

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 994 746**

51 Int. Cl.:

G06T 9/00 (2006.01)
G06T 7/50 (2007.01)
H04N 19/42 (2014.01)
H04N 19/70 (2014.01)
H04N 19/593 (2014.01)
H04N 19/105 (2014.01)
G06T 9/40 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **05.01.2021** **PCT/KR2021/000071**
87 Fecha y número de publicación internacional: **15.07.2021** **WO21141352**
96 Fecha de presentación y número de la solicitud europea: **05.01.2021** **E 21738173 (0)**
97 Fecha y número de publicación de la concesión europea: **23.10.2024** **EP 4068213**

54 Título: **Dispositivo de transmisión de datos de la nube de puntos, método de transmisión de datos de la nube de puntos, dispositivo de recepción de datos de la nube de puntos y método de recepción de datos de la nube de puntos**

30 Prioridad:

07.01.2020 US 202062958264 P
15.01.2020 KR 20200005544

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
30.01.2025

73 Titular/es:

LG ELECTRONICS INC. (100.00%)
128, Yeoui-daero Yeongdeungpo-gu
Seoul 07336, KR

72 Inventor/es:

HUR, HYEJUNG y
OH, SEJIN

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 994 746 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

Dispositivo de transmisión de datos de la nube de puntos, método de transmisión de datos de la nube de puntos, dispositivo de recepción de datos de la nube de puntos y método de recepción de datos de la nube de puntos

Campo técnico

5 Las realizaciones se refieren a un método y aparato para procesar contenido de la nube de puntos.

Técnica anterior

10 El contenido de la nube de puntos es contenido representado por una nube de puntos, que es un conjunto de puntos que pertenecen a un sistema de coordenadas que representa un espacio tridimensional. El contenido de la nube de puntos puede expresar medios configurados en tres dimensiones, y se utiliza para proporcionar diversos servicios tales como realidad virtual (VR), realidad aumentada (AR), realidad mixta (MR), XR (realidad extendida) y servicios autónomos. Sin embargo, se requieren decenas de miles a cientos de miles de datos de puntos para representar el contenido de la nube de puntos. Por lo tanto, existe la necesidad de un método para procesar de manera eficiente una gran cantidad de datos de puntos.

15 La técnica anterior se encuentra en Khaled Mammou y col., "G-PCC codec description v2" (descripción de códec G-PCC v2), ISO/IEC JTC1/SC29/WG11 N18189.

Descripción

Problema técnico

20 Un objeto de la presente descripción diseñada para resolver los problemas descritos anteriormente es proporcionar un dispositivo de transmisión de datos de la nube de puntos, un método de transmisión de datos de la nube de puntos, un dispositivo de recepción de datos de la nube de puntos y un método de recepción de datos de la nube de puntos para transmitir y recibir eficientemente una nube de puntos

25 Otro objeto de la presente descripción es proporcionar un dispositivo de transmisión de datos de la nube de puntos, un método de transmisión de datos de la nube de puntos, un dispositivo de recepción de datos de la nube de puntos y un método de recepción de datos de la nube de puntos para abordar la latencia y la complejidad de codificación/decodificación

30 Otro objeto de la presente descripción es proporcionar un dispositivo de transmisión de datos de la nube de puntos, un método de transmisión de datos de la nube de puntos, un dispositivo de recepción de datos de la nube de puntos y un método de recepción de datos de la nube de puntos para mejorar el rendimiento de compresión de la nube de puntos mejorando la técnica de codificación de información de atributo de compresión de la nube de puntos basada en geometría (G-PCC).

Otro objeto de la presente descripción es proporcionar un dispositivo de transmisión de datos de la nube de puntos, un método de transmisión de datos de la nube de puntos, un dispositivo de recepción de datos de la nube de puntos y un método de recepción de datos de la nube de puntos para mejorar la eficiencia de compresión mientras soporta el tratamiento paralelo de información de atributo de G-PCC.

