ha descrito más arriba, en el área 1, el aparato para decodificar datos de valores clave codificados, que recibe datos diferenciales de rotación, rota un objeto en el sentido de las agujas del reloj. Sin embargo, el objeto debe rotar desde un lugar en el que esté situado actualmente después de rotar los grados predeterminados indicados por  $Q_{i-1}$  hasta un lugar en el que se supone que está situado después de rotar los grados predeterminados indicados por  $Q_i$ , y, así, la dirección de rotación original del objeto debe ser una dirección en el sentido contrario a las agujas del reloj.

10 En consecuencia, el corrector 815 de la dirección de rotación corrige la dirección de la rotación del objeto, de modo que el objeto pueda rotar en hasta un ángulo de rotación indicado por  $\hat{Q}_{i-1}$  en la misma dirección que ha rotado en base a  $\hat{Q}_i$ , tal como se muestra en la FIG. 7, es decir, de modo que el objeto pueda rotar en sentido contrario a las agujas del reloj desde el lugar en el que está situado actualmente después de rotar en base a  $\hat{Q}_{i-1}$  hasta el lugar en el que se supone que está situado después de rotar en hasta un ángulo de rotación indicado por  $\hat{Q}_i$ .

15 Para hacer esto, el corrector 815 de la dirección de rotación configura nueva información de rotación para hacer que el objeto rote en sentido contrario a las agujas del reloj en hasta 180 grados desde el lugar en el que está ubicado actualmente después de rotar grados predeterminados según  $\hat{Q}_{i-1}$ . En consecuencia, se corrige la dirección de la rotación del objeto para que sea igual que la original, y puede minimizarse el error de la dirección de rotación. El corrector 815 de la dirección de rotación recibe los datos diferenciales de rotación  $\hat{Q}_i$  de la trama clave actual del primer multiplicador 810 de cuaterniones, genera datos diferenciales de rotación que tienen una dirección corregida de la rotación, y da salida a los datos diferenciales de rotación corregidos al selector 835 de la dirección de rotación en la etapa S926.

25 Con referencia a la FIG. 9A, el selector 835 de la dirección de rotación comprueba si los valores lógicos introducidos desde el detector 830 del error de la dirección de rotación son verdaderos en la etapa S930. Si los valores lógicos introducidos son verdaderos, el selector 835 de la dirección de rotación determina que ha ocurrido el mismo fenómeno que ha ocurrido en el área 1 mostrada en la FIG. 7 y da salida al valor diferencial de rotación corregido  $Q_s$  definido por la Ecuación (16) al cuantificador 840 en la etapa S940.

30 Por otra parte, si los valores lógicos introducidos son falsos, el selector 835 de la dirección de rotación determina que no ha ocurrido el mismo fenómeno que ha ocurrido en el área 1 mostrada en la FIG. 7 y da salida al valor diferencial de rotación  $\hat{Q}_i$  introducido desde el primer multiplicador 810 de cuaterniones al cuantificador 840 en la etapa S950.

En lo que sigue, se describirá un aparato para codificar datos de valores clave según una tercera realización de la presente invención.

35 Dado que los aparatos para codificar datos de valores clave según las realizaciones primera y segunda de la presente invención codifican solo tres componentes entre los cuatro componentes de un valor diferencial de rotación, es posible que no puedan restaurar un primer valor componente del cuaternión usando la Ecuación (11) debido a un error de cuantificación. Por ejemplo, un primer componente de un valor diferencial restaurado de rotación puede ser un número imaginario.

40 Para evitar el problema con la codificación de solo tres componentes de un cuaternión, el aparato para codificar datos de valores clave según la tercera realización de la presente invención incluye un cuantificador, que puede ajustar debidamente tres valores componentes cuantificados para que, cuando se decodifiquen los datos de valores clave codificados, el otro componente pueda ser restaurado a un número real positivo y pueda minimizarse la distorsión de imágenes.

45 El aparato para codificar datos de valores clave según la tercera realización de la presente invención es igual que los aparatos para codificar datos de valores clave según las realizaciones primera o segunda de la presente invención salvo por la estructura de un cuantificador, y, así, en lo que sigue se describirá únicamente la estructura del cuantificador.

La FIG. 10A es un diagrama de bloques de un cuantificador incluido en un aparato para codificar datos de valores clave según la tercera realización de la presente invención. Con referencia a la FIG. 10A, un cuantificador 1000 incluye una unidad 1010 de cuantificación, que cuantifica un valor diferencial de rotación introducido en la misma, un ajustador 1020 de datos de cuantificación, que ajusta un valor diferencial de rotación cuantificado, un cuantificador inverso 1030, que cuantifica de manera inversa datos diferenciales de rotación cuantificados, un restaurador 1040 de valores diferenciales de rotación, que restaura todos los valores componentes de los datos diferenciales de rotación cuantificados restaurando un primer componente de los datos diferenciales de rotación cuantificados usando los componentes cuantificados de manera inversa, y una unidad 1050 de medición de errores, que mide un error entre un valor diferencial restaurado y un valor diferencial de rotación introducido originalmente y renueva los datos diferenciales de rotación cuantificados.

La FIG. 10B es un diagrama de flujo de la operación del cuantificador 1010. Con referencia a la FIG. 10B, cuando se introduce un valor diferencial de rotación  $\dot{Q}$  desde un primer multiplicador de cuaterniones, el cuantificador 1010 cuantifica el valor diferencial introducido de rotación  $\dot{Q}$  usando la Ecuación (10) y da salida a los datos diferenciales de rotación cuantificados  $\tilde{Q}$  al ajustador 1020 de datos de cuantificación en la etapa S1005.

El ajustador 1020 de datos de cuantificación ajusta tres componentes de los datos diferenciales de rotación cuantificados  $\tilde{Q}$  siguiendo la Ecuación (22) en la etapa S1010.

$$I_{ijk} = \tilde{Q} + D_{ijk}, D_{ijk} = (i, j, k)^T \quad (-d \leq i, j, k \leq d \text{ en las que } i, j, k \text{ son enteros}) \quad \dots(22)$$

En la Ecuación (22), i, j y k son variables, que se sumarán a los datos diferenciales de rotación para ajustar los datos diferenciales de rotación, y d se usa para definir el intervalo de i, j y k. A los tres componentes ajustados de los datos diferenciales de rotación cuantificados  $\tilde{Q}$  les da salida el cuantificador inverso 1030.

El cuantificador inverso 1030 cuantifica de manera inversa los datos diferenciales de rotación cuantificados ajustados  $I_{ijk}$  y da salida al valor diferencial de rotación  $\hat{I}_{ijk}$  cuantificado de manera inversa (o los resultados de la cuantificación inversa) al restaurador 1040 del valor diferencial de rotación en la etapa S1020.

El restaurador 1040 del valor diferencial de rotación, que recibe tres componentes del valor diferencial de rotación  $\hat{I}_{ijk}$  cuantificado de manera inversa restaura un primer componente de los datos diferenciales de rotación siguiendo la Ecuación (11) y da salida de un valor diferencial restaurado de rotación a la unidad 1050 de medición de errores en la etapa S1030.

La unidad 1050 de medición de errores comprueba un primer valor componente de un valor diferencial de rotación introducido en la misma. Si el primer valor componente es un número real, la unidad 1050 de medición de errores lleva a cabo una medición de error. Por otra parte, si el primer valor componente es un número imaginario, el procedimiento prosigue a la etapa S1060 en la etapa S1040.

Cuando el primer componente del valor diferencial introducido de rotación es un número real, la unidad 1050 de medición de errores mide un error  $e_{ijk}$  entre un valor diferencial de rotación original y un valor diferencial restaurado de rotación en la etapa S1045 y comprueba si  $e_{ijk}$  es menor que un error ajustado  $e_{i^*j^*k^*}$  en la etapa S1050. El procedimiento de medición de  $e_{ijk}$  entre el valor diferencial de rotación original y el valor diferencial restaurado de rotación se describirá más abajo con referencia a la FIG. 14.

Si el error medido  $e_{ijk}$  es menor que el error ajustado  $e_{i^*j^*k^*}$ ,  $e_{ijk}$  sustituye  $e_{i^*j^*k^*}$  en la etapa S1055 y, a continuación, se comprueba si las variables i, j y k pertenecen o no a un intervalo de ajuste [-d, +d] en la etapa S1060. Si las variables i, j y k pertenecen al intervalo de ajuste [-d, +d], la unidad 1040 de medición de errores lleva a cabo de manera reiterada las etapas S1010 a S1055. En la etapa S1010, durante cada ciclo de las etapas S1010 a S1055, se suma 1 a los componentes segundo a cuarto de los datos diferenciales de rotación cuantificados en forma de bucle anidado.

Por ejemplo, un ajustador 1020 de datos cuantificados mantiene los componentes segundo y tercero, intenta hacer que el primer componente restaurado se convierta en un valor real añadiendo un valor de k al cuarto valor componente mientras se aumenta gradualmente el valor de k sumando continuamente 1 al valor de k ( $-d \leq k \leq d$ ) antes de que el valor de k supere +d, y luego encuentra cuatro valores componentes ajustados, lo que puede minimizar un error entre un valor diferencial introducido de rotación y un valor diferencial restaurado de rotación.

Si el valor de k alcanza +d, el ajustador 1020 de datos cuantificados intenta hacer que el primer componente restaurado se convierta en un valor real inicializando el valor de k con -d y sumando un valor de j al tercer valor

componente mientras se aumenta el valor de j sumando 1 al valor de j ( $-d \leq j \leq d$ ) y sumando un valor de k al cuarto valor componente mientras se aumenta gradualmente el valor de k sumando continuamente 1 al valor de k ( $-d \leq k \leq d$ ) antes de que el valor de k supere +d, y, a continuación, encuentra cuatro valores componentes ajustados, lo que puede minimizar un error entre un valor diferencial introducido de rotación y un valor diferencial restaurado de rotación.

Si el valor de j alcanza +d, el ajustador 1020 de datos cuantificados intenta hacer que el primer componente restaurado se convierta en un valor real inicializando los valores de j y k con -d y sumando un valor de i al segundo valor componente mientras se aumenta el valor de i sumando 1 al valor de i ( $-d \leq i \leq d$ ) y sumando un valor de j al tercer valor componente y sumando un valor de k al cuarto valor componente mientras se aumenta gradualmente el valor de k sumando continuamente 1 al valor de k ( $-d \leq k \leq d$ ) antes de que el valor de k supere +d, y, a continuación, encuentra cuatro valores componentes ajustados, lo que puede minimizar un error entre un valor diferencial introducido de rotación y un valor diferencial restaurado de rotación.

El procedimiento mencionado anteriormente se repite hasta que i, j, k alcanzan +d y el ajustador 1020 de datos cuantificados halla cuatro valores componentes ajustados, lo que puede minimizar un error entre un valor diferencial introducido de rotación y un valor diferencial restaurado de rotación.

La unidad 1050 de medición de errores comprueba si el error ajustado  $e_{i^*j^*k^*}$  es menor que un error final  $e_{g_i^*g_j^*g_k^*}$  en la etapa S1070 mientras se cambian los valores componentes segundo a cuarto. Si  $e_{i^*j^*k^*}$  es menor que  $e_{g_i^*g_j^*g_k^*}$ ,  $e_{i^*j^*k^*}$  sustituye a  $e_{g_i^*g_j^*g_k^*}$  y los datos diferenciales de rotación cuantificados se corrigen siguiendo la Ecuación (23) en la etapa S1075.

$$\tilde{Q}^* = (\tilde{q}_1, \tilde{q}_2, \tilde{q}_3)^T + (i^*, j^*, k^*)^T \quad \dots(23)$$

Después, la unidad 1050 de medición de errores da salida a los datos diferenciales de rotación corregidos al ajustador 1020 de datos cuantificados.

El ajustador 1020 de datos cuantificados fija los valores de las variables i, j y k a -d y lleva a cabo nuevamente las etapas S1010 a S1060 con los datos diferenciales de rotación introducidos en el mismo. A continuación, el ajustador 1020 de datos cuantificados comprueba si existen datos diferenciales de rotación que tengan un error menor con los datos diferenciales de rotación introducidos que un error final almacenado previamente.

Si el error ajustado  $e_{i^*j^*k^*}$  no es menor que el error final  $e_{g_i^*g_j^*g_k^*}$ , la unidad 1050 de medición de errores da salida a los datos diferenciales de rotación cuantificados  $\tilde{Q}_g^* = \tilde{Q}^* = (\tilde{q}_1^*, \tilde{q}_2^*, \tilde{q}_3^*)^T$  correspondientes al error final almacenado actualmente  $e_{g_i^*g_j^*g_k^*}$  a un operador de DPCM circular en la etapa S1080.

En lo que sigue, se describirán un aparato y un procedimiento para decodificar una corriente codificada de bits, en la que codifican los datos de valores clave are usando el aparato para codificar datos de valores clave según la presente invención, con referencia a las FIGURAS 11A a 12.

La FIG. 11A es un diagrama de bloques de un aparato para decodificar una corriente codificada de bits según una realización preferente de la presente invención. Con referencia a la FIG. 11A, el aparato para decodificar una corriente codificada de bits según la presente invención incluye un decodificador 1170 de cabeceras de valor clave, que decodifica la información de cabecera requerida para decodificar datos de valores clave representados por un cuaternión procedentes de una corriente de bits de entrada y proporciona la información decodificada de la cabecera a un decodificador 1100 de datos de valores clave, un decodificador 1110 de entropía, que genera datos diferenciales de rotación sometidos a DPCM circular o datos diferenciales de rotación cuantificados decodificando por entropía datos de valores clave codificados por entropía procedentes de la corriente de bits de entrada, un operador 1120 de DPCM circular inversa, que genera datos diferenciales de rotación cuantificados llevando a cabo una operación de DPCM circular inversa con los datos diferenciales de rotación sometidos a DPCM circular introducidos en el mismo, un cuantificador inverso 1130, que genera un valor diferencial de rotación cuantificando de manera inversa datos diferenciales de rotación cuantificados, y un multiplicador 1140 de cuaterniones, que genera un valor de transformación rotacional de una trama clave actual multiplicando por cuaterniones un valor diferencial de rotación de la trama clave actual por un valor de transformación rotacional de una trama clave anterior.

La FIG. 11B es un diagrama de flujo de un procedimiento para decodificar datos de valores clave según una realización preferente de la presente invención. Con referencia a la FIG. 11B, se introduce una corriente de datos, en la que se codifican datos de valores clave usando el aparato para codificar datos de valores clave según la presente invención, en el decodificador 1170 de cabeceras de valor clave y el decodificador 1110 de entropía del decodificador 1100 de datos de valores clave .

El decodificador 1170 de cabeceras de valor clave decodifica la información de cabecera requerida para decodificar datos de valores clave de la corriente de bits de entrada y proporciona la información decodificada de la cabecera al decodificador 1100 de datos de valores clave en la etapa S1100.

En la información de la cabecera, los datos de valores intraclave primero y segundo cuantificados en cuaterniones se basan en el orden de DPCM y está codificada una pluralidad de banderas usadas para decodificar otros datos de valores clave.

- 5 Si el orden DPCM es 1 (por ejemplo, si nKVDPCMOrder, que es una de la pluralidad de banderas, está fijado a 0), se incluyen los primeros datos de valores clave cuantificados en la información de cabecera como datos de valores intraclave. Si un valor de cuaternión de los primeros datos de valores clave cuantificados de manera inversa satisface  $\hat{Q}_0 = (\hat{q}_{0,0}, \hat{q}_{0,1}, \hat{q}_{0,2}, \hat{q}_{0,3})^T$ , los primeros datos de valores clave cuantificados de manera inversa se calculan siguiendo la Ecuación (24).

$$\begin{aligned}\hat{q}_{0,0} &= \tan\left(\frac{\pi}{4} * \left(\frac{|firstQKV\_S|}{2^{nKVQBit-1} - 1}\right)\right) \\ \hat{q}_{0,1} &= \tan\left(\frac{\pi}{4} * \left(xSign * \frac{|firstQKV\_X|}{2^{nKVQBit-1} - 1}\right)\right) \\ \hat{q}_{0,2} &= \tan\left(\frac{\pi}{4} * \left(ySign * \frac{|firstQKV\_Y|}{2^{nKVQBit-1} - 1}\right)\right) \\ \hat{q}_{0,3} &= \tan\left(\frac{\pi}{4} * \left(zSign * \frac{|firstQKV\_Z|}{2^{nKVQBit-1} - 1}\right)\right)\end{aligned}\quad \dots(24)$$

- 10 En la Ecuación (24), xSign es 1 solo cuando nFirstXSign en la clase OrilDPCMKeyValueHeader, que se describirá más tarde, es 1, y es -1 en otras condiciones. ySign y zSign tienen las mismas relaciones con nFirstYSign y nFirstZSign, respectivamente, que la relación entre xSign y nFirstXSign.