35 Otro objeto de la presente descripción es proporcionar un aparato de transmisión de datos de la nube de puntos, un método de transmisión de datos de la nube de puntos, un aparato de recepción de datos de la nube de puntos y un método de recepción de datos de la nube de puntos para reducir el tamaño de un flujo de bits de atributos y mejorar la eficiencia de compresión de atributos seleccionando puntos contiguos utilizados para la predicción de atributos en consideración de una correlación de atributos entre puntos de contenido cuando se codifica información de atributo de G-PCC.

40 Otro objeto de la presente descripción es proporcionar un aparato de transmisión de datos de la nube de puntos, un método de transmisión de datos de la nube de puntos, un aparato de recepción de datos de la nube de puntos y un método de recepción de datos de la nube de puntos para reducir el tamaño de un flujo de bits de atributos y mejorar la eficiencia de compresión de atributos aplicando una distancia máxima entre puntos contiguos y seleccionando puntos contiguos utilizados para la predicción de atributos cuando se codifica la información de atributo de G-PCC.

45 Los objetos de la presente descripción no se limitan a los objetos mencionados anteriormente, y otros objetos de la presente descripción que no se mencionan anteriormente resultarán evidentes para los expertos en la técnica tras el examen de la siguiente descripción.

50 Solución técnica

- Para lograr estos objetos y otras ventajas y según el propósito de la descripción, según un primer aspecto, se describe un método de transmisión de datos de la nube de puntos, comprendiendo el método: adquirir datos de la nube de puntos; codificar información de geometría que incluye posiciones de puntos de los datos de la nube de puntos; generar uno o más niveles de detalle, LOD, basándose en un árbol octal y seleccionar uno o más puntos contiguos de cada punto a codificar por atributo basándose en el uno o más LOD, en donde cada nivel de profundidad del árbol octal coincide con cada LOD, en donde un LOD superior incluye puntos pertenecientes a un LOD inferior, en donde el uno o más puntos contiguos seleccionados de cada punto están ubicados dentro de una distancia máxima entre puntos contiguos, en donde la distancia máxima entre puntos contiguos se obtiene multiplicando una distancia base por un intervalo máximo de puntos contiguos, en donde la distancia base es una distancia diagonal de un nodo en cada LOD, en donde el intervalo máximo de puntos contiguos es un número de nodos alrededor de cada punto, y en donde el uno o más puntos contiguos seleccionados de cada punto se utiliza para generar información de atributo predicha de cada punto que codifica información de atributo de cada punto basándose en la información de atributo predicha de cada punto; y transmitir la información de geometría codificada, la información de atributo codificada e información de señalización.
- El uno o más puntos contiguos seleccionados de cada punto están ubicados dentro de una distancia máxima entre puntos contiguos.
- La distancia máxima entre puntos contiguos se determina basándose en una distancia base entre puntos contiguos y un intervalo máximo de puntos contiguos.
- Cuando el uno o más LOD se generan basándose en un árbol octal, la distancia base entre puntos contiguos se determina basándose en una distancia diagonal de un nodo en un LOD específico.
- El intervalo máximo de puntos contiguos es el número de nodos contiguos de cada punto y, según las realizaciones, puede señalizarse en la información de señalización.
- Según un segundo aspecto, describimos un aparato de transmisión de datos de la nube de puntos, comprendiendo el aparato: una unidad de adquisición configurada para adquirir datos de la nube de puntos; un codificador de geometría configurado para codificar información de geometría que incluye posiciones de puntos de los datos de la nube de puntos; un codificador de atributo configurado para generar uno o más niveles de detalle, LOD, basándose en un árbol octal, seleccionar uno o más puntos contiguos de cada punto a codificar por atributo basándose en el uno o más LOD, en donde cada nivel de profundidad del árbol octal coincide con cada LOD, en donde un LOD superior incluye puntos pertenecientes a un LOD inferior, en donde el uno o más puntos contiguos seleccionados de cada punto están ubicados dentro de una distancia máxima entre puntos contiguos, en donde la distancia máxima entre puntos contiguos se obtiene multiplicando una distancia base por un intervalo máximo de puntos contiguos, en donde la distancia base es una distancia diagonal de un nodo en cada LOD, en donde el intervalo máximo de puntos contiguos es un número de nodos alrededor de cada punto, y en donde el uno o más puntos contiguos seleccionados de cada punto se utilizan para generar información de atributo predicha de cada punto, y codificar información de atributo de cada punto basándose en la información de atributo predicha de cada punto; y un transmisor configurado para transmitir la información de geometría codificada, la información de atributo codificada e información de señalización.
- El uno o más puntos contiguos seleccionados de cada punto están ubicados dentro de una distancia máxima entre puntos contiguos.
- La distancia máxima entre puntos contiguos se determina basándose en una distancia base entre puntos contiguos y un intervalo máximo de puntos contiguos.
- Cuando el uno o más LOD se generan basándose en un árbol octal, la distancia base entre puntos contiguos se determina basándose en una distancia diagonal de un nodo en un LOD específico.
- El intervalo máximo de puntos contiguos es el número de nodos contiguos de cada punto y, según las realizaciones, puede señalizarse en la información de señalización.
- Según un tercer aspecto, describimos un aparato de recepción de datos de la nube de puntos, comprendiendo el aparato: un receptor configurado para recibir información de geometría, información de atributo e información de señalización; un decodificador de geometría configurado para decodificar la información de geometría basándose en la información de señalización; un decodificador de atributo configurado para generar uno o más niveles de detalle, LOD, basándose en un árbol octal, seleccionar uno o más puntos contiguos de cada punto a decodificar por atributo basándose en el uno o más LOD, en donde cada nivel de profundidad del árbol octal coincide con cada LOD, en donde un LOD superior incluye puntos pertenecientes a un LOD inferior, en donde el uno o más puntos contiguos seleccionados de cada punto están ubicados dentro de una distancia máxima entre puntos contiguos, en donde la distancia máxima entre puntos contiguos se obtiene multiplicando una distancia base por un intervalo máximo de puntos contiguos, en donde la distancia base es una distancia diagonal de un nodo en cada LOD, en donde el intervalo máximo de puntos contiguos es un número de nodos alrededor de cada punto, y en donde el uno o más puntos contiguos seleccionados de cada punto se utilizan para generar información de

atributo predicha de cada punto, y decodificar la información de atributo de cada punto basándose en la información de atributo predicha de cada punto y la información de señalización; y un renderizador configurado para renderizar datos de la nube de puntos restaurados basándose en la información de geometría decodificada y la información de atributo decodificado.

- 5 El uno o más puntos contiguos seleccionados de cada punto están ubicados dentro de una distancia máxima entre puntos contiguos.

La distancia máxima entre puntos contiguos se determina basándose en una distancia base entre puntos contiguos y un intervalo máximo de puntos contiguos.

- 10 Cuando el uno o más LOD se generan basándose en un árbol octal, la distancia base entre puntos contiguos se determina basándose en una distancia diagonal de un nodo en un LOD específico.

El intervalo máximo de puntos contiguos es el número de nodos contiguos de cada punto y, según las realizaciones, puede señalizarse en la información de señalización.

- 15 Según un cuarto aspecto, describimos un método de recepción de datos de la nube de puntos, comprendiendo el método: la recepción de información de geometría, información de atributo e información de señalización; la decodificación de la información de geometría basándose en la información de señalización; la generación de uno o más niveles de detalle, LOD, basándose en un árbol octal; la selección de uno o más puntos contiguos de cada punto que se ha de decodificar por atributo basándose en el uno o más LOD, en donde cada nivel de profundidad del árbol octal coincide con cada LOD, en donde un LOD superior incluye puntos pertenecientes a un LOD inferior, en donde el uno o más puntos contiguos seleccionados de cada punto están ubicados dentro de una distancia máxima entre puntos contiguos, en donde la distancia máxima entre puntos contiguos se obtiene multiplicando una distancia base por un intervalo máximo de puntos contiguos, en donde la distancia base es una distancia diagonal de un nodo en cada LOD, en donde el intervalo máximo de puntos contiguos es un número de nodos alrededor de cada punto, y en donde el uno o más puntos contiguos seleccionados de cada punto se utiliza para generar información de atributo predicha de cada punto; la decodificación de la información de atributo de cada punto basándose en la información de atributo predicha de cada punto y la información de señalización; y la renderización de datos de la nube de puntos restaurados basándose en la información de geometría decodificada y la información de atributo decodificado.
- 20
- 25