- 15 Los valores componentes restaurados del cuaternión definidos por la Ecuación (24) se convierten en desplazamientos angulares que han de ser usados como un interpolador de orientación. Cada uno de los desplazamientos angulares restaurados a partir de los datos de valores clave puede ser expresado por un vector de cuatro dimensiones  $(\hat{x}_i, \hat{y}_i, \hat{z}_i, \hat{\theta}_i)^T$  en el que i representa datos clave actuales,  $(\hat{x}_i, \hat{y}_i, \hat{z}_i)$  representa el vector de un eje de rotación, y  $\hat{\theta}_i$  representa un ángulo de rotación en el sentido contrario a las agujas del reloj. En consecuencia, los valores componentes restaurados del cuaternión se convierten en desplazamientos angulares siguiendo la Ecuación (25).

$$\begin{aligned}\hat{x}_0 &= \hat{q}_{0,1} * \frac{1}{\text{sen}\left(\frac{\hat{\theta}_0}{2}\right)} \\ \hat{y}_0 &= \hat{q}_{0,2} * \frac{1}{\text{sen}\left(\frac{\hat{\theta}_0}{2}\right)} \\ \hat{z}_0 &= \hat{q}_{0,3} * \frac{1}{\text{sen}\left(\frac{\hat{\theta}_0}{2}\right)} \\ \hat{\theta}_0 &= \arccos(\hat{q}_{0,0}) * 2\end{aligned}\quad \dots(25)$$

- 20 Si el orden de DPCM es 2, por ejemplo, si nKVDPCMOrder se fija a 1, se incluyen en la información de la cabecera los datos de valores clave primeros y segundos. Los primeros datos de valores clave cuantificados se restauran de la misma manera que se ha descrito más arriba. Sin embargo, los segundos datos de valores clave se restauran en un procedimiento diferente. En otras palabras, solo se transmiten tres componentes de los segundos datos de valores clave cuantificados, junto con una corriente codificada de bits, y sus valores no son datos de valores

intraclave, sino valores diferenciales ( $\tilde{Q}_1 = (\tilde{q}_{1,1}, \tilde{q}_{1,2}, \tilde{q}_{1,3})$ ) con los primeros datos de valores clave. Suponiendo que un cuaternión que represente los segundos datos de valores clave de datos de valores clave cuantificados de manera inversa satisface  $\hat{Q}_1 = (\hat{q}_{1,0}, \hat{q}_{1,1}, \hat{q}_{1,2}, \hat{q}_{1,3})^T$ ,  $\hat{Q}_1$  se calcula siguiendo la Ecuación (26).

$$\begin{aligned}\hat{q}_{1,0} &= \sqrt{1 - (\hat{q}_{1,1}^2 + \hat{q}_{1,2}^2 + \hat{q}_{1,3}^2)} \\ \hat{q}_{1,1} &= \tan\left(\frac{\pi}{4} * \left(\text{secondXSign} * \frac{|\text{secondQKV\_X}|}{2^{nKVQBit-1} - 1}\right)\right) \\ \hat{q}_{1,2} &= \tan\left(\frac{\pi}{4} * \left(\text{secondYSign} * \frac{|\text{secondQKV\_Y}|}{2^{nKVQBit-1} - 1}\right)\right) \\ \hat{q}_{1,3} &= \tan\left(\frac{\pi}{4} * \left(\text{secondZSign} * \frac{|\text{secondQKV\_Z}|}{2^{nKVQBit-1} - 1}\right)\right)\end{aligned}\quad \dots(26)$$

5 En la Ecuación (26), secondXSign es 1 únicamente cuando nSecondXSign en la clase OriIDPCMKeyValueHeader es 1, y es -1 en otras condiciones. secondYSign y secondZSign tienen las mismas relaciones con nSecondYSign y nSecondZSign, respectivamente, que la relación entre secondXSign y nSecondXSign. Si un cuaternión  $\hat{Q}_1$  que representa segundos datos de valores clave cuantificados  $\hat{Q}_1 = (\hat{q}_{1,0}, \hat{q}_{1,1}, \hat{q}_{1,2}, \hat{q}_{1,3})^T$ ,  $\hat{Q}_1$  se calcula multiplicando  $\hat{Q}_1$  por  $\hat{Q}_0$ . En otras palabras,  $\hat{Q}_1 = \hat{Q}_1 \times \hat{Q}_0$ .

10 El decodificador 1170 de cabeceras de valor clave da salida a los datos de valores clave decodificados y a la información decodificada de la cabecera al decodificador 1100 de valores clave.

El decodificador 1110 de entropía recibe una corriente de datos, en la que están codificados datos diferenciales de datos de valores clave, y decodifica por entropía la corriente de bits de entrada usando información de decodificación decodificada por el decodificador 1170 de cabeceras de valor clave en las etapas S1120 a S1128.

15 La FIG. 12 es un diagrama que ilustra la estructura de una corriente de datos introducida en el decodificador 1110 de entropía. En la FIG. 12, suponiendo que N (nNumberOfKeyValue) represente el número de datos de valores clave codificados, el número de datos diferenciales de rotación de cada componente incluido en una corriente de datos es N-1 (0, 1, ..., nNumberOfKeyValue-2) cuando el orden de DPCM es 0. Cuando el orden de DPCM es 1, el número de datos diferenciales de rotación de cada uno de los componentes incluidos en la corriente de bits es N-2 (0, 1, ..., nNumberOfKeyValue-3).

20 El decodificador 1110 de entropía recibe x\_keyvalue\_flag, y\_keyvalue\_flag y z\_keyvalue\_flag del decodificador 1170 de cabeceras de valor clave y comprueba si x\_keyvalue\_flag, y\_keyvalue\_flag y z\_keyvalue\_flag están fijadas cada una a 1 en la etapa S1120.

25 Cuando x\_keyvalue\_flag, y\_keyvalue\_flag y z\_keyvalue\_flag están fijadas cada una a 0, se considera que todos los datos de valores clave cuantificados o todos los datos diferenciales de cada componente son iguales a nAllKeyValues en las clases OriIKeyValue-CodingBit. En consecuencia, el decodificador 1110 de entropía decodifica los datos de valores clave de cada uno de los componentes en el mismo valor que nAllKeyValues introducido desde el decodificador 1170 de cabeceras de valor clave y da salida a los datos de valores clave decodificados al operador 1120 de DPCM circular inversa en la etapa S1122.

30 Si x\_keyvalue\_flag, y\_keyvalue\_flag y z\_keyvalue\_flag no están fijadas a 0, por ejemplo si keyvalue\_flag, y\_keyvalue\_flag y z\_keyvalue\_flag están fijadas a 1, el decodificador 1110 de entropía comprueba una entrada de un modo de decodificación por entropía desde el decodificador 1170 de cabeceras de valor clave para decodificar cada componente de datos de valores clave introducido ( $\bar{Q}_i = (\bar{q}_{i,1}, \bar{q}_{i,2}, \bar{q}_{i,3})$ ) en la etapa S1124.

35 Cuando el modo de decodificación por entropía es un modo binario de decodificación, el decodificador 1110 de entropía decodifica una corriente de bits codificada aritméticamente adaptable usando una función decodeSignedAAC() mostrada en la Tabla 1 y da salida a los datos diferenciales de rotación cuantificados al operador 1120 de DPCM circular inversa en la etapa S1126.

Tabla 1

```

void decodeSignedAAC(int *nDecodedValue, int qstep, QState *signContext, QState *valueContext) {
    int b;
    b = qstep - 2;
    int msb = 0;
    do {
        qf_decode(&msb, &valueContext[b]);
        msb = msb << b;
        b--;
    } while (msb == 0 && b >= 0);
    int sgn = 0;
    int rest = 0;
    if(msb != 0) {
        qf_decode(&sgn, signContext);
        while (b >= 0) {
            int temp = 0;
            qf_decode(&temp, &valueContext[b]);
            rest |= (temp << b);
            b--;
        }
    }
    if(sgn)
        *nDecodedValue = -(msb+rest);
    else
        *nDecodedValue = (msb+rest);
}

```

- 5 Por otra parte, cuando el modo de decodificación por entropía no es un modo binario de decodificación, el decodificador 1110 de entropía decodifica la corriente de bits de entrada usando una función decodeUnaryAAC() en la etapa S1128. La función decodeUnaryAAC() decodifica la corriente de bits de entrada leyendo consecutivamente 0 hasta que se lee 1 de la corriente de bits, convirtiendo el número de ceros sucesivos en su valor absoluto, leyendo un bit a continuación del bit '1', y convirtiendo el bit en un bit de signo, y a continuación da salida a los datos diferenciales de rotación cuantificados al operador 1120 de DPCM circular inversa. En la Tabla 2 se muestra un ejemplo de la función decodeUnaryAAC().

Tabla 2

```

void decodeUnaryAAC(int* nDecodedValue, QState* signContext, QState* valueContext) {
    int nBits = -1;
    bit bBit;
    do {
        qf_decode(&bBit, valueContext);
        nBits++;
    } while(bBit == 0);
    if(nBits != 0) {
        qf_decode(&bBit, signContext);
        if(bBit == 0)
            *nDecodedValue = nBits;
        else
            *nDecodedValue = -nBits;
    }
    else
        *nDecodedValue = 0;
}

```

- 10 Se usa una función qf\_decode() adoptada en las funciones anteriormente mencionadas decodeSignedAAC() y decodeUnaryAAC () para leer el bit 1 de una corriente de bits codificada aritméticamente adaptable y está definida por un documento ISO/IEC 14496-2: 1999 Coding of Audio-Visual Objects: Visual, aprobado por la Organización Internacional de Estandarización.

- 15 El operador 1120 de DPCM circular inversa recibe del decodificador 1110 de entropía los datos de valores clave decodificados por entropía y comprueba el orden de DPCM introducido desde el decodificador 1170 de cabeceras de valor clave. Si el orden de DPCM es 0, el operador 1120 de DPCM circular inversa da salida a los datos diferenciales de rotación decodificados  $Q_{i,2}$  al cuantificador inverso 1130 en la etapa S1130, dado que los datos de valores clave decodificados por entropía introducidos desde el decodificador 1110 de entropía son datos diferenciales de rotación cuantificados.

Por otra parte, si el orden de DPCM es 1, el operador 1120 de DPCM circular inversa lleva a cabo una operación de DPCM circular inversa en la etapa S1135, dado que los datos de valores clave decodificados por entropía introducidos desde el decodificador 1110 de entropía son datos diferenciales de rotación sometidos a DPCM circular.

- 5 Suponiendo que nKVQBit represente bits de cuantificación inversa, el operador 1120 de DPCM circular inversa lleva a cabo una operación de DPCM circular inversa con los datos diferenciales de rotación  $Q_{i-2}$  siguiendo la Ecuación (27) y genera datos diferenciales de rotación cuantificados  $Q'_{i-2}$  en la etapa S1135.

$$\begin{aligned} \tilde{Q}'_{i-2} &= \tilde{Q}_{i-2} - (2^{nKVQBits} - 1) \quad (si \tilde{Q}_i \geq 0) \\ \tilde{Q}'_{i-2} &= \tilde{Q}_{i-2} + (2^{nKVQBits} - 1) \quad (si \tilde{Q}_i < 0) \end{aligned} \quad \dots(27)$$

$$(i = 2, \dots, nNumberOfKeyValue - 1)$$

Después, el operador 1120 de DPCM circular inversa obtiene un valor A sometido a DPCM inversa y un valor B sometido a DPCM inversa usando  $Q_{i-2}$  y  $Q'_{i-2}$ , respectivamente, lo que se muestra en la Ecuación (28).

$$\begin{aligned} A &= \tilde{Q}_{i-2} + \tilde{Q}_{i-1} \\ B &= \tilde{Q}'_{i-2} + \tilde{Q}_{i-1} \end{aligned} \quad \dots(28)$$

$$(i = 2, \dots, nNumberOfKeyValue - 1)$$

- 10 El operador 1120 de DPCM circular inversa da salida a B como datos  $\tilde{Q}_i$  sometidos a DPCM circular inversa si  $B + (2^{nKVQBit-1} - 1)$  está entre 0 y un valor máximo en un intervalo de cuantificación. Por otra parte, el operador 1120 de DPCM circular inversa da salida a A como  $\tilde{Q}_i$  si  $B + (2^{nKVQBit-1} - 1)$  es menor que 0 o mayor que el valor máximo en el intervalo de cuantificación.

En la Tabla 3 se muestra un ejemplo de códigos de programa en C++ en los que está escrita la operación anteriormente mencionada del operador 1120 de DPCM circular inversa.

- 15 Tabla 3

```
ICDPCM(int* curIDPCMKeyValue, int deltaKeyValue, int prevICDPCMKeyValue) {
    int circularDelta;
    int tempIDPCMKeyValue;
    prevICDPCMKeyValue += ((1 << (nKVQBit-1))-1);
    if(deltaKeyValue >= 0.0)
        circularDelta = deltaKeyValue - ((1 << nKVQBit)-1);
    else
        circularDelta = deltaKeyValue + ((1 << nKVQBit)-1);

    tempIDPCMKeyValue = circularDelta + prevICDPCMKeyValue;

    if((tempIDPCMKeyValue >= 0.0) && (tempIDPCMKeyValue < ((1 << nKVQBit)-1)))
        *curIDPCMKeyValue = tempIDPCMKeyValue;
    else
        *curIDPCMKeyValue = deltaKeyValue + prevICDPCMKeyValue;

    *curIDPCMKeyValue -= ((1 << (nKVQBit-1))-1);
}
```

El cuantificador inverso 1130 recibe los datos diferenciales de rotación cuantificados  $(\tilde{Q}_i = (\tilde{q}_{i,1}, \tilde{q}_{i,2}, \tilde{q}_{i,3}))$  generados por el operador 1120 de DPCM circular inversa llevando a cabo una operación de DPCM circular inversa, restaura un valor diferencial de rotación  $(\hat{Q}_i = (\hat{q}_{i,0}, \hat{q}_{i,1}, \hat{q}_{i,2}, \hat{q}_{i,3})^T)$  llevando a cabo una operación de cuantificación inversa en  $\tilde{Q}_i$  siguiendo la Ecuación (29), y da salida al valor diferencial de rotación  $\hat{Q}_i$  al multiplicador 1140 de cuaterniones en la etapa S1140.

20

$$\hat{q}_{i,0} = \sqrt{1 - (\hat{q}_{f,1}^2 + \hat{q}_{f,2}^2 + \hat{q}_{f,3}^2)}$$

$$\hat{q}_{i,j} = \tan \left( \frac{\pi}{4} * \left( \text{sgn}(\tilde{q}_{i,j}) * \frac{|\tilde{q}_{i,j}|}{2^{nKVQBit-1} - 1} \right) \right) \quad \dots(29)$$

$$(i = 2, \dots, nNumberOfKeyValue - 1, j = 1, 2, 3)$$

Después, el multiplicador 1140 de cuaterniones recibe el valor diferencial de rotación  $\hat{Q}_i$ . El multiplicador 1140 de cuaterniones restaura un valor de transformación rotacional  $\hat{Q}_i$  (en el que  $\hat{Q}_i = (\hat{q}_{i,0}, \hat{q}_{i,1}, \hat{q}_{i,2}, \hat{q}_{i,3})^T$ ) de una trama clave actual en la etapa S1154 mediante la multiplicación por cuaterniones del valor diferencial introducido de rotación  $\hat{Q}_i$  por un valor de transformación rotacional  $\hat{Q}_{i-1}$  de una trama clave anterior siguiendo la Ecuación (30).

$$\hat{Q}_i = \hat{Q}_i \times \hat{Q}_{i-1} \quad \dots(30)$$

$$(i = 2, \dots, nNumberOfKeyValue - 1)$$

5 Tras restaurar un valor de transformación rotacional, el decodificador 1100 de datos de valores clave comprueba si el valor restaurado de transformación rotacional corresponde a los últimos datos de valores clave en la etapa S1156. Si el valor restaurado de transformación rotacional no corresponde a los últimos datos de valores clave, el decodificador 1100 de datos de valores clave lleva a cabo reiteradamente las etapas S1140 a S1154. Por otra parte, si el valor restaurado de transformación rotacional es el de los últimos datos de valores clave, el decodificador 1100 de datos de valores clave da salida al valor restaurado de transformación rotacional en la etapa S1158.

10 Si el orden de DPCM es 0, el operador 1120 de DPCM circular inversa da salida a los datos diferenciales de rotación cuantificados decodificados  $\tilde{Q}_i$  (en los que  $\tilde{Q}_i = (q_{i,1}, q_{i,2}, q_{i,3})$ ) al cuantificador inverso 1130. A continuación, el cuantificador inverso 1130 genera un valor diferencial de rotación  $\hat{Q}_i$  (en el que  $\hat{Q}_i = (\hat{q}_{i,0}, \hat{q}_{i,1}, \hat{q}_{i,2}, \hat{q}_{i,3})^T$ ) cuantificando de manera inversa los datos diferenciales de rotación  $Q_i$  introducidos desde el operador 1120 de DPCM circular inversa y da salida a  $\hat{Q}_i$  al multiplicador 1140 de cuaterniones en la etapa S1140.