- 30 La invención se define por las realizaciones en donde los LOD se generan basándose en un árbol octal, tal como se define por las características de las reivindicaciones. Otros ejemplos basados en diferentes métodos de generación de LOD no son parte de la invención y se proporcionan con fines ilustrativos solo para soportar una mejor comprensión de la invención.

Efectos ventajosos

- 35 Un método de transmisión de datos de la nube de puntos, un dispositivo de transmisión de datos de la nube de puntos, un método de recepción de datos de la nube de puntos y un dispositivo de recepción de datos de la nube de puntos según las realizaciones pueden proporcionar un servicio de la nube de puntos de buena calidad.

Un método de transmisión de datos de la nube de puntos, un dispositivo de transmisión de datos de la nube de puntos, un método de recepción de datos de la nube de puntos y un dispositivo de recepción de datos de la nube de puntos según las realizaciones pueden lograr diversos métodos de códec de vídeo.

- 40 Un método de transmisión de datos de la nube de puntos, un dispositivo de transmisión de datos de la nube de puntos, un método de recepción de datos de la nube de puntos y un dispositivo de recepción de datos de la nube de puntos según las realizaciones pueden proporcionar contenido universal de la nube de puntos tal como un servicio de autónomo (o un servicio de conducción autónoma).

- 45 Un método de transmisión de datos de la nube de puntos, un dispositivo de transmisión de datos de la nube de puntos, un método de recepción de datos de la nube de puntos y un dispositivo de recepción de datos de la nube de puntos según las realizaciones pueden realizar una partición adaptativa al espacio de datos de la nube de puntos para la codificación y decodificación independientes de los datos de la nube de puntos, mejorando así el tratamiento paralelo y proporcionando escalabilidad.

- 50 Un método de transmisión de datos de la nube de puntos, un dispositivo de transmisión de datos de la nube de puntos, un método de recepción de datos de la nube de puntos y un dispositivo de recepción de datos de la nube de puntos según las realizaciones pueden realizar codificación y decodificación dividiendo en particiones espacialmente los datos de la nube de puntos en unidades de mosaicos y/o segmentos, y señalar los datos necesarios para ello, mejorando así el rendimiento de codificación y decodificación de la nube de puntos.

- 55 Un método y aparato de transmisión de datos de la nube de puntos, y un método y aparato de recepción de datos de la nube de puntos según las realizaciones reducen el tamaño de un flujo de bits de atributos y mejoran la eficiencia de compresión de atributos seleccionando puntos contiguos utilizados para la predicción de atributos

en consideración de una correlación de atributos entre puntos de contenido cuando se codifica información de atributo de G-PCC.

- 5 Un método y aparato de transmisión de datos de la nube de puntos, y un método y aparato de recepción de datos de la nube de puntos según las realizaciones reducen el tamaño de un flujo de bits de atributos y mejoran la eficiencia de compresión de atributos aplicando una distancia máxima entre puntos contiguos para seleccionar puntos contiguos utilizados para la predicción de atributos cuando se codifica la información de atributo de G-PCC.

Descripción de los dibujos

- 10 Los dibujos adjuntos, que se incluyen para proporcionar una comprensión adicional de la descripción y se incorporan en y constituyen una parte de esta solicitud, ilustran realización(es) de la descripción y junto con la descripción sirven para explicar el principio de la descripción. En los dibujos:

La fig. 1 ilustra un sistema de suministro de contenido de la nube de puntos ejemplar según las realizaciones.

La fig. 2 es un diagrama de bloques que ilustra una operación de suministro de contenido de la nube de puntos según las realizaciones.

- 15 La fig. 3 ilustra un proceso ejemplar de captura de un vídeo de la nube de puntos según las realizaciones.

La fig. 4 ilustra un diagrama de bloques ejemplar de codificador de vídeo de la nube de puntos según las realizaciones.

La fig. 5 ilustra un ejemplo de vóxeles en un espacio 3D según las realizaciones.

La fig. 6 ilustra un ejemplo de árbol octal y código de ocupación según las realizaciones.

- 20 La fig. 7 ilustra un ejemplo de un patrón de nodo contiguo según las realizaciones.

La fig. 8 ilustra un ejemplo de configuración de puntos de un contenido de la nube de puntos para cada LOD según las realizaciones.

La fig. 9 ilustra un ejemplo de configuración de puntos de un contenido de la nube de puntos para cada LOD según las realizaciones.

- 25 La fig. 10 ilustra un ejemplo de un diagrama de bloques de un decodificador de vídeo de la nube de puntos según las realizaciones.

La fig. 11 ilustra un ejemplo de un decodificador de vídeo de la nube de puntos según las realizaciones.

La fig. 12 ilustra una configuración para la codificación de vídeo de la nube de puntos de un dispositivo de transmisión según las realizaciones.

- 30 La fig. 13 ilustra una configuración para la decodificación de vídeo de la nube de puntos de un dispositivo de recepción según las realizaciones.

La fig. 14 ilustra una estructura ejemplar conectable operativamente con un método/dispositivo para transmitir y recibir datos de la nube de puntos según las realizaciones.

La fig. 15 ilustra un ejemplo de un dispositivo de transmisión de la nube de puntos según las realizaciones.

- 35 (a) de la fig. 16 a (c) de la fig. 16 ilustran una realización de división en particiones de una zona de delimitación en uno o más mosaicos.

La fig. 17 ilustra un ejemplo de un codificador de geometría y un codificador de atributo según las realizaciones.

La fig. 18 es un diagrama que ilustra un ejemplo de generación de LOD basándose en un árbol octal según las realizaciones.

- 40 La fig. 19 es un diagrama que ilustra un ejemplo de disposición de puntos de una nube de puntos en un orden de código de Morton según las realizaciones.

La fig. 20 es un diagrama que ilustra un ejemplo de búsqueda de puntos contiguos basándose en LOD según las realizaciones.

(a) de la fig. 21 y (b) de la fig. 21 ilustran ejemplos de contenidos de la nube de puntos según las realizaciones.