15 DPCM circular inversa y da salida a  $\hat{Q}_i$  al multiplicador 1140 de cuaterniones en la etapa S1140.

$$\hat{q}_{i,0} = \sqrt{1 - (\hat{q}_{f,1}^2 + \hat{q}_{f,2}^2 + \hat{q}_{f,3}^2)}$$

$$\hat{q}_{i,j} = \tan \left( \frac{\pi}{4} * \left( \text{sgn}(\tilde{q}_{i-1,j}) * \frac{|\tilde{q}_{i-1,j}|}{2^{nKVQBit-1} - 1} \right) \right) \quad \dots(31)$$

$$(i = 1, \dots, nNumberOfKeyValue - 1, j = 1, 2, 3)$$

Aunque el orden de DPCM sea 0, el multiplicador 1140 de cuaterniones restaura valores de transformación rotacionales casi de la misma manera (etapas S1150 a S1158) que cuando el orden de DPCM no es 0. Cuando el orden de DPCM no es 0, el multiplicador 1140 de cuaterniones lleva a cabo la multiplicación de cuaterniones siguiendo la Ecuación (32).

$$\hat{Q}_i = \hat{Q}_i \times \hat{Q}_{i-1} \quad \dots(32)$$

$$(i = 1, \dots, nNumberOfKeyValue - 1)$$

20 Las FIGURAS 13A a 13E son diagramas que ilustran ejemplos de códigos de programa en lenguaje SDL en los que se realiza el aparato para decodificar datos de valores clave según la presente invención. En lo que sigue se describirá el significado de cada variable mostrada en las FIGURAS 13A a 13E.

La FIG. 13A es un diagrama que ilustra una clase CompressedOrientationInterpolator. CompressedOrientationInterpolator es una clase superior usada para leer una corriente codificada de bits de un interpolador de orientación. OriKeyValueHeader y OriDPCMKeyValue son clases para la información sobre datos de valores clave correspondientes a los datos de los campos de valores clave en un interpolador convencional de

orientación. Se usa una función `qf_start()` para inicializar un decodificador aritmético antes de leer una corriente de datos, en la que están codificados datos de valores clave.

La FIG. 13B es un diagrama que ilustra una clase `OrilKeyValueHeader`. El significado de cada variable usada en esta clase es el siguiente. `nKVQBit` representa un tamaño de los bits de cuantificación inversa de los datos de valores clave. `nNumKeyValueCodingBit` representa el número de los datos de valores clave. `nKVDPCMOrder` representa el orden de la operación de DPCM inversa usada para decodificar datos de valores clave. Cuando `nKVDPCMOrder = 0`, no es preciso llevar a cabo una operación de DPCM circular inversa. Por otra parte, cuando `nKVDPCMOrder = 1`, se supone que hay que llevar a cabo una operación de DPCM circular inversa.

La FIG. 13C es un diagrama que ilustra una clase `OriDPCMKeyValueHeader`. El significado de cada variable usada en esta clase es el siguiente. `firstQKV_S`, `firstQKV_X`, `firstQKV_Y` y `firstQKV_Z` representan primeros valores de cuatro componentes *s*, *x*, *y* y *z*, respectivamente, que constituyen un cuaternión (*s*, *x*, *y*, *z*) que representa datos de valores clave cuantificados. `nFirstXSign`, `nFirstYSign` y `nFirstZSign` representan los signos de `firstQKV_X`, `firstQKV_Y` y `firstQKV_Z`, respectivamente. `secondQKV_X`, `secondQKV_Y` y `secondQKV_Z` representan segundos valores de los tres componentes *x*, *y* y *z*, respectivamente, y `nSecondXSign`, `nSecondYSign` y `nSecondZSign` representan los signos de `secondQKV_X`, `secondQKV_Y` y `secondQKV_Z`, respectivamente. `bisMoreTwoKVs` indica si hay más de dos datos de valores clave por ser decodificados cuando se supone que debe llevarse a cabo una DPCM circular inversa. `x_keyvalue_flag`, `y_keyvalue_flag` y `z_keyvalue_flag` indican si todos los valores cuantificados de cada uno de los componentes *x*, *y* y *z*, respectivamente, son iguales.

La FIG. 13D es un diagrama que ilustra una clase `OrilKeyValueCodingBit`. El significado de cada variable usada en esta clase es el siguiente. `nKVCodingBit` representa el número de bits usados para almacenar cada componente de todos los datos de valores clave del cuaternión, salvo los datos de valores intraclave (`firstQKV_S`, `firstQKV_X`, `firstQKV_Y`, `firstQKV_Z`, `secondQKV_X`, `secondQKV_Y` y `secondQKV_Z` de la clase `OriDPCMKeyValueHeader`) después de la cuantificación.

`nAllKeyValue` representa un valor cuantificado de cada componente de todos los datos de valores clave cuando `keyvalue_flag` para cada uno de los componentes se fija a 0. `nSign` representa el signo de `nAllKeyValue`. `bisUnaryAAC` representa un procedimiento de cuantificación aritmético adaptable usado para decodificar los valores cuantificados de cada uno de los componentes *x*, *y* y *z*. Si `bisUnaryAAC` está fijado como 'verdadero', se usará una función de decodificación `unaryAAC`. En cambio, si `bisUnaryAAC` está fijado como 'falso', se usará una función de decodificación `binaryAAC`.

La FIG. 13E es un diagrama que ilustra una clase `OriDPCMKeyValue`. El significado de cada valor usado en esta clase es el siguiente. `DeltaKeyValue` se usa para almacenar datos de valores clave cuantificados, incluyendo los tres componentes *x*, *y* y *z* en forma de cuaternión. Los datos de valores clave cuantificados almacenados en `DeltaKeyValue` son decodificados a partir de una corriente de datos usando una función `decodeUnaryAAC` o `decodeSignedAAC`.

`kVXSignContext`, `kVYSignContext` y `kVZSignContext` son contextos usados para decodificar los tres componentes *x*, *y* y *z* de `DeltaKeyValue` usando la función `decodeUnaryAAC` o `decodeSignedAAC`.

`kVXUnaryContext`, `kVYUnaryContext` y `kVZUnaryContext` son contextos usados para decodificar los tres componentes *x*, *y* y *z* de `DeltaKeyValue` usando la función `decodeUnaryAAC`.

`kVXContext`, `kVYContext` y `kVZContext` son contextos usados para decodificar los tres componentes *x*, *y* y *z* de `DeltaKeyValue` usando la función `decodeSignedAAC`.

En lo que sigue se describirá un procedimiento de medición de un error entre un valor diferencial de rotación original y un valor diferencial de rotación restaurado cuantificando de manera inversa un valor diferencial de rotación cuantificado en la unidad 1050 de medición de errores incluida en un cuantificador 1000 según una tercera realización de la presente invención.

El procedimiento de medición de un error entre un valor diferencial de rotación original y un valor diferencial restaurado de rotación según la presente invención también puede ser usado para medir un error entre datos de valores clave de un interpolador de orientación antes de la codificación y datos de valores clave del interpolador de orientación obtenidos decodificando datos de valores clave codificados. En consecuencia, en la siguiente descripción del procedimiento de medición de errores según la presente invención, el valor diferencial de rotación original y el valor diferencial restaurado de rotación corresponderán a los datos de valores clave originales antes de la codificación y a los datos de valores clave obtenidos decodificando los datos de valores clave codificados, respectivamente.

En el procedimiento de codificar datos de valores clave de un interpolador de orientación, durante la cuantificación ocurre un error entre los datos de valores clave originales y los datos de valores clave restaurados. Aquí, el error entre los datos de valores clave originales y los datos de valores clave restaurados está definido por la diferencia

entre un ángulo de rotación obtenido usando los datos de valores clave originales y un ángulo de rotación obtenido a partir de los datos de valores clave restaurados.

En otras palabras, suponiendo que se haga referencia a un dato de los valores clave incluido en un nodo interpolador de orientación y a su dato de los valores clave en un decodificador como  $(\vec{r}, \theta)$  y  $(\vec{r}, \hat{\theta})$ ,

- 5 respectivamente, representando  $\vec{r}$  un eje de rotación y representando  $\theta$  un ángulo de rotación, y satisfaciendo  $\theta \in [-\pi, \pi]$  y que un objeto en un espacio tridimensional se mueva de una posición arbitraria  $\vec{x}$  a una posición arbitraria  $\vec{y}$  y de  $\vec{y}$  a una posición arbitraria  $\hat{y}$  en base a  $(\vec{r}, \theta)$  y  $(\vec{r}, \hat{\theta})$ , respectivamente, por transformación por rotación, un error de cuantificación es la diferencia entre  $\vec{y}$  y  $\hat{y}$ , y satisface  $\vec{e}(\vec{x}) = \vec{y} - \hat{y}$ .  $\vec{x}$ ,  $\vec{y}$  y  $\hat{y}$ , representadas en forma de cuaternión, se muestran en la Ecuación (33).

$$X = (0, \vec{x}), \quad Y = (0, \vec{y}), \quad \hat{Y} = (0, \hat{y}) \quad \dots(33)$$

- 10 Cuando se hace referencia a los cuaterniones que representan a  $(\vec{r}, \theta)$  y  $(\vec{r}, \hat{\theta})$  como Q and  $\hat{Q}$ , pueden derivarse las ecuaciones siguientes.

$$Y = Q \times X \times Q^* \quad \dots(34)$$

$$X = Q^* \times Y \times Q$$

Aquí, A×B representa una multiplicación de cuaterniones, y A\* representa un conjugado de A. En base a las Ecuaciones (33) y (34), puede derivarse la ecuación siguiente.

$$\hat{Y} = \hat{Q} \times X \times \hat{Q}^* = \hat{Q} \times \hat{Q}^* \times Y \times \hat{Q} \times \hat{Q}^* = Q_{error} \times Y \times Q_{error}^* \quad \dots(35)$$

- 15 Aquí,  $Q_{error}$  es un valor que representa la relación entre  $\vec{y}$  y  $\hat{y}$  en términos de transformación de rotación y está definida por la ecuación siguiente.

$$Q_{error} = \hat{Q} \times Q^* \quad \dots(36)$$

Cuando se hace referencia a un ángulo diferencial de rotación entre  $\vec{y}$  y  $\hat{y}$  como  $\theta_{error}$ ,  $\theta_{error}$  puede ser obtenido usando una fórmula de transformación por cuaterniones y la Ecuación (36).

$$\theta_{error} = 2 \cos^{-1} q_{0,error} = 2 \cos^{-1} (\hat{Q} \cdot Q) \quad \theta_{error} \in [0, \pi] \quad \dots(37)$$

- 20 Aquí, X representa una operación de producto interno. La Ecuación (37) define un error instantáneo de cuantificación que ocurre entre todas las tramas clave de la animación en un instante predeterminado de tiempo, y un error instantáneo de cuantificación en un instante predeterminado de tiempo (t) puede ser definido por la ecuación siguiente para derivar una fórmula para obtener un error de cuantificación en un intervalo completo de animación a partir de la Ecuación (37).

$$e(t) = 2 \arccos \{ Q(t) \cdot \hat{Q}(t) \} \quad \dots(38)$$

Un error medio  $E_m$  y un error máximo  $E_p$  pueden derivarse aplicando extensamente la Ecuación (38) a todo el intervalo de la trama clave, durante el cual se está realizando una animación usando un interpolador de orientación.

- 25 Aquí, para obtener el error medio  $E_m$ , debe obtenerse en primer lugar la suma parcial  $E_m^i$  de errores en un intervalo  $[t_{i-1}, t_i]$ , como se muestra en la FIG. 14. En lo sucesivo, se hará referencia a los datos de valores clave decodificados correspondientes a los datos de valores clave originales  $Q_{i-1}$  y  $Q_i$  como  $\hat{Q}_{i-1}$  y  $\hat{Q}_i$ , respectivamente.

- Debido al hecho de que una trayectoria de animación está distorsionada en un espacio tridimensional, el intervalo  $[t_{i-1}, t_i]$  tiene que ser dividido en dos subintervalos  $[t_{i-1}, t_i'']$  y  $[t_i'' - t_i]$  y los dos subintervalos deben ser calculados separadamente el uno del otro. Asumamos aquí que una distancia entre dos trayectorias de animación, o sea, entre un grupo de valores de  $Q_i$  y un grupo de valores de  $\hat{Q}_i$  se minimiza en  $t_i''$  en el intervalo  $[t_{i-1}, t_i]$ . Los errores instantáneos en  $t_{i-1}$  y  $t_i$  se calculan usando las ecuaciones siguientes.
- 30

$$\begin{aligned} e_{i-1} &= 2 \arccos(Q'_{i-1} \cdot \hat{Q}_{i-1}) \\ e_i &= 2 \arccos(Q'_i \cdot \hat{Q}_i) \end{aligned} \quad \dots(39)$$

Además,  $t_i''$  en el intervalo  $[t_{i-1}, t_i]$  es proporcional a  $\frac{e_{i-1}}{e_i}$ , lo que se muestra en la Ecuación (40).

$$t_i'' = t_{i-1} + \frac{e_{i-1}}{e_{i-1} + e_i} (t_i - t_{i-1}) \quad \dots(40)$$

Los valores intermedios del cuaternión y un error instantáneo en  $t_i''$  se calculan siguiendo la Ecuación (41).

$$\begin{aligned} Q_i'' &= SLERP\left(Q_{i-1}, Q_i, \frac{t_i'' - t_{i-1}}{t_i - t_{i-1}}\right) \\ \hat{Q}_i'' &= SLERP\left(\hat{Q}_{i-1}, \hat{Q}_i, \frac{t_i'' - t_{i-1}}{t_i - t_{i-1}}\right) \\ e_i'' &= 2 \cdot \arccos(Q_i'' \cdot \hat{Q}_i'') \end{aligned} \quad \dots(41)$$

Sin embargo, un error instantáneo en un instante arbitrario de tiempo (t) se obtiene siguiendo la Ecuación (42).

$$e(t) = 2 \arccos\{Q(t) \cdot \hat{Q}(t)\} \quad \dots(42)$$

Aquí,  $Q(t) = SLERP\left(Q_{i-1}, Q_i, \frac{t - t_{i-1}}{t_i - t_{i-1}}\right)$  y  $\hat{Q}(t) = SLERP\left(\hat{Q}_{i-1}, \hat{Q}_i, \frac{t - t_{i-1}}{t_i - t_{i-1}}\right)$ . Sin embargo, no es fácil

5 calcular un error instantáneo e(t) en un instante arbitrario de tiempo (t). En consecuencia, e(t) se determina usando aproximación lineal, lo que se muestra en la Aproximación (43).

$$e(t) = \begin{cases} e_{i-1} + \frac{t - t_{i-1}}{t_i'' - t_{i-1}} (e_i'' - e_{i-1}) & (t_{i-1} \leq t \leq t_i'') \\ e_i'' + \frac{t - t_i''}{t_i - t_i''} (e_i - e_i'') & (t_i'' \leq t \leq t_i) \end{cases} \quad \dots(43)$$

Una suma parcial  $E_m^i$  de errores en el intervalo  $[t_{i-1}, t_i]$  y una suma parcial  $E_p^i$  de errores máximos en el intervalo  $[t_{i-1}, t_i]$  pueden obtenerse también usando las Aproximaciones (43) y (44).

$$\begin{cases} E_p^i \equiv \max |e(t)| \\ E_m^i \equiv \int_{t_{i-1}}^{t_i''} e^2(t) dt + \int_{t_i''}^{t_i} e^2(t) dt \end{cases} \quad \dots(44)$$

La Aproximación (45) puede reordenarse como sigue.

$$\begin{cases} E_p^i \cong \max \{|e_{i-1}|, |e_i''|, |e_i|\} \\ E_m^i \cong \frac{1}{3} (t_i'' - t_{i-1}) \left\{ (e_i'')^2 + e_{i-1}^2 + e_i'' e_{i-1} \right\} + \frac{1}{3} (t_i - t_i'') \left\{ e_i^2 + (e_i'')^2 + e_i e_i'' \right\} \end{cases} \quad \dots(45)$$

10 Por último, un error medio  $E_m$  y un error máximo  $E_p$  en un intervalo temporal  $[t_{\min}, t_{\max}]$  se calculan con la Aproximación (46) a continuación.

$$E_m \cong \sqrt{\frac{1}{t_{\max} - t_{\min}} \sum_i E_m^i} \quad \dots(46)$$

$$E_p \cong \max_i E_p^i$$

En consecuencia, la unidad 1050 de medición de errores en el cuantificador 1000 según la tercera realización de la presente invención calcula un error entre los datos de valores clave originales y los datos de valores clave restaurados siguiendo la Ecuación (37) y puede medir de manera más precisa el grado en el que las imágenes están distorsionadas en un espacio de cuaterniones debido a un error de cuantificación.

- 5 La FIG. 15 es un gráfico que muestra el rendimiento del procedimiento para codificar datos de valores clave según la presente invención en comparación con un procedimiento convencional para codificar datos de valores clave. Tal como se muestra en la FIG. 15, según el procedimiento para codificar datos de valores clave de la presente invención, el grado en el que se distorsionan las imágenes cuando se da un número predeterminado de bits requeridos para la codificación puede ser disminuido más considerablemente que en el procedimiento de MPEG\_4  
10 BIFS PMFC convencional.

La presente invención puede ser realizada como códigos legibles por ordenador escritos sobre un medio de grabación legible por ordenador. Aquí, el medio de grabación legible por ordenador incluye cualquier tipo de medio de grabación que pueda ser leído por un sistema informático. Por ejemplo, el medio de grabación legible por ordenador incluye una ROM, una RAM, un CD-ROM, una cinta magnética, un disquete, un almacenamiento óptico  
15 de datos, una onda portadora (transmisión a través de Internet). El medio de grabación legible por ordenador puede estar descentralizado en sistemas de ordenadores conectados por red, y un ordenador puede leer el medio de grabación de manera descentralizada.