- (a) de la fig. 22 y (b) de la fig. 22 ilustran ejemplos de una distancia promedio, una distancia mínima y una distancia máxima de cada punto que pertenece al conjunto de puntos contiguos según las realizaciones.
- (a) de la fig. 23 ilustra un ejemplo de adquisición de una distancia diagonal de un nodo de árbol octal de LOD0 según las realizaciones.
- 5 (b) de la fig. 23 ilustra un ejemplo de adquisición de una distancia diagonal de un nodo de árbol octal de LOD1 según las realizaciones.
- (a) de la fig. 24 y (b) de la fig. 24 ilustran ejemplos de un intervalo que puede seleccionarse como un punto contiguo en cada LOD.
- 10 La fig. 25 ilustra ejemplos de una distancia base entre puntos contiguos que pertenece a cada LOD según las realizaciones.
- La fig. 26 ilustra otro ejemplo de búsqueda de puntos contiguos basándose en LOD según las realizaciones.
- La fig. 27 ilustra otro ejemplo de búsqueda de puntos contiguos basándose en LOD según las realizaciones.
- La fig. 28 ilustra un ejemplo de un dispositivo de recepción en la nube de puntos según las realizaciones.
- 15 La fig. 29 ilustra un ejemplo de un decodificador de geometría y un decodificador de atributo según las realizaciones.
- La fig. 30 ilustra una estructura de flujo de bits ejemplar para datos de la nube de puntos para transmisión/recepción según las realizaciones.
- La fig. 31 ilustra una estructura de flujo de bits ejemplar para datos de la nube de puntos según las realizaciones.
- 20 La fig. 32 ilustra una relación de conexión entre componentes en un flujo de bits de datos de la nube de puntos según las realizaciones.
- La fig. 33 ilustra una realización de una estructura sintáctica de un conjunto de parámetros de secuencia según las realizaciones.
- La fig. 34 ilustra una realización de una estructura sintáctica de un conjunto de parámetros geométricos según las realizaciones.
- 25 La fig. 35 ilustra una realización de una estructura sintáctica de un conjunto de parámetros de atributo según las realizaciones.
- La fig. 36 ilustra otra realización de una estructura sintáctica de un conjunto de parámetros de atributo según las realizaciones.
- 30 La fig. 37 ilustra una realización de una estructura sintáctica de un conjunto de parámetros de mosaico según las realizaciones.
- La fig. 38 ilustra una realización de una estructura sintáctica de flujo de bits() de segmento de geometría según las realizaciones.
- La fig. 39 ilustra una realización de una estructura sintáctica de cabecera de segmento de geometría según las realizaciones.
- 35 La fig. 40 ilustra una realización de una estructura sintáctica de datos de segmento de geometría según las realizaciones.
- La fig. 41 ilustra una realización de una estructura sintáctica de flujo de bits() de segmento de atributo según las realizaciones.
- 40 La fig. 42 ilustra una realización de una estructura sintáctica de cabecera de segmento de atributo según las realizaciones.
- La fig. 43 ilustra otra realización de una estructura sintáctica de cabecera de segmento de atributo según las realizaciones.
- La fig. 44 ilustra una realización de una estructura sintáctica de datos de segmento de atributo según las realizaciones.
- 45 La fig. 45 es un diagrama de flujo de un método de transmisión de datos de la nube de puntos según las realizaciones.

La fig. 46 es un diagrama de flujo de un método de recepción de datos de la nube de puntos según las realizaciones.

Mejor modo

5 A continuación se proporcionará la descripción en detalle según las realizaciones ejemplares descritas en la presente memoria, con referencia a los dibujos adjuntos. En aras de una breve descripción con referencia a los dibujos, los mismos componentes o componentes equivalentes pueden estar provistos de los mismos números de referencia, y no se repetirá la descripción de los mismos. Debería observarse que los siguientes ejemplos son solo para realizar la presente descripción y no limitan el alcance de la presente descripción. Lo que puede inferirse fácilmente por un experto en el campo técnico al que pertenece la presente invención a partir de la descripción detallada y los ejemplos de la presente descripción han de interpretarse como que están dentro del alcance de la presente descripción.

10 La descripción detallada en esta presente especificación debería interpretarse en todos los aspectos como ilustrativa y no restrictiva. El alcance de la descripción debería determinarse por las reivindicaciones adjuntas y sus equivalentes legales, y todos los cambios que entran dentro del significado y el intervalo de equivalencia de las reivindicaciones adjuntas pretenden incluirse en las mismas.

15 A continuación se hará referencia en detalle a las realizaciones preferidas de la presente descripción, cuyos ejemplos se ilustran en los dibujos adjuntos. La descripción detallada, que se dará a continuación con referencia a los dibujos adjuntos, pretende explicar realizaciones ejemplares de la presente descripción, en lugar de mostrar las únicas realizaciones que pueden implementarse según la presente descripción. La siguiente descripción detallada incluye detalles específicos para proporcionar una comprensión exhaustiva de la presente descripción. Sin embargo, será evidente para los expertos en la técnica que la presente descripción puede ponerse en práctica sin tales detalles específicos. Aunque la mayoría de los términos utilizados en esta especificación se han seleccionado de los generales utilizados ampliamente en la técnica, algunos términos se han seleccionado arbitrariamente por el solicitante y sus significados se explican en detalle en la siguiente descripción según sea necesario. Por lo tanto, la presente descripción debería comprenderse basándose en los significados previstos de los términos en lugar de sus nombres o significados sencillos. Además, los siguientes dibujos y descripción detallada no deberían interpretarse como limitados a las realizaciones descritas específicamente, sino que deberían interpretarse como que incluyen equivalentes o sustitutos de las realizaciones descritas en los dibujos y descripción detallada.

20 La fig. 1 muestra un sistema ejemplar de suministro de contenido de la nube de puntos según las realizaciones.

El sistema de suministro de contenido de la nube de puntos ilustrado en la fig. 1 puede incluir un dispositivo 10000 de transmisión y un dispositivo 10004 de recepción. El dispositivo 10000 de transmisión y el dispositivo 10004 de recepción son capaces de comunicación por cable o inalámbrica para transmitir y recibir datos de la nube de puntos.

35 El dispositivo 10000 de transmisión de datos de la nube de puntos según las realizaciones puede asegurar y procesar vídeo de la nube de puntos (o contenido de la nube de puntos) y transmitir el mismo. Según las realizaciones, el dispositivo 10000 de transmisión puede incluir una estación fija, un sistema transceptor base (BTS), una red, un dispositivo y/o sistema de inteligencia artificial (AI), un robot, un dispositivo y/o servidor AR/VR/XR. Según las realizaciones, el dispositivo 10000 de transmisión puede incluir un dispositivo, un robot, un vehículo, un dispositivo AR/VR/XR, un dispositivo portátil, un electrodoméstico, un dispositivo de Internet de las Cosas (IoT) y un dispositivo/servidor de AI que están configurados para realizar la comunicación con una estación base y/u otros dispositivos inalámbricos utilizando una tecnología de acceso por radio (por ejemplo, 5G Nueva RAT (NR), Evolución a Largo Plazo (LTE)).

40 El dispositivo 10000 de transmisión según las realizaciones incluye una unidad 10001 de adquisición de vídeo de la nube de puntos, un codificador 10002 de vídeo de la nube de puntos y/o un transmisor (o módulo de comunicación) 10003.

45 La unidad 10001 de adquisición de vídeo de la nube de puntos según las realizaciones adquiere un vídeo de la nube de puntos a través de un proceso de tratamiento tal como captura, síntesis o generación. El vídeo de la nube de puntos es contenido de la nube de puntos representado por una nube de puntos, que es un conjunto de puntos colocados en un espacio 3D, y puede denominarse datos de vídeo de la nube de puntos. El vídeo de la nube de puntos según las realizaciones puede incluir una o más tramas. Una trama representa una foto/imagen fija. Por lo tanto, el vídeo de la nube de puntos puede incluir una imagen/trama/foto de la nube de puntos, y puede denominarse imagen, trama o foto de la nube de puntos.

50 El codificador 10002 de vídeo de la nube de puntos según las realizaciones codifica los datos de vídeo de la nube de puntos adquiridos. El codificador 10002 de vídeo de la nube de puntos puede codificar los datos de vídeo de la nube de puntos basándose en la codificación de compresión de la nube de puntos. La codificación de compresión en la nube de puntos según las realizaciones puede incluir codificación de compresión en la nube de

puntos basada en geometría (G-PCC) y/o codificación de compresión en la nube de puntos basada en vídeo (V-PCC) o codificación de próxima generación. La codificación de compresión en la nube de puntos según las realizaciones no se limita a la realización descrita anteriormente. El codificador 10002 de vídeo de la nube de puntos puede emitir un flujo de bits que contiene los datos de vídeo de la nube de puntos codificados. El flujo de bits puede contener no solo los datos de vídeo de la nube de puntos codificados, sino también información de señalización relacionada con la codificación de los datos de vídeo de la nube de puntos.