Aunque la presente invención se ha mostrado y descrito en particular con referencia a realizaciones preferentes de la misma, las personas expertas en la técnica entenderán que pueden hacerse en las mismas diversos cambios en  
20 forma y en detalles sin apartarse del alcance de la invención según es definida en las reivindicaciones adjuntas.

El procedimiento para la codificación y la decodificación de datos de valores clave de un interpolador de orientación según la presente invención puede reducir de manera efectiva la redundancia entre los datos de valores clave sucesivos que han de ser codificados codificando valores diferenciales de rotación que representan diferencias entre los valores de transformación rotacionales de tramas clave.

- 25 Además, según la presente invención, es posible evitar que un objeto sea restaurado para que rote en una dirección opuesta a una dirección original debido a un error de cuantificación corrigiendo directamente la dirección de rotación del objeto.

Además, dado que, en la presente invención, solo se codifican los componentes segundo a cuarto, que tienen valores relativamente pequeños, entre cuatro componentes que constituyen un cuaternión que representa los datos de valores clave, y se restaura un primer componente usando los componentes segundo a cuarto, que son ajustados para que el primer componente pueda ser restaurado a un número real, resulta posible reducir considerablemente el  
30 tamaño de los datos requeridos para la codificación.

## REIVINDICACIONES

1. Un aparato de generación de datos diferenciales de rotación para un interpolador de orientación de una animación gráfica basada en tramas clave, estando adaptado el aparato para generar, usando un valor de transformación rotacional de una trama clave actual y un valor restaurado de transformación rotacional de una trama clave anterior, un valor diferencial de rotación usado para rotar el objeto en hasta una diferencia entre una transformación rotacional aplicada al objeto en la trama clave actual por los datos de valores clave y una transformación rotacional aplicada al objeto en la trama clave anterior por los datos de valores clave, y para producir datos diferenciales de rotación cuantificando el valor diferencial de rotación.
2. El aparato de la reivindicación 1 que comprende:
  - un primer multiplicador (310, 810) de cuaterniones, que genera un valor diferencial de rotación usado para rotar un objeto en hasta una diferencia entre una transformación rotacional aplicada al objeto en una trama clave actual por un valor de transformación rotacional de la trama actual y una transformación rotacional aplicada al objeto en una trama clave anterior por un valor restaurado de transformación rotacional de la trama anterior;
  - una unidad (340, 840, 1000) de cuantificación, que genera datos diferenciales de rotación cuantificando el valor diferencial de rotación;
  - una unidad (350, 850) de cuantificación inversa, que genera un valor diferencial restaurado de rotación cuantificando de manera inversa los datos diferenciales de rotación; y
  - un segundo multiplicador (370, 870) de cuaterniones, que genera un valor restaurado de transformación rotacional de la trama actual mediante la multiplicación por cuaterniones del valor diferencial restaurado de rotación por un valor de transformación rotacional de la trama anterior.
3. El aparato de la reivindicación 2 en el que el primer multiplicador (310, 810) de cuaterniones genera un valor diferencial de rotación mediante la multiplicación por cuaterniones del valor de transformación rotacional de la trama actual por un conjugado del valor restaurado de transformación rotacional de la trama anterior.
4. El aparato de las reivindicaciones 2 o 3 en el que la unidad (340, 840, 1000) de cuantificación lleva a cabo una cuantificación no lineal mediante la cual una región que tiene un valor diferencial de rotación pequeño puede ser cuantificada para que tenga una resolución mayor que una región que tenga un valor diferencial de rotación grande.
5. El aparato de las reivindicaciones 2, 3 o 4 en el que la unidad (340, 840, 1000) de cuantificación cuantifica solamente tres componentes entre los componentes primero a cuarto del valor diferencial de rotación representado por un cuaternión.
6. El aparato de cualquiera de las reivindicaciones 2 a 5 en el que la unidad (340, 840, 1000) de cuantificación ajusta los componentes de los datos diferenciales de rotación para que un valor componente de un valor diferencial de rotación, que se restaura usando los valores componentes de los datos diferenciales de rotación y que no ha sido cuantificado, pueda ser convertido en un número real y da salida a los componentes ajustados de los datos diferenciales de rotación.
7. El aparato de la reivindicación 6 en el que la unidad (340, 840, 1000) de cuantificación genera datos diferenciales de rotación finales sumando/restando una variable en un intervalo predeterminado de ajuste a/de cada uno de los componentes cuantificados de los datos diferenciales de rotación o a/de cada uno de los componentes ajustados de los datos diferenciales de rotación.
8. El aparato de cualquiera de las reivindicaciones 2 a 7 en el que la unidad (1000) de cuantificación comprende:
  - un cuantificador (1010) que genera datos diferenciales de rotación cuantificando tres valores componente de un valor diferencial de rotación introducido en el mismo;
  - un ajustador (1020) de datos cuantificados, que ajusta tres valores componentes de los datos diferenciales de rotación introducidos en el mismo;
  - un cuantificador inverso (1030), que cuantifica de manera inversa los valores componentes ajustados;
  - un restaurador (1040) del valor diferencial de rotación, que genera un valor diferencial restaurado de rotación restaurando un valor componente, que no ha sido cuantificado, usando los tres valores componentes cuantificados de manera inversa; y

una unidad (1050) de medición de errores, que mide un error entre un valor diferencial de rotación introducido en la unidad de cuantificación y el valor diferencial restaurado de rotación y da salida a los datos diferenciales de rotación que tienen componentes ajustados para que el error pueda minimizarse.

- 5 **9.** El aparato de la reivindicación 8 en el que la unidad (1050) de medición de errores renueva un error ajustado con un error actual entre el valor diferencial de rotación introducido en la unidad (1000) de cuantificación y el valor diferencial restaurado de rotación si el error actual es menor que el error ajustado, renueva el error final con el error ajustado si el error ajustado es menor que el error final, y da salida a los datos diferenciales de rotación ajustados correspondientes al error ajustado hacia el ajustador (1020) de datos cuantificados.
- 10 **10.** El aparato de cualquiera de las reivindicaciones 2 a 9 en el que el cuantificador inverso (350, 850) cuantifica de manera inversa valores componentes cuantificados y restaura un valor componente, que no ha sido cuantificado, usando los valores componentes cuantificados de manera inversa.
- 15 **11.** El aparato de cualquiera de las reivindicaciones 2 a 10 que comprende, además, un detector (830) de error de la dirección de rotación, que genera un primer valor diferencial de rotación entre el valor de transformación rotacional de la trama actual y el valor de transformación rotacional de la trama anterior, un segundo valor diferencial de rotación entre el valor de transformación rotacional de la trama anterior y el valor restaurado de transformación rotacional de la trama anterior introducido desde el segundo multiplicador de cuaterniones, y un tercer valor diferencial de rotación entre el valor de transformación rotacional de la trama actual y el valor restaurado de transformación rotacional de la trama anterior y detecta si ha ocurrido o no un error de la dirección de rotación.
- 20 **12.** El aparato de la reivindicación 11 en el que el detector (830) de error de la dirección de rotación determina que el error de la dirección de rotación ha ocurrido cuando la dirección de rotación del objeto obtenida del primer valor diferencial de rotación es opuesta a la dirección de rotación del objeto obtenida del tercer valor diferencial de rotación, la dirección de rotación del objeto obtenida del segundo valor diferencial de rotación es opuesta a la dirección de rotación del objeto obtenida del tercer valor diferencial de rotación, y los ángulos de rotación obtenidos de los valores diferenciales de rotación primero a tercero superan un ángulo crítico predeterminado de rotación.
- 25 **13.** El aparato de cualquiera de las reivindicaciones 2 a 12 que, además, comprende:  
 un detector (830) de error de la dirección de rotación, que recibe del segundo multiplicador de cuaterniones el valor de transformación rotacional de la trama actual y el valor restaurado de transformación rotacional de la trama anterior y detecta si ha ocurrido o no un error de la dirección de rotación de modo que una dirección original de rotación del objeto sea opuesta a una dirección decodificada de rotación del objeto;  
 un corrector (815) de la dirección de rotación, que ajusta el valor diferencial de rotación introducido desde el primer multiplicador (810) de cuaterniones para que la dirección original de rotación del objeto pueda ser la misma que la dirección decodificada de rotación del objeto y da salida al valor diferencial de rotación ajustado; y  
 un selector (835) de la dirección de rotación, que da salida al valor de transformación rotacional introducido desde el primer multiplicador (810) de cuaterniones o al valor diferencial de rotación ajustado introducido desde el corrector (815) de la dirección de rotación a la unidad (840) de cuantificación, dependiendo del resultado de la detección llevada a cabo en el detector (830) de error de la dirección de rotación.
- 30 **14.** El aparato de la reivindicación 13 en el que la unidad (1000) de cuantificación comprende:  
 un cuantificador (1010) que genera datos diferenciales de rotación cuantificando tres valores componentes de un valor diferencial de rotación introducido en el mismo;  
 un ajustador (1020) de datos cuantificados, que ajusta tres valores componentes de los datos diferenciales de rotación introducidos en el mismo;  
 un cuantificador inverso (1030), que cuantifica de manera inversa los valores componentes ajustados;  
 un restaurador (1040) del valor diferencial de rotación, que genera un valor diferencial restaurado de rotación restaurando un valor componente, que no ha sido cuantificado, usando los tres valores componentes cuantificados de manera inversa; y  
 una unidad (1050) de medición de errores, que mide un error entre un valor diferencial de rotación introducido en la unidad de cuantificación y el valor diferencial restaurado de rotación y da salida a los datos diferenciales de rotación que tienen componentes ajustados para que el error pueda minimizarse.
- 35  
40  
45  
50

15. Un aparato para codificar datos de valores clave de un interpolador de orientación que representan una transformación rotacional de un objeto en tramas clave de una animación gráfica basada en tramas clave, comprendiendo el aparato:
- 5 el aparato generador de datos diferenciales de rotación de la reivindicación 1; y  
un codificador (450) de entropía, que codifica por entropía los datos diferenciales de rotación.
16. El aparato de la reivindicación 15 en el que el generador de datos diferenciales de rotación genera el valor diferencial de rotación mediante la multiplicación por cuaterniones del valor de transformación rotacional de la trama clave actual por un conjugado del valor restaurado de transformación rotacional de la trama clave anterior.
- 10 17. El aparato de las reivindicaciones 15 o 16 en el que el generador de datos diferenciales de rotación lleva a cabo una cuantificación no lineal mediante la cual una región que tiene un valor diferencial de rotación pequeño puede ser cuantificada para que tenga una resolución mayor que una región que tenga un valor diferencial de rotación grande.
- 15 18. El aparato de las reivindicaciones 15, 16 o 17 en el que el generador de datos diferenciales de rotación ajusta los valores componentes de los datos diferenciales de rotación para que los valores componentes no cuantificados de los datos diferenciales de rotación puedan ser restaurados a números reales cuantificando de manera inversa los valores componentes cuantificados de los datos diferenciales de rotación.
- 20 19. El aparato de la reivindicación 18 en el que el generador de datos diferenciales de rotación genera los datos diferenciales de rotación para que estén codificados por entropía sumando/restando una variable en un intervalo predeterminado de ajuste a/de cada uno de los valores componentes cuantificados de los datos diferenciales de rotación o a/de cada uno de los valores componentes ajustados de los datos diferenciales de rotación.
- 25 20. El aparato de cualquiera de las reivindicaciones 15 a 19 en el que el generador de datos diferenciales de rotación comprende:
- un cuantificador (1010) que genera datos diferenciales de rotación cuantificando tres valores componentes del valor diferencial de rotación;
- un ajustador (1020) de datos cuantificados, que ajusta tres valores componentes de los datos diferenciales de rotación introducidos en el mismo;
- un cuantificador inverso (1030), que cuantifica de manera inversa los valores componentes ajustados;
- 30 un restaurador (1040) del valor diferencial de rotación, que restaura un valor componente, que no ha sido cuantificado, usando los tres valores componentes cuantificados de manera inversa y genera así un valor diferencial restaurado de rotación; y
- 35 una unidad (1050) de medición de errores, que mide un error entre un valor diferencial de rotación introducido en el cuantificador y el valor diferencial restaurado de rotación y da salida a los datos diferenciales de rotación que tienen componentes ajustados para que el error pueda minimizarse.
- 40 21. El aparato de la reivindicación 20 en el que la unidad (1050) de medición de errores renueva un error ajustado con un error actual entre el valor diferencial de rotación introducido en el cuantificador (1010) y el valor diferencial restaurado de rotación si el error actual es menor que el error ajustado, renueva el error final con el error ajustado si el error ajustado es menor que el error final, y da salida a los datos diferenciales de rotación ajustados correspondientes al error ajustado hacia el ajustador (1020) de datos cuantificados.
- 45 22. El aparato de cualquiera de las reivindicaciones 15 a 21 en el que el generador de datos diferenciales de rotación comprende:
- un primer multiplicador (310, 810) de cuaterniones, que genera el valor diferencial de rotación usando el valor de transformación rotacional de la trama clave actual y el valor restaurado de transformación rotacional de la trama clave anterior;
- un cuantificador (340, 840, 1000) que genera datos diferenciales de rotación cuantificando el valor diferencial de rotación;
- un cuantificador inverso (350, 850) que genera un valor diferencial restaurado de rotación cuantificando de manera inversa los datos diferenciales de rotación; y

un segundo multiplicador (370, 870) de cuaterniones, que genera un valor diferencial restaurado de rotación de la trama actual mediante la multiplicación por cuaterniones del valor diferencial restaurado de rotación por un valor de transformación rotacional de la trama clave anterior.

- 5       **23.** El aparato de la reivindicación 22 en el que el cuantificador (340, 840, 1000) cuantifica solamente tres valores componentes entre los valores componentes primero a cuarto del valor diferencial de rotación representado por un cuaternión.
- 10       **24.** El aparato de cualquiera de las reivindicaciones 15 a 23 en el que el generador de datos diferenciales de rotación genera un primer valor diferencial de rotación entre el valor de transformación rotacional de la trama clave actual y el valor de transformación rotacional de la trama clave anterior, un segundo valor diferencial de transformación rotacional de la trama clave anterior, y un tercer valor diferencial de rotación entre el valor de transformación rotacional de la trama clave actual y el valor restaurado de transformación rotacional de la trama clave anterior y detecta si ha ocurrido o no un error de la dirección de rotación de modo que el objeto rote en una dirección opuesta a una dirección original.
- 15       **25.** El aparato de la reivindicación 24 en el que el generador de datos diferenciales de rotación determina que el error de la dirección de rotación ha ocurrido cuando la dirección de rotación del objeto obtenida del primer valor diferencial de rotación es opuesta a la dirección de rotación del objeto obtenida del tercer valor diferencial de rotación, la dirección de rotación del objeto obtenida del segundo valor diferencial de rotación es opuesta a la dirección de rotación del objeto obtenida del tercer valor diferencial de rotación, y los ángulos de rotación obtenidos de los valores diferenciales de rotación primero a tercero superan un ángulo crítico predeterminado de rotación.
- 20       **26.** El aparato de cualquiera de las reivindicaciones 15 a 25 en el que el generador de datos diferenciales de rotación comprende:
- 25               un detector (830) de error de la dirección de rotación, que detecta si ha ocurrido o no un error de la dirección de rotación de modo que una dirección original de rotación del objeto sea opuesta a una dirección decodificada de rotación del objeto en base al valor de transformación rotacional de la trama clave actual y el valor restaurado de transformación rotacional de la trama clave anterior;
- 30               un corrector (815) de la dirección de rotación, que ajusta el valor diferencial de rotación para que la dirección decodificada de rotación del objeto pueda ser la misma que la dirección original de rotación del objeto; y
- un selector (835) de la dirección de rotación, que selecciona el valor de transformación rotacional o el valor diferencial de rotación introducido desde el corrector (815) de la dirección de rotación como datos diferenciales que deben ser cuantificados, dependiendo del resultado de la detección introducida desde el detector (830) de error de la dirección de rotación.
- 35       **27.** El aparato de la reivindicación 26 en el que el generador de datos diferenciales de rotación comprende, además, una unidad (1000) de cuantificación, que cuantifica los valores diferenciales seleccionados por el selector de la dirección de rotación y da salida a los datos diferenciales cuantificados y la unidad (1000) de cuantificación comprende:
- 40               un cuantificador (1010) que genera datos diferenciales de rotación cuantificando tres valores componentes de un valor diferencial de rotación introducido en el mismo;
- un ajustador (1020) de datos cuantificados, que ajusta tres valores componentes de los datos diferenciales de rotación introducidos en el mismo;
- un cuantificador inverso (1030), que cuantifica de manera inversa los valores componentes ajustados;
- 45               un restaurador (1040) del valor diferencial de rotación, que genera un valor diferencial restaurado de rotación restaurando un valor componente, que no ha sido cuantificado, usando los tres valores componentes cuantificados de manera inversa; y
- una unidad (1050) de medición de errores, que mide un error entre un valor diferencial de rotación introducido en la unidad de cuantificación y el valor diferencial restaurado de rotación y da salida a los datos diferenciales de rotación ajustados en base al error medido.
- 50       **28.** El aparato de la reivindicación 27 en el que el ajustador (1020) de datos cuantificados genera datos diferenciales de rotación ajustados sumando/restando una variable en un intervalo predeterminado de ajuste a/de cada uno de los valores componentes cuantificados de los datos diferenciales de rotación introducidos desde el cuantificador (1010) o a/de cada uno de los valores componentes ajustados de los datos diferenciales de rotación introducidos desde la unidad (1050) de medición de errores.

- 5 29. El aparato de las reivindicaciones 27 o 28 en el que la unidad (1050) de medición de errores renueva un error ajustado con un error actual entre el valor diferencial de rotación introducido en el cuantificador (1010) y el valor diferencial restaurado de rotación si el error actual es menor que el error ajustado, renueva el error final con el error ajustado si el error ajustado es menor que el error final, y da salida a los datos diferenciales de rotación ajustados correspondientes al error ajustado hacia el ajustador (1020) de datos cuantificados.
- 10 30. El aparato de cualquiera de las reivindicaciones 15 a 29 que comprende, además, un operador (400) de DPCM que genera datos diferenciales que tienen un tamaño reducido llevando a cabo una operación de DPCM lineal con los datos diferenciales de rotación y llevando a cabo, a continuación, una operación predeterminada con los datos diferenciales sometidos a DPCM, comprendiendo la operación la representación de los datos que tienen un valor absoluto mayor que un valor máximo en un intervalo de cuantificación mediante datos de un valor absoluto menor y polaridad opuesta,
- en el que el codificador (450) de entropía codifica por entropía datos diferenciales de rotación sometidos a DPCM circular.
- 15 31. El aparato de la reivindicación 30 en el que el operador (400) de DPCM circular da salida a los menores entre los datos diferenciales sometidos a DPCM lineal y diferencias entre los datos diferenciales sometidos a DPCM lineal y el valor máximo del intervalo de cuantificación.
- 20 32. El aparato de cualquiera de las reivindicaciones 30 o 31 en el que el generador de datos diferenciales de rotación comprende:
- un primer multiplicador (310, 810) de cuaterniones, que genera el valor diferencial de rotación usando un valor de transformación rotacional de una trama clave actual y un valor restaurado de transformación rotacional de una trama clave anterior;
- un cuantificador (340, 840, 1000) que genera datos diferenciales de rotación cuantificando el valor diferencial de rotación;
- 25 un cuantificador inverso (350, 850) que genera un valor diferencial restaurado de rotación cuantificando de manera inversa los datos diferenciales de rotación; y
- un segundo multiplicador (370, 870) de cuaterniones, que genera un valor diferencial restaurado de transformación rotacional de la trama actual mediante la multiplicación por cuaterniones del valor diferencial restaurado de rotación por un valor de transformación rotacional de la trama clave anterior.
- 30 33. El aparato de cualquiera de las reivindicaciones 15 a 32 en el que el codificador (450) de entropía codifica datos diferenciales de rotación para ser codificados en una serie de primeras cifras, el número de las cuales se corresponde con la magnitud de los datos diferenciales de rotación que han de codificarse, y una segunda cifra de un bit que indica que termina la serie de la primera cifra.
34. El aparato de cualquiera de las reivindicaciones 15 a 33 en el que el codificador (450) de entropía codifica datos diferenciales de rotación que han de ser codificados en cada plano de bits.
- 35 35. Un aparato para decodificar una corriente de bits en el que se codifican datos de valores clave de un interpolador de orientación que representan la transformación rotacional de un objeto en tramas clave, comprendiendo el aparato:
- un decodificador (1110) de entropía que genera datos diferenciales de rotación sometidos a DPCM circular o datos diferenciales de rotación cuantificados decodificando por entropía la corriente de bits;
- 40 un operador (1120) de DPCM circular inversa que genera datos diferenciales de rotación cuantificados llevando a cabo una operación de DPCM circular inversa con los datos diferenciales de rotación introducidos desde el decodificador (1110) de entropía dependiendo del orden de la DPCM decodificada de la corriente de bits, comprendiendo la operación de DPCM circular inversa la sustitución de datos sometidos a DPCM que tienen un valor absoluto menor que un valor máximo en un intervalo de cuantificación por datos sometidos a DPCM de valor absoluto mayor y polaridad opuesta y llevar a cabo una operación de DPCM lineal inversa;
- 45 un cuantificador inverso (1130) que genera datos diferenciales de rotación, usados para rotar un objeto en hasta una diferencia entre una transformación rotacional aplicada al objeto en la trama clave actual por los datos de valores clave y una transformación rotacional aplicada al objeto en la trama clave anterior por los datos de valores clave, cuantificando de manera inversa los datos diferenciales de rotación cuantificados; y
- 50 un multiplicador (1140) de cuaterniones que genera un valor de transformación rotacional de una trama clave actual multiplicando por cuaterniones un valor diferencial de rotación de la trama clave actual por un valor de transformación rotacional decodificado de una trama clave anterior.

36. El aparato de la reivindicación 35 en el que el decodificador (1110) de entropía decodificada cada componente de los datos de valores clave de las tramas clave en valores predeterminados decodificados de la corriente de bits si banderas de valores clave decodificadas de la corriente de bits indican que cada componente de los datos diferenciales de todas las tramas clave tienen el mismo valor.
- 5 37. El aparato de las reivindicaciones 35 o 36 en el que la operación de DPCM circular comprende sumar a los datos diferenciales de rotación sometidos a DPCM circular o restar de los mismos un valor máximo en un intervalo de cuantificación de los datos diferenciales de rotación.
38. El aparato de las reivindicaciones 35, 36 o 37 en el que el cuantificador inverso (1130) genera un valor diferencial de rotación cuantificando de manera inversa tres valores componentes de un cuaternión unitario y usando los tres valores componentes restaurados.
- 10 39. Un procedimiento de generación de datos diferenciales de rotación para generar datos diferenciales de valores de transformación rotacionales que representan la rotación de un objeto en tramas clave de una animación gráfica basada en tramas clave, comprendiendo el procedimiento:
- 15 (a) generar un valor diferencial de rotación usado para rotar el objeto en hasta una diferencia entre una transformación rotacional aplicada al objeto en la trama clave actual por los datos de valores clave y una transformación rotacional aplicada al objeto en la trama clave anterior por los datos de valores clave; y
- (b) generar diferenciales de rotación cuantificando el valor diferencial de rotación.
- 20 40. El procedimiento de la reivindicación 39 en el que la etapa (a) comprende la generación de un valor diferencial de rotación usado para rotar el objeto en hasta una diferencia entre transformaciones rotacionales aplicadas al objeto en cada una de las tramas usando un valor de transformación rotacional de una trama actual y un valor restaurado de transformación rotacional de una trama anterior, y que comprende las etapas de:
- (c) generar un valor diferencial restaurado de rotación cuantificando de manera inversa el valor diferencial de rotación; y
- 25 (d) generar un valor restaurado de transformación rotacional de una trama actual mediante la multiplicación por cuaterniones del valor diferencial restaurado de rotación por un valor de transformación rotacional de una trama anterior y llevar a cabo nuevamente las etapas (a) a (d) con el valor restaurado de transformación rotacional de la trama actual y un valor de transformación rotacional de una trama posterior.
- 30 41. El procedimiento de la reivindicación 40 en el que, en la etapa (a), se genera un valor diferencial de rotación mediante la multiplicación por cuaterniones del valor de transformación rotacional de la trama actual por un conjugado del valor restaurado de transformación rotacional de la trama anterior.
42. El procedimiento de las reivindicaciones 40 o 41 en el que, en la etapa (b), se lleva a cabo una cuantificación no lineal para que una región que tiene un valor diferencial de rotación pequeño pueda ser cuantificada para que tenga una resolución mayor que una región que tenga un valor diferencial de rotación grande.
- 35 43. El procedimiento de las reivindicaciones 40, 41 o 42 en el que, en la etapa (b), solo se cuantifican tres componentes entre los valores componentes primero a cuarto del valor diferencial de rotación representado por un cuaternión.
- 40 44. El procedimiento de las reivindicaciones 40, 41, 42 o 43 en el que, en la etapa (b), los valores componentes cuantificados de los datos diferenciales de rotación se ajustan para que un valor componente de un valor diferencial de rotación, que no ha sido cuantificado, sea restaurado a un número real cuantificando de manera inversa los valores componentes de los datos diferenciales de rotación.
- 45 45. El procedimiento de la reivindicación 44 en el que, en la etapa (b), se generan datos diferenciales de rotación finales sumando/restando una variable en un intervalo predeterminado de ajuste a/de cada uno de los componentes cuantificados de los datos diferenciales de rotación o a/de cada uno de los componentes ajustados de los datos diferenciales de rotación.
46. El procedimiento de cualquiera de las reivindicaciones 40 a 45 en el que la etapa (b) comprende:
- (b1) generar datos diferenciales de rotación cuantificando tres valores componentes del valor diferencial de rotación;
- (b2) ajustar tres valores componentes de los datos diferenciales de rotación;
- (b3) cuantificar de manera inversa valores componentes ajustados;

- (b4) generar un valor diferencial restaurado de rotación restaurando un valor componente, que no ha sido cuantificado, usando los tres valores componentes cuantificados de manera inversa; y
- (b5) medir un error entre el valor diferencial de rotación y el valor diferencial restaurado de rotación y dar salida a datos diferenciales de rotación que tienen valores componentes ajustados para que pueda minimizarse el error.
- 5
- 47.** El procedimiento de la reivindicación 46 en el que la etapa (b5) comprende:
- (b51) renovar un error ajustado con un error actual entre el valor diferencial de rotación y el valor diferencial restaurado de rotación si el error actual es menor que el error ajustado;
- (b52) llevar a cabo de forma reiterada las etapas (b2) a (b51) hasta que la variable sumada o restada del valor diferencial de rotación en la etapa (b2) esté más allá del intervalo predeterminado de ajuste; y
- (b53) renovar un error final con el error ajustado si el error ajustado es menor que el error final, llevando a cabo las etapas (b2) a (b52) con datos diferenciales de rotación ajustados correspondientes al error ajustado, y dando salida a datos diferenciales de rotación ajustados correspondientes al error final.
- 10
- 48.** El procedimiento de cualquiera de las reivindicaciones 40 a 47 en el que, en la etapa (c), se restaura un valor componente de un valor diferencial de rotación, que no ha sido cuantificado, cuantificando de manera inversa los datos diferenciales de rotación.
- 15
- 49.** El procedimiento de cualquiera de las reivindicaciones 40 a 48 en el que la etapa (a) comprende además la detección (a2) de un error de la dirección de rotación generando un primer valor diferencial de rotación entre el valor de transformación rotacional de la trama actual y el valor de transformación rotacional de la trama anterior, un segundo valor diferencial de rotación entre el valor de transformación rotacional de la trama anterior y el valor restaurado de transformación rotacional de la trama anterior, y un tercer valor diferencial de rotación entre el valor de transformación rotacional de la trama actual y el valor restaurado de transformación rotacional de la trama anterior.
- 20
- 50.** El procedimiento de la reivindicación 49 en el que, en la etapa (a2), se determina que el error de la dirección de rotación ocurrió cuando la dirección de rotación del objeto obtenida del primer valor diferencial de rotación es opuesta a la dirección de rotación del objeto obtenida del tercer valor diferencial de rotación, la dirección de rotación del objeto obtenida del segundo valor diferencial de rotación es opuesta a la dirección de rotación del objeto obtenida del tercer valor diferencial de rotación, y los ángulos de rotación obtenidos de los valores diferenciales de rotación primero a tercero superan un ángulo crítico predeterminado de rotación.
- 25
- 51.** Un procedimiento para codificar datos de valores clave de un interpolador de orientación que representan la rotación de un objeto en cada trama clave de una animación gráfica basada en tramas clave, comprendiendo el procedimiento:
- 30
- el procedimiento de generación de datos diferenciales de rotación de la reivindicación 39, seguido por una etapa de
- (c) codificación por entropía de los datos diferenciales de rotación.
- 35
- 52.** El procedimiento de la reivindicación 51 en el que, en la etapa (a), se genera el valor diferencial de rotación mediante la multiplicación por cuaterniones del valor de transformación rotacional de la trama clave actual por el valor restaurado de transformación rotacional de la trama clave anterior.
- 53.** El procedimiento de las reivindicaciones 51 o 52 en el que, en la etapa (b), se lleva a cabo una cuantificación no lineal para que una región que tiene un valor diferencial de rotación pequeño pueda ser cuantificada para que tenga una resolución mayor que una región que tiene un valor diferencial de rotación grande.
- 40
- 54.** El procedimiento de las reivindicaciones 51, 52 o 53 en el que, en la etapa (b), se ajustan los valores componentes de los datos diferenciales de rotación para que un valor componente de un valor diferencial de rotación, que se restaura cuantificando de manera inversa los valores componentes cuantificados de los datos diferenciales de rotación y que no ha sido cuantificado, pueda ser convertido en un número real.
- 45
- 55.** El procedimiento de la reivindicación 54 en el que, en la etapa (b), se generan los datos diferenciales de rotación que han de ser codificados por entropía sumando/restando una variable en un intervalo predeterminado de ajuste a/de cada uno de los valores componentes de los datos diferenciales de rotación o a/de cada uno de los valores componentes ajustados de los datos diferenciales de rotación.
- 50
- 56.** El procedimiento de cualquiera de las reivindicaciones 51 a 55 en el que la etapa (b) comprende:

- (b1) generar datos diferenciales de rotación cuantificando tres valores componentes del valor diferencial de rotación;
- (b2) ajustar tres valores componentes de los datos diferenciales de rotación;
- (b3) cuantificar de manera inversa valores componentes ajustados;
- 5 (b4) generar un valor diferencial restaurado de rotación restaurando un valor componente usando los tres valores componentes cuantificados de manera inversa; y
- (b5) medir un error entre el valor diferencial de rotación y el valor diferencial restaurado de rotación y determinar los datos diferenciales de rotación que tienen valores componentes ajustados para que pueda minimizarse el error como datos diferenciales de rotación que han de ser codificados por entropía.
- 10 **57.** El procedimiento de la reivindicación 56 en el que la etapa (b5) comprende:
- (b51) renovar un error ajustado con un error actual entre el valor diferencial de rotación y el valor diferencial restaurado de rotación si el error actual es menor que el error ajustado;
- (b52) llevar a cabo de forma reiterada las etapas (b2) a (b51) hasta que la variable sumada o restada del valor diferencial de rotación esté más allá del intervalo predeterminado de ajuste; y
- 15 (b53) renovar un error final con el error ajustado si el error ajustado es menor que el error final, llevando a cabo las etapas (b2) a (b52) con datos diferenciales de rotación ajustados correspondientes al error ajustado, y determinar datos diferenciales de rotación ajustados correspondientes al error final como datos diferenciales de rotación que han de ser codificados por entropía.
- 58.** El procedimiento de cualquiera de las reivindicaciones 51 a 57 en el que la etapa (a) comprende:
- 20 (a1) generar el valor diferencial de rotación usando un valor de transformación rotacional de la trama clave actual y un valor restaurado de transformación rotacional de la trama clave anterior
- y en el que la etapa (b) comprende:
- (b1) generar datos diferenciales de rotación cuantificando el valor diferencial de rotación;
- (b2) generar un valor diferencial restaurado de rotación cuantificando de manera inversa los datos diferenciales de rotación; y
- 25 (b3) generar un valor restaurado de transformación rotacional de la trama clave actual mediante la multiplicación por cuaterniones del valor diferencial restaurado de rotación por un valor de transformación rotacional de la trama clave anterior.
- 59.** El procedimiento de la reivindicación 58 en el que, en la etapa (b1), solo se cuantifican tres componentes entre los componentes primero a cuarto del valor diferencial de rotación representado por un cuaternión.
- 30 **60.** El procedimiento de cualquiera de las reivindicaciones 51 a 59 en el que, en la etapa (a), se generan un primer valor diferencial de rotación entre el valor de transformación rotacional de la trama clave actual y el valor de transformación rotacional de la trama clave anterior, un segundo valor diferencial de rotación entre el valor de transformación rotacional de la trama clave anterior y el valor restaurado de transformación rotacional de la trama clave anterior, y un tercer valor diferencial de rotación entre el valor de transformación rotacional de la trama clave actual y el valor restaurado de transformación rotacional de la trama clave anterior, y se detecta así si ha ocurrido o no un error de la dirección de rotación, lo que causa que la dirección de rotación decodificada del objeto sea opuesta a la dirección de rotación original.
- 35 **61.** El procedimiento de la reivindicación 60 en el que, en la etapa (a), se determina que el error de la dirección de rotación ocurrió cuando la dirección de rotación del objeto obtenida del primer valor diferencial de rotación es opuesta a la dirección de rotación del objeto obtenida del tercer valor diferencial de rotación, la dirección de rotación del objeto obtenida del segundo valor diferencial de rotación es opuesta a la dirección de rotación del objeto obtenida del tercer valor diferencial de rotación, y los ángulos de rotación obtenidos de los valores diferenciales de rotación primero a tercero superan un ángulo crítico predeterminado de rotación.
- 40 **62.** El procedimiento de cualquiera de las reivindicaciones 51 a 61 en el que la etapa (a) comprende:
- (a1) detectar si ha ocurrido o no un error de la dirección de rotación de modo que una dirección original de rotación del objeto sea opuesta a una dirección decodificada de rotación del objeto dependiendo del valor de transformación rotacional de la trama clave actual y del valor restaurado de transformación rotacional de la trama clave anterior;
- 45

- (a2) ajustar el valor diferencial de rotación para que la dirección original de rotación del objeto pueda ser la misma que la dirección decodificada de rotación del objeto; y
- (a3) seleccionar el valor diferencial de rotación o el valor diferencial de rotación ajustado en la etapa (a2) como datos diferenciales que han de ser cuantificados, dependiendo del resultado de la detección llevada a cabo en la etapa (a1).
- 5
- 63.** El procedimiento de la reivindicación 62 en el que la etapa (b) comprende cuantificar el valor diferencial de rotación seleccionado en la etapa (a3), que comprende:
- (b1) generar datos diferenciales de rotación cuantificando tres valores componentes del valor diferencial de rotación seleccionado;
- 10 (b2) ajustar tres valores componentes cuantificados de los datos diferenciales de rotación;
- (b3) cuantificar de manera inversa valores componentes ajustados;
- (b4) generar un valor diferencial restaurado de rotación restaurando un valor componente, que no ha sido cuantificado, usando los tres valores componentes cuantificados de manera inversa; y
- 15 (b5) medir un error entre el valor diferencial de rotación y el valor diferencial restaurado de rotación y renovar los datos diferenciales de rotación que han de ser codificados por entropía en base al error.
- 64.** El procedimiento de la reivindicación 63 en el que, en la etapa (b2), se ajustan los tres componentes de los datos diferenciales de rotación sumando/restando una variable en un intervalo predeterminado de ajuste a/de cada uno de los valores componentes de los datos diferenciales de rotación generados en la etapa (b1) o a/de cada uno de los valores componentes de los datos diferenciales de rotación renovados en la etapa (b5).
- 20 **65.** El procedimiento de las reivindicaciones 63 o 64 en el que la etapa (b5) comprende:
- (b51) renovar un error ajustado con un error actual entre el valor diferencial de rotación y el valor diferencial restaurado de rotación si el error actual es menor que el error ajustado;
- (b52) llevar a cabo de forma reiterada las etapas (b2) a (b51) hasta que la variable sumada o restada del valor diferencial de rotación esté más allá del intervalo predeterminado de ajuste; y
- 25 (b53) renovar un error final con el error ajustado si el error ajustado es menor que el error final, llevando a cabo las etapas (b2) a (b52) con datos diferenciales de rotación ajustados correspondientes al error ajustado, y determinando datos diferenciales de rotación ajustados correspondientes al error final como datos diferenciales de rotación que han de ser codificados por entropía.
- 66.** El procedimiento de cualquiera de las reivindicaciones 51 a 65 que comprende, además, la generación (bb) de datos diferenciales que tienen un tamaño reducido llevando a cabo una operación de DPCM lineal con los datos diferenciales de rotación y llevando a cabo una operación predeterminada con los datos diferenciales sometidos a DPCM y un valor máximo en un intervalo de cuantificación entre las etapas (b) y (c),
- 30 en el que, en la etapa (c), se codifican por entropía los datos diferenciales cuyo tamaño se reduce en la etapa (bb).
- 67.** El procedimiento de la reivindicación 66 en el que, en la etapa (bb), se da salida a los menores entre los datos diferenciales sometidos a DPCM lineal y las diferencias entre los datos diferenciales sometidos a DPCM lineal y el valor máximo en el intervalo de cuantificación.
- 35 **68.** El procedimiento de las reivindicaciones 66 o 67 en el que la etapa (a) comprende:
- (a1) generar el valor diferencial de rotación usando un valor de transformación rotacional de la trama clave actual y un valor restaurado de transformación rotacional de la trama clave anterior,
- 40 y en el que la etapa (b) comprende:
- (b1) generar datos diferenciales de rotación cuantificando el valor diferencial de rotación;
- (b2) generar un valor diferencial restaurado de rotación cuantificando de manera inversa los datos diferenciales de rotación; y
- 45 (b3) generar un valor restaurado de transformación rotacional de la trama clave actual mediante la multiplicación por cuaterniones del valor diferencial restaurado de rotación por un valor de transformación rotacional de la trama clave anterior.

69. El procedimiento de cualquiera de las reivindicaciones 51 a 68 en el que, en la etapa (c), se codifican datos diferenciales de rotación en una serie de primeras cifras, el número de las cuales se corresponde con la magnitud de los datos diferenciales de rotación, y una segunda cifra de un bit que indica que termina la serie de la primera cifra.
- 5 70. El procedimiento de cualquiera de las reivindicaciones 51 a 69 en el que, en la etapa (c), se codifican datos diferenciales de rotación en cada plano de bits.
71. Un procedimiento de decodificación de una corriente de bits en un procedimiento de animación gráfica basada en tramas clave, obteniéndose la corriente de bits codificando datos de valores clave de un interpolador de orientación que representan la rotación de un objeto en cada trama clave, comprendiendo el procedimiento:
- 10 restaurar datos diferenciales de rotación codificados por entropía cuantificando un valor diferencial de rotación que representa la rotación de un objeto en hasta una diferencia entre transformaciones rotacionales aplicadas al objeto por datos de valores clave del cuaternión de cada trama clave;
- restaurar información de decodificación por entropía que incluye un modo de decodificación por entropía que indica un procedimiento de decodificación por entropía para los datos diferenciales de rotación;
- 15 restaurar información de modulación diferencial de impulsos codificados (DPCM) circular inversa, incluyendo el orden de una operación de DPCM inversa, que indica si debe llevarse a cabo o no la operación de DPCM inversa con los datos diferenciales de rotación decodificados por entropía según el modo de decodificación por entropía; y
- 20 restaurar información de cuantificación, incluyendo el número de bits de cuantificación inversa que se usan para restaurar el valor diferencial de rotación, mediante la cuantificación inversa de los datos diferenciales de rotación con los que se ha llevado a cabo la operación de DPCM circular inversa según el orden de la operación de DPCM inversa.
72. El procedimiento de la reivindicación 71 en el que se codifican los datos diferenciales de rotación de modo que solo se cuantifiquen tres componentes del valor diferencial de rotación representado por un cuaternión.
- 25 73. El procedimiento de la reivindicación 72 en el que la información de decodificación por entropía comprende además:
- una bandera de valores clave que indica si los datos diferenciales de rotación de cada componente de los datos de valores clave tienen o no el mismo valor; y
- 30 datos diferenciales de rotación predeterminados que han de ser decodificados en cada componente de los datos diferenciales de rotación de todos los datos de valores clave cuando la bandera de valores clave indica que los datos diferenciales de rotación de cada uno de los componentes de los datos de valores clave tiene el mismo valor.
74. El procedimiento de la reivindicación 71 en el que la información de DPCM circular inversa comprende datos diferenciales intrarrotacionales usados para llevar a cabo la operación de DPCM circular inversa con los datos diferenciales de rotación.
- 35 75. Una corriente de bits en la que se codifican datos de valores clave de un interpolador de orientación que representan la rotación de un objeto en cada trama clave, comprendiendo la corriente de bits:
- 40 datos diferenciales de rotación codificados por entropía cuantificando un valor diferencial de rotación usados para rotar el objeto en hasta una diferencia entre transformaciones rotacionales aplicadas al objeto por datos de valores clave del cuaternión de cada una de las tramas clave;
- información de decodificación por entropía que comprende un modo de decodificación por entropía, que indica un procedimiento de decodificación por entropía que se ha llevado a cabo con los datos diferenciales de rotación;
- 45 información de DPCM circular inversa que comprende el orden de la operación de DPCM circular inversa, que indica si se llevará a cabo o no una operación de DPCM circular inversa con los datos diferenciales de rotación decodificados por entropía según el modo de decodificación por entropía, comprendiendo la operación de DPCM circular inversa la sustitución de datos sometidos a DPCM que tienen un valor absoluto menor que un valor máximo en un intervalo de cuantificación con datos sometidos a DPCM de valor absoluto mayor y polaridad opuesta y la realización de una operación inversa de DPCM lineal; y
- 50 bits de cuantificación inversa que se usan para restaurar un valor diferencial de rotación cuantificando de manera inversa datos diferenciales de rotación sometidos a DPCM circular inversa según el orden de la operación de DPCM circular inversa.

- 5
76. La corriente de bits de la reivindicación 75 en la que los datos diferenciales de rotación están codificados de modo que solo se cuantifiquen tres componentes de un valor diferencial de rotación representado por un cuaternión.
77. La corriente de bits de las reivindicaciones 75 o 76 en la que la información de decodificación por entropía comprende además:
- 10
- una bandera de valores clave que indica si los datos diferenciales de rotación de cada componente de los datos de valores clave tienen o no el mismo valor; y
- datos diferenciales de rotación predeterminados que han de ser decodificados en cada componente de los datos diferenciales de rotación de todos los datos de valores clave cuando la bandera de valores clave indica que los datos diferenciales de rotación de cada uno de los componentes de los datos de valores clave tiene el mismo valor.
- 15
78. La corriente de bits de las reivindicaciones 75, 76 o 77 en la que la información de DPCM circular inversa comprende datos diferenciales intrarrotacionales usados para llevar a cabo la operación de DPCM circular inversa con los datos diferenciales de rotación.
79. Un programa de ordenador que comprende un medio de código para llevar a cabo todas las etapas de cualquiera de las reivindicaciones 39 a 74 cuando dicho programa se ejecuta en un ordenador.
80. Un programa de ordenador según se reivindica en la reivindicación 79 implementado en un medio legible por ordenador.

FIG. 1

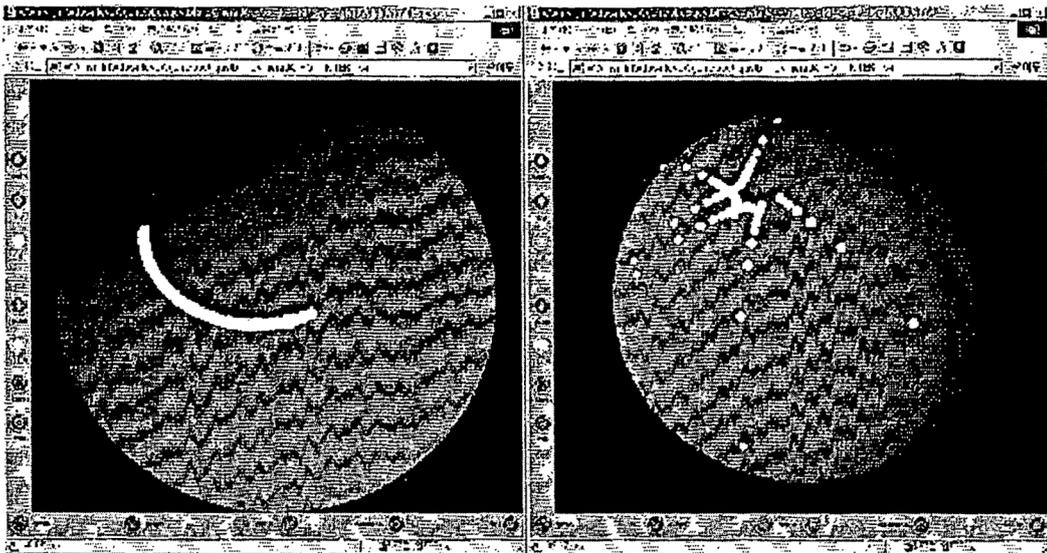


FIG. 2A (TÉCNICA ANTERIOR)

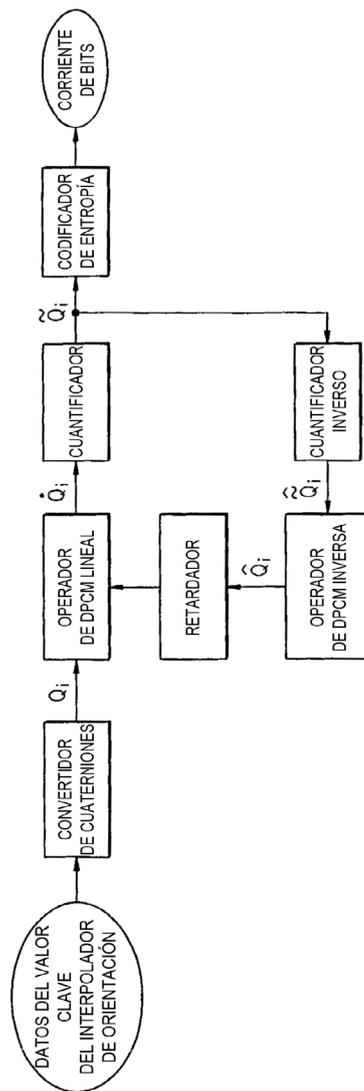


FIG. 2B (TÉCNICA ANTERIOR)

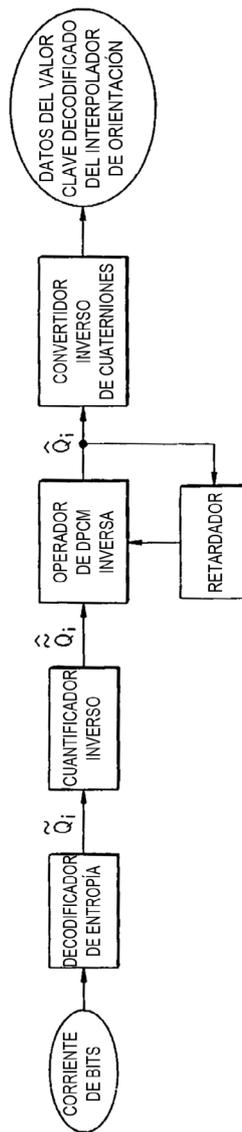


FIG. 3A

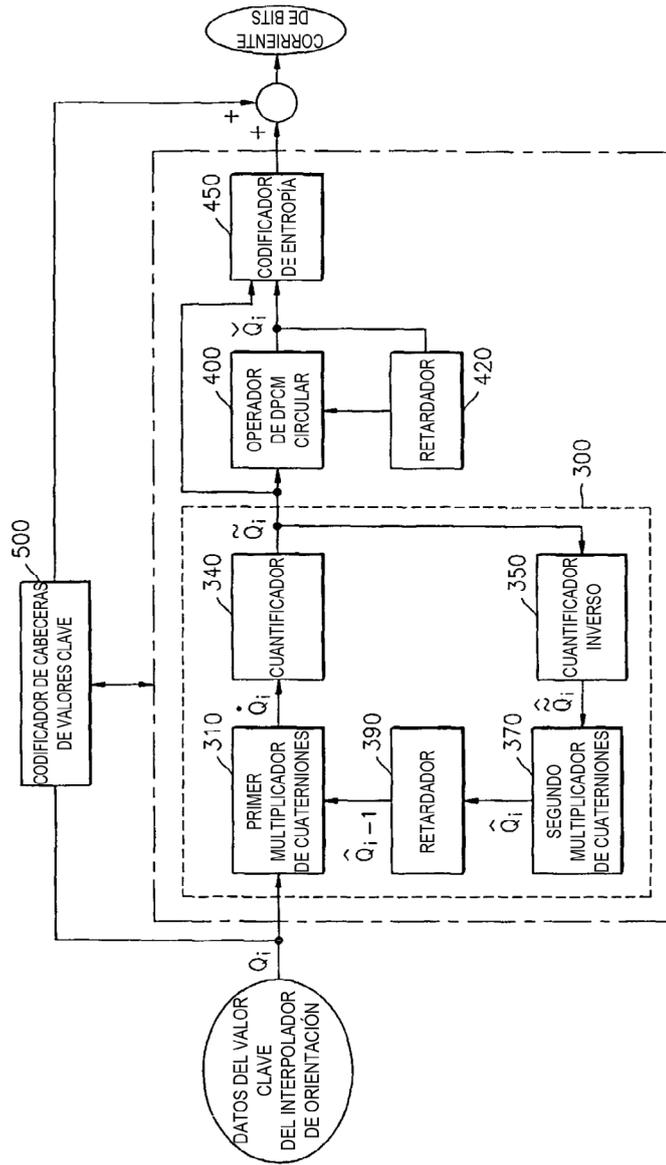


FIG. 3B

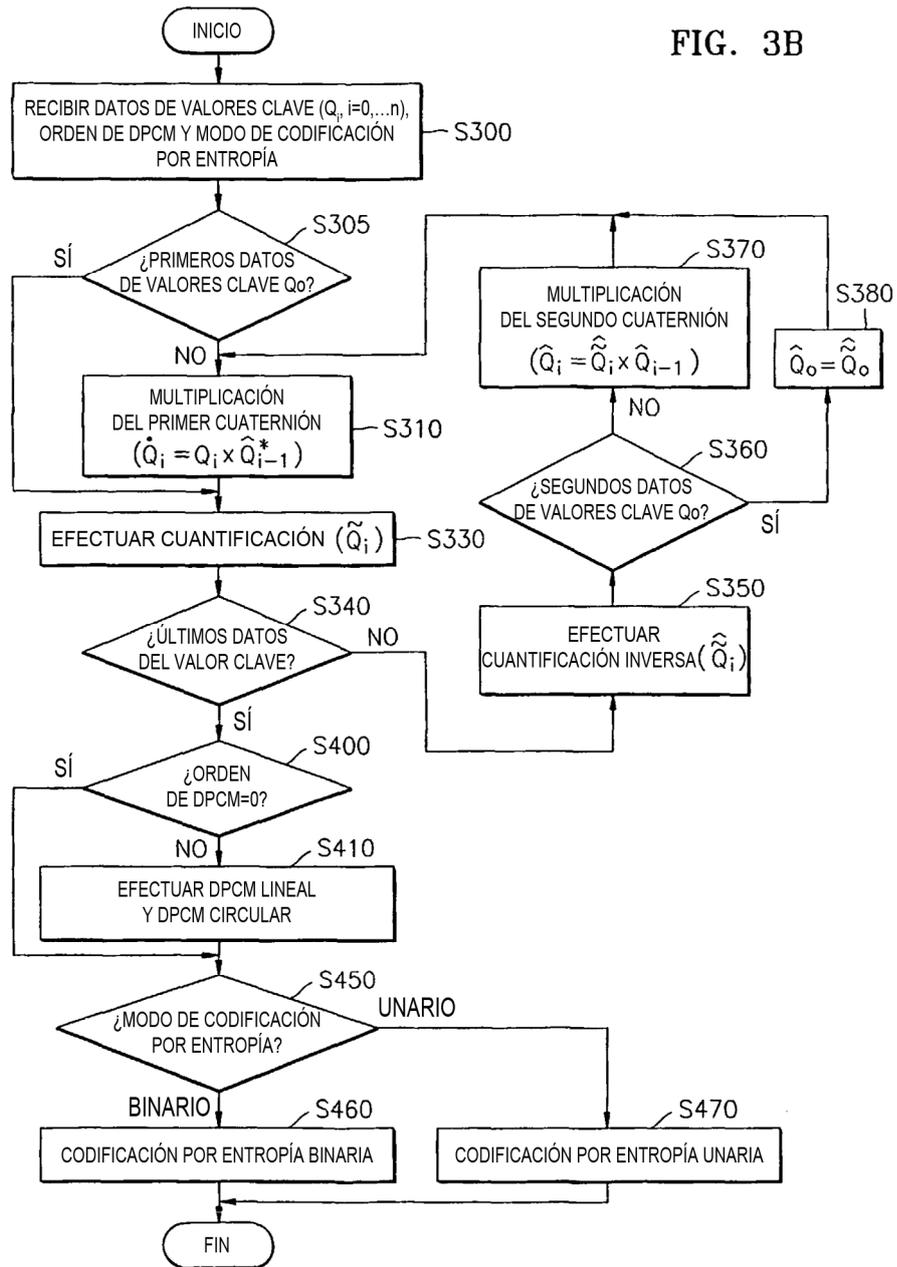


FIG. 4A

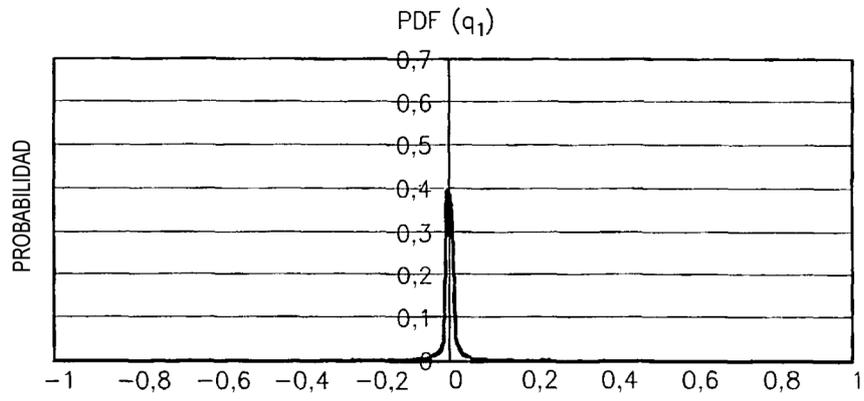


FIG. 4B

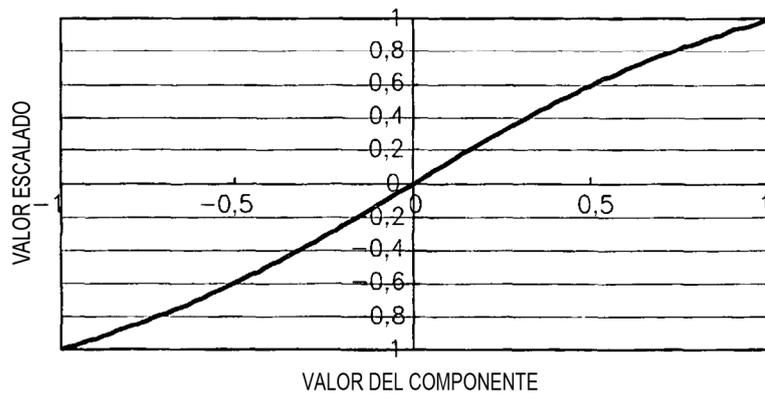


FIG. 5A

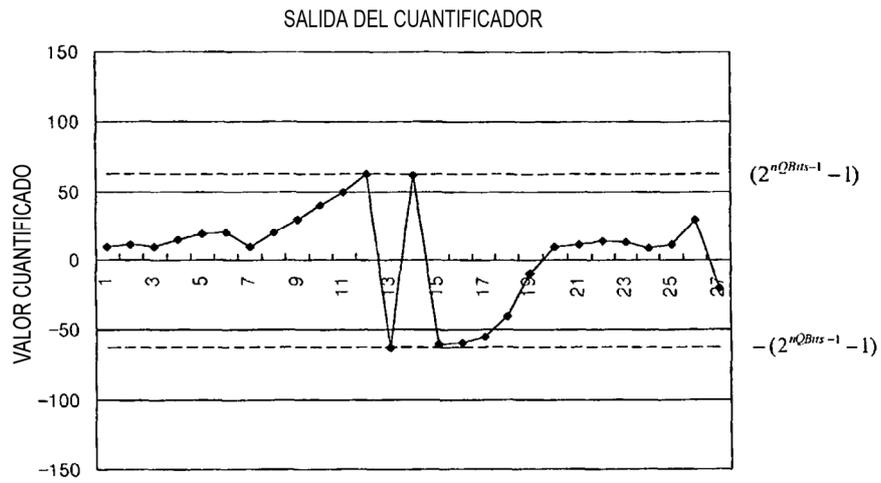


FIG. 5B

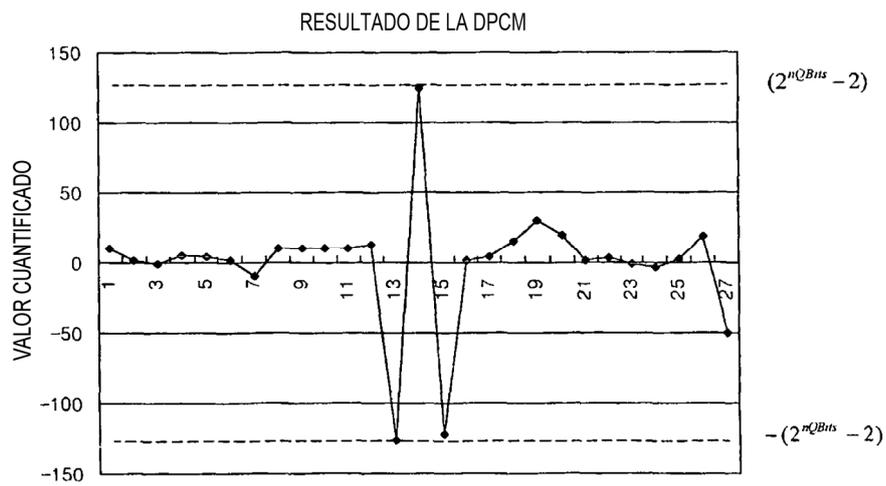


FIG. 5C

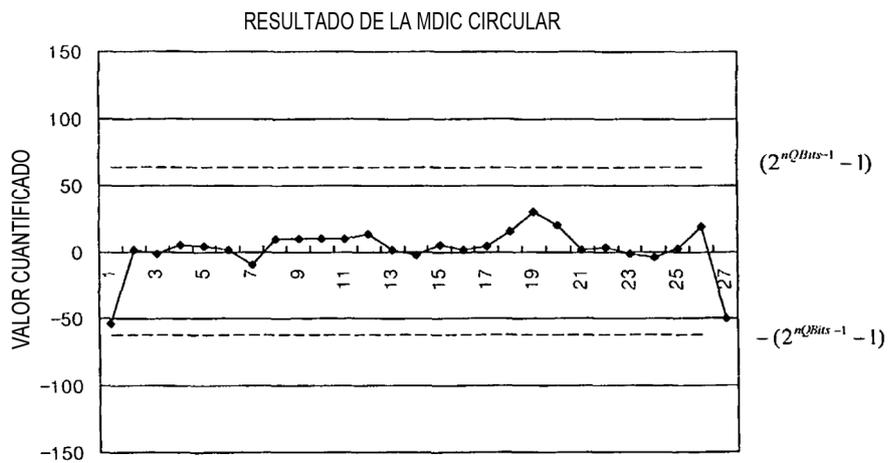


FIG. 6A

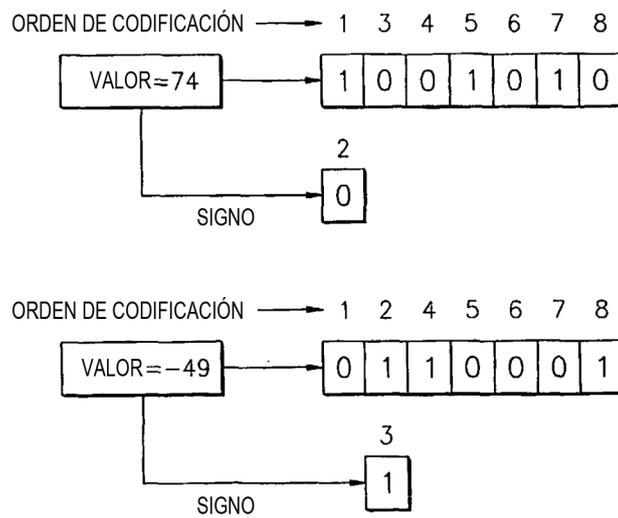


FIG. 6B

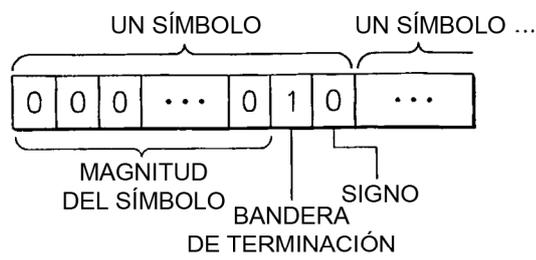


FIG. 7

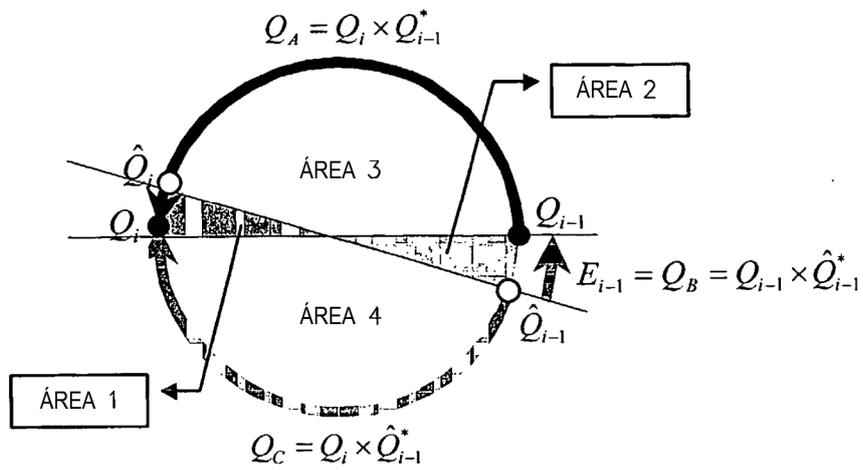




FIG. 8B

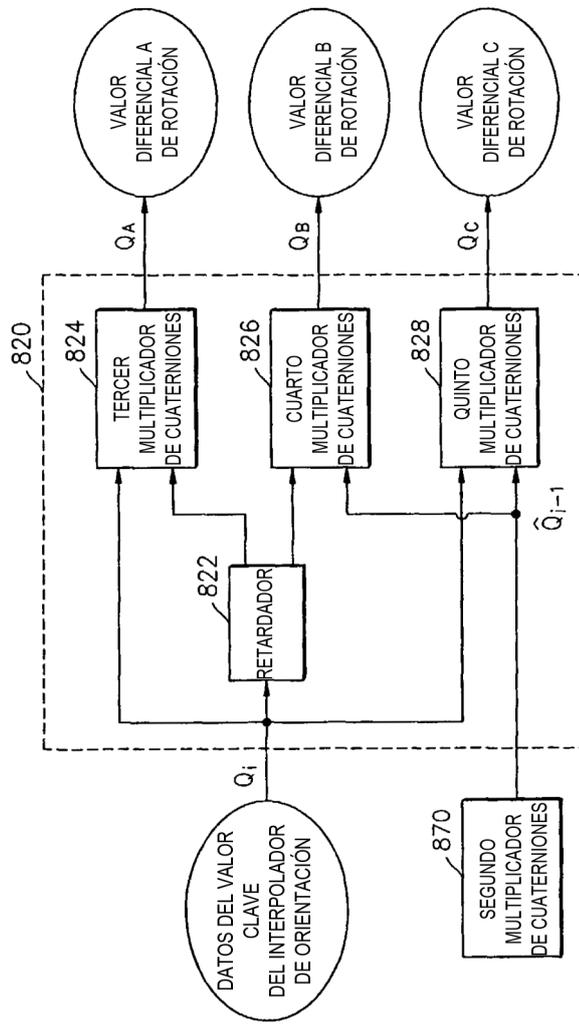


FIG. 9A

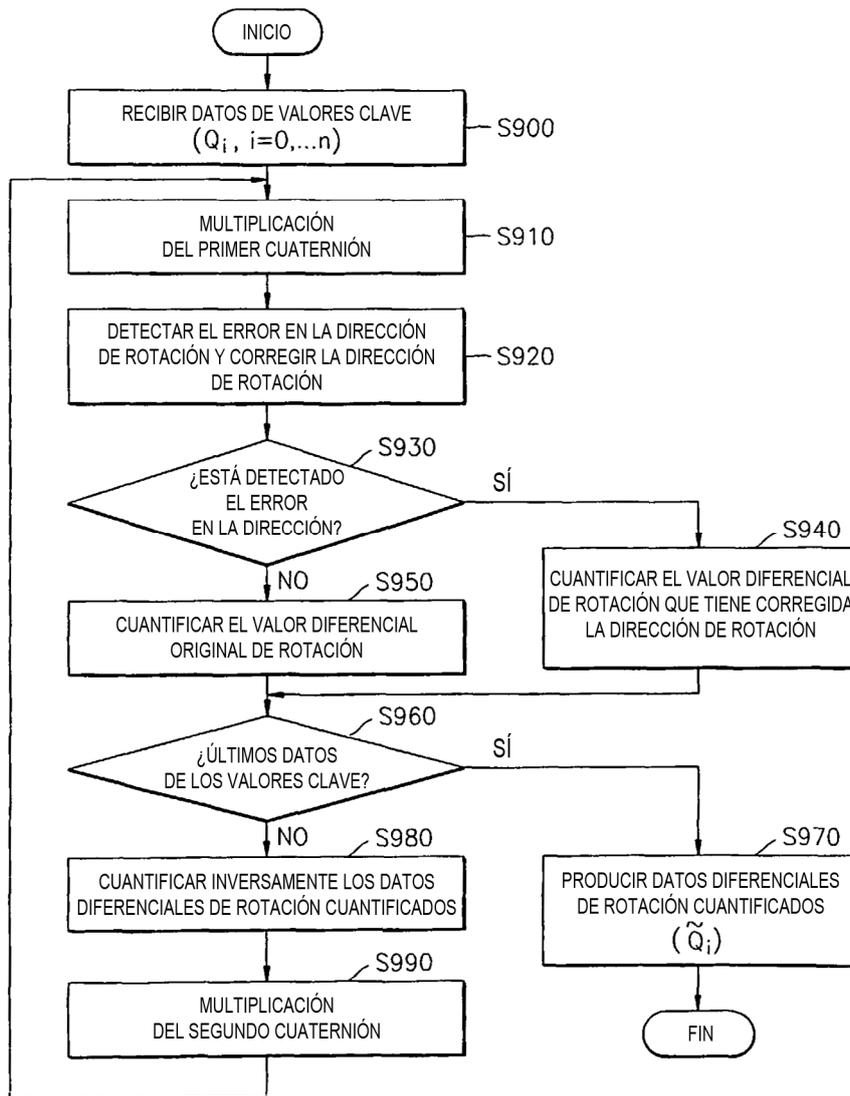


FIG. 9B

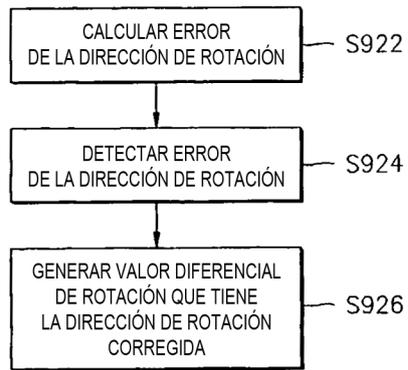


FIG. 10A

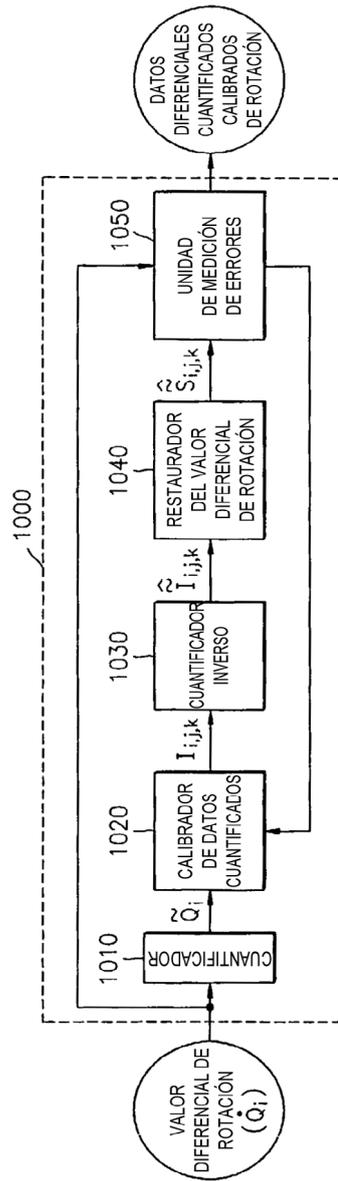


FIG. 10B

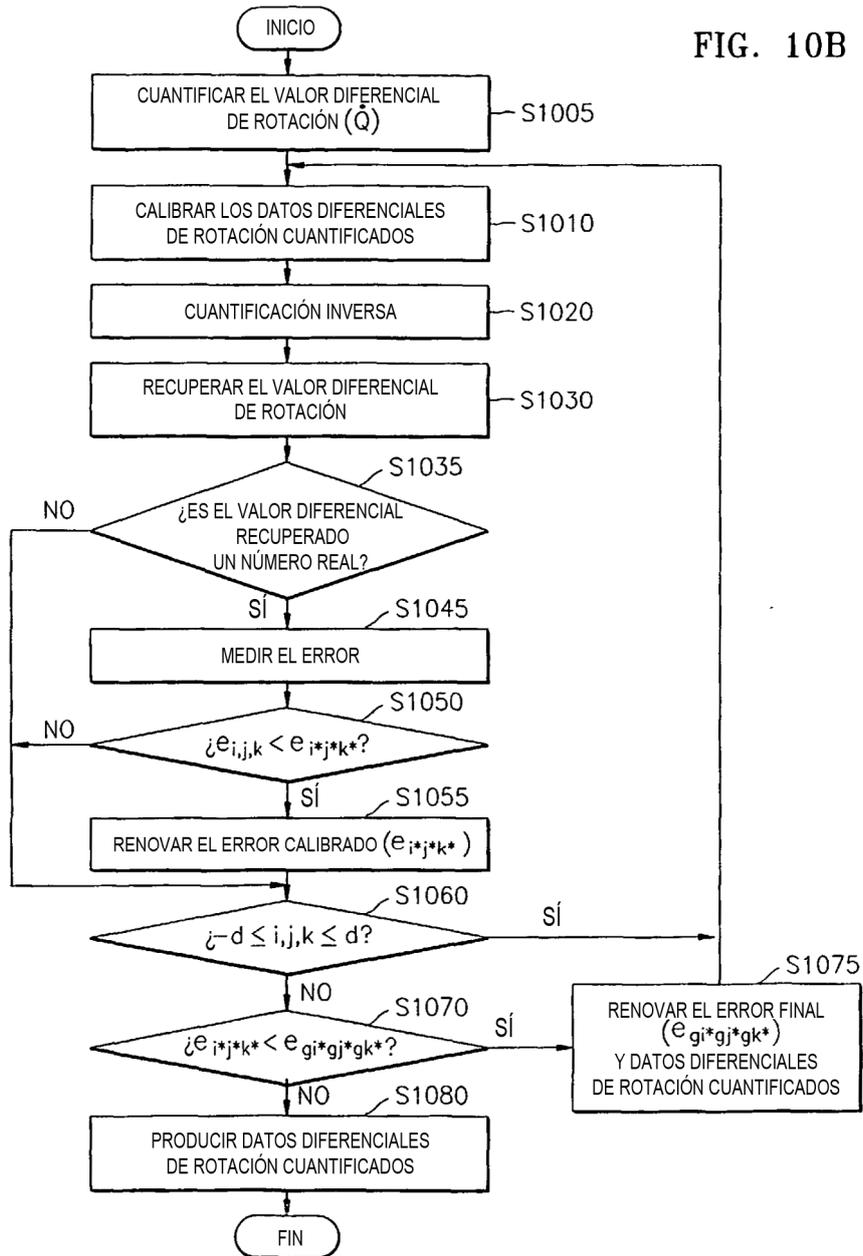


FIG. 11A

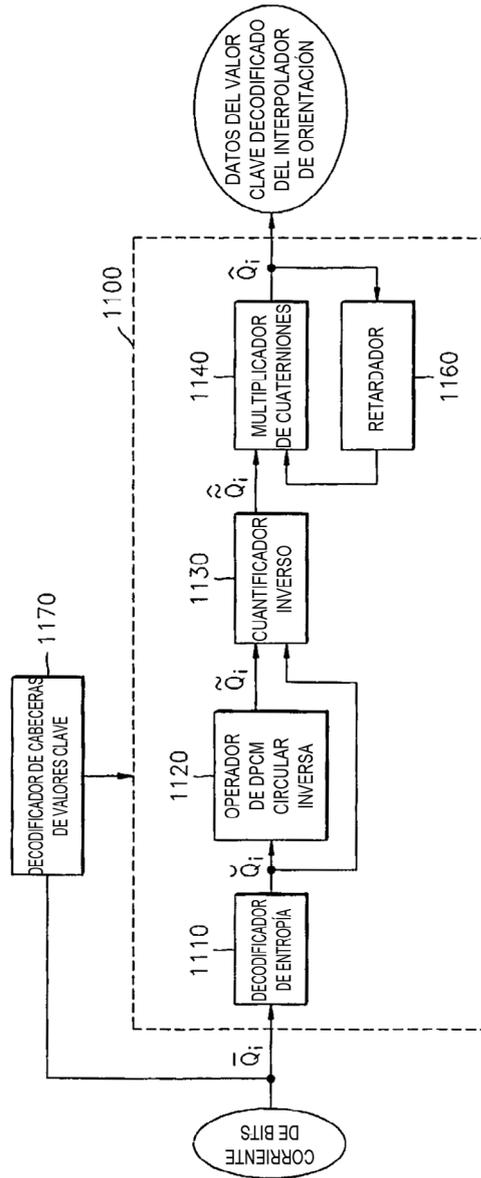


FIG. 11B

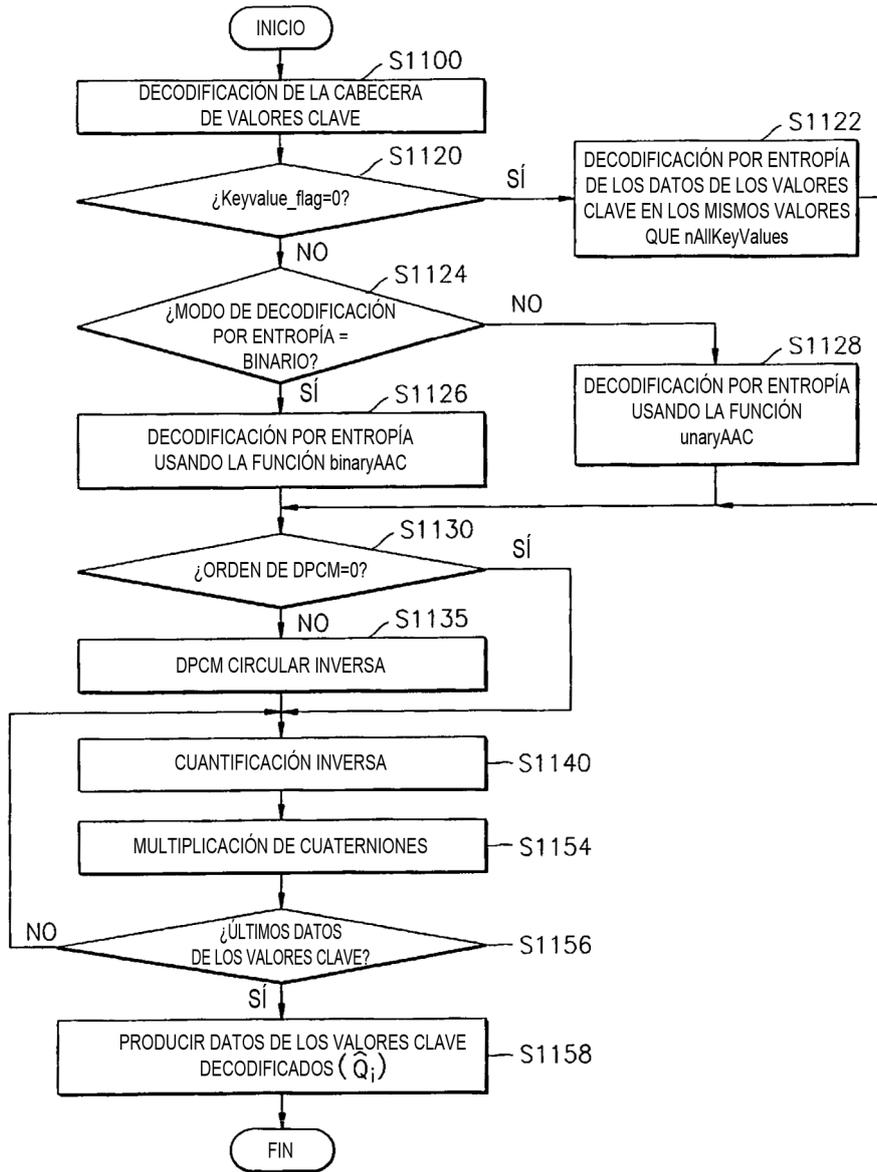


FIG. 12

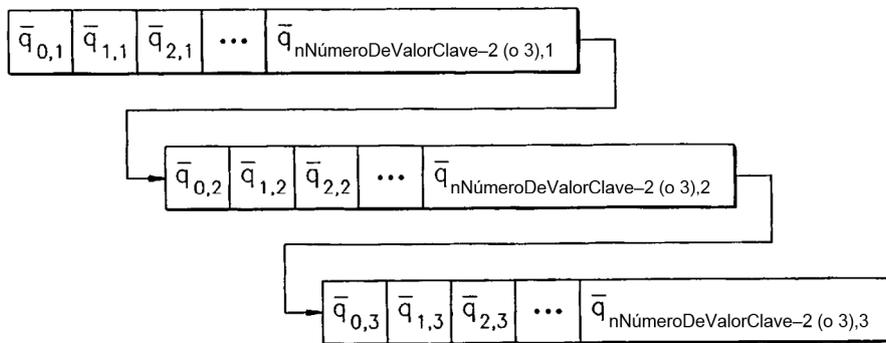


FIG. 13A

```

class CompressedOrientationInterpolator {
    OriKeyValueHeader oriKVHeader;
    qf_start();
    if(oriKVHeader.nKVDPCMOrder == 0) //DPCM de 1er orden
        OriDPCMKeyValue oriDPCMKeyValue(oriKVHeader.oriDPCMKVHeader.oriKVHeader.nNumberOfKeyValue-1);
    else //DPCM de 2º orden
        OriDPCMKeyValue oriDPCMKeyValue(oriKVHeader.oriDPCMKVHeader.oriKVHeader.nNumberOfKeyValue-2);
}
    
```

FIG. 13B

```
class OrilKeyValueHeader () {  
    unsigned int(5) nKVQBit;  
    unsigned int(5) nNumKeyValueCodingBit;  
    unsigned int(nNumKeyValueCodingBit) nNumberOfKeyValue;  
    bit(1) nKVDPCMOrder;  
    OrilDPCMKeyValueHeader orilDPCMKVHeader(nKVQBit, nKVDPCMOrder);  
}
```

FIG. 13C

```

class OrilDPCMKeyValueHeader (int nKVQBit, unsigned int nKVDPCMOrder) {
    unsigned int(nKVQBit-1) firstQKV_S;
    bit(1) nFirstXSign;
    unsigned int(nKVQBit-1) firstQKV_X;
    bit(1) nFirstYSign;
    unsigned int(nKVQBit-1) firstQKV_Y;
    bit(1) nFirstZSign;
    unsigned int(nKVQBit-1) firstQKV_Z;
    if (nKYDPCMOrder == 1) { //DPCM de 2º orden
        bit(1) nSecondXSign;
        unsigned int(nKVQBit-1) secondQKV_X;
        bit(1) nSecondYSign;
        unsigned int(nKVQBit-1) secondQKV_Y;
        bit(1) nSecondZSign;
        unsigned int(nKVQBit-1) secondQKV_Z;
        bit(1) bIsMoreTwoKVs;
    }
    if (nKVDPCMOrder == 0 || bIsMoreTwoKVs == 1) {
        bit(1) x_keyvalue_flag;
        OrilKeyValueCodingBit orilKVCodingBit_X(x_keyvalue_flag, nKVQBit);
        bit(1) y_keyvalue_flag;
        OrilKeyValueCodingBit orilKVCodingBit_Y(y_keyvalue_flag, nKVQBit);
        bit(1) z_keyvalue_flag;
        OrilKeyValueCodingBit orilKVCodingBit_Z(z_keyvalue_flag, nKVQBit);
    }
}

```

FIG. 13D

```
class OriKeyValueCodingBit (unsigned int flag_bit, int nKVQBit) {
    int count = (int)(log10(nKVQBit)/log10(2)) + 1;
    if(flag_bit == 0) {
        unsigned int(count) nKVCodingBit;
        if(nKVCodingBit == 1)
            unsigned int(nKVCodingBit) nAllKeyValue;
        else {
            bit(1) nSign;
            unsigned int(nKVCodingBit-1) nAllKeyValue;
        } else {
            bit(1) bIsUnaryAAC;
            if(bIsUnaryAAC != 1)
                unsigned int(count) nKVCodingBit;
        }
    }
}
```

FIG. 13E

```

class OrilDPCMKeyValue(OrilDPCMKeyValueHeader kvHeader, int nNumKV) {
    int i;
    if(kvHeader.x_keyvalue_flag != 0) {
        if(kvHeader.orilKVCodingBit_X.bIsUnaryAAC == 1)
            for(i = 0; i < nNumKV; i++)
                decodeUnaryAAC(&DeltaKeyValue[i].x, kvXSignContext, kvXUnaryContext);
        else
            for(i = 0; i < nNumKV; i++)
                decodeSignedAAC(&DeltaKeyValue[i].x, kvHeader.orilKVCodingBit_X.nKVCodingBit, kvXSignContext,
                    kvXContext);
    }
    if(kvHeader.y_keyvalue_flag != 0) {
        if(kvHeader.orilKVCodingBit_Y.bIsUnaryAAC == 1)
            for(i = 0; i < nNumKV; i++)
                decodeUnaryAAC(&DeltaKeyValue[i].y, kvYSignContext, kvYUnaryContext);
        else
            for(i = 0; i < nNumKV; i++)
                decodeSignedAAC(&DeltaKeyValue[i].y, kvHeader.orilKVCodingBit_Y.nKVCodingBit, kvYSignContext,
                    kvYContext);
    }
    if(kvHeader.z_keyvalue_flag != 0) {
        if(kvHeader.orilKVCodingBit_Z.bIsUnaryAAC == 1)
            for(i = 0; i < nNumKV; i++)
                decodeUnaryAAC(&DeltaKeyValue[i].z, kvZSignContext, kvZUnaryContext);
        else
            for(i = 0; i < nNumKV; i++)
                decodeSignedAAC(&DeltaKeyValue[i].z, kvHeader.orilKVCodingBit_Z.nKVCodingBit, kvZSignContext,
                    kvZContext);
    }
}
}

```

FIG. 14

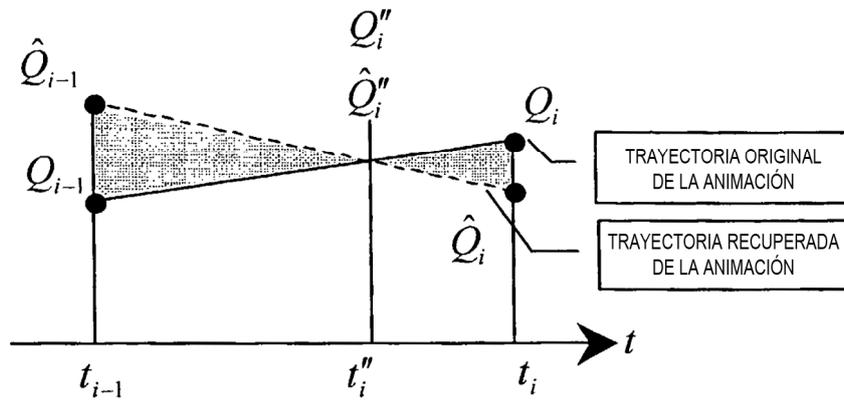


FIG. 15