El transmisor 10003 según las realizaciones transmite el flujo de bits que contiene los datos de vídeo de la nube de puntos codificados. El flujo de bits según las realizaciones se encapsula en un archivo o segmento (por ejemplo, un segmento de transmisión), y se transmite a través de diversas redes tales como una red de radiodifusión y/o una red de banda ancha. Aunque no se muestra en la figura, el dispositivo 10000 de transmisión puede incluir un encapsulador (o un módulo de encapsulación) configurado para realizar una operación de encapsulación. Según las realizaciones, el encapsulador puede incluirse en el transmisor 10003. Según las realizaciones, el archivo o segmento puede transmitirse al dispositivo 10004 de recepción a través de una red, o almacenarse en un medio de almacenamiento digital (por ejemplo, USB, SD, CD, DVD, Blu-ray, HDD, SSD, etc.). El transmisor 10003 según las realizaciones es capaz de comunicación por cable/inalámbrica con el dispositivo 10004 de recepción (o el receptor 10005) a través de una red de 4G, 5G, 6G, etc. Además, el transmisor puede realizar una operación de tratamiento de datos necesaria según el sistema de red (por ejemplo, un sistema de red de comunicación 4G, 5G o 6G). El dispositivo 10000 de transmisión, puede transmitir los datos encapsulados, en una forma bajo demanda.

El dispositivo 10004 de recepción según las realizaciones incluye un receptor 10005, un decodificador 10006 de vídeo de la nube de puntos y/o un renderizador 10007. Según las realizaciones, el dispositivo 10004 de recepción puede incluir un dispositivo, un robot, un vehículo, un dispositivo AR/VR/XR, un dispositivo portátil, un electrodoméstico, un dispositivo de Internet de las cosas (IoT) y un dispositivo/servidor de AI que están configurados para realizar la comunicación con una estación base y/u otros dispositivos inalámbricos utilizando una tecnología de acceso por radio (por ejemplo, 5G Nueva RAT (NR), Evolución a largo plazo (LTE)).

El receptor 10005 según las realizaciones recibe el flujo de bits que contiene los datos de vídeo de la nube de puntos o el archivo/segmento en el que el flujo de bits está encapsulado desde la red o medio de almacenamiento. El receptor 10005 puede realizar el tratamiento de datos necesario según el sistema de red (por ejemplo, un sistema de red de comunicación de 4G, 5G, 6G, etc.). El receptor 10005 según las realizaciones puede desencapsular el archivo/segmento recibido y emitir un flujo de bits. Según las realizaciones, el receptor 10005 puede incluir un desencapsulador (o un módulo de desencapsulación) configurado para realizar una operación de desencapsulación. El desencapsulador puede implementarse como un elemento (o componente o módulo) separado del receptor 10005.

El decodificador 10006 de vídeo de la nube de puntos decodifica el flujo de bits que contiene los datos de vídeo de la nube de puntos. El decodificador 10006 de vídeo de la nube de puntos puede decodificar los datos de vídeo de la nube de puntos según el método mediante el cual se codifican los datos de vídeo de la nube de puntos (por ejemplo, en un proceso inverso de la operación del codificador 10002 de vídeo de la nube de puntos). Por consiguiente, el decodificador 10006 de vídeo de la nube de puntos puede decodificar los datos de vídeo de la nube de puntos realizando una codificación de descompresión de la nube de puntos, que es el proceso inverso de la compresión de la nube de puntos. La codificación de descompresión de la nube de puntos incluye codificación de G-PCC.

El renderizador 10007 renderiza los datos de vídeo de la nube de puntos decodificados. El renderizador 10007 puede emitir contenido de la nube de puntos renderizando no solo los datos de vídeo de la nube de puntos sino también datos de audio. Según las realizaciones, el renderizador 10007 puede incluir un dispositivo de presentación configurada para mostrar el contenido de la nube de puntos. Según realizaciones, el dispositivo de presentación puede implementarse como un dispositivo o componente separado en lugar de incluirse en el renderizador 10007.

Las flechas indicadas por líneas de puntos en el dibujo representan una trayectoria de transmisión de información de retroalimentación adquirida por el dispositivo 10004 de recepción. La información de retroalimentación es información para reflejar la interactividad con un usuario que consume el contenido de la nube de puntos, e incluye información sobre el usuario (por ejemplo, información de orientación de la cabeza, información de la ventana de visualización y similares). En particular, cuando el contenido de la nube de puntos es contenido para un servicio (por ejemplo, servicio de conducción autónoma, etc.) que requiere interacción con el usuario, la información de retroalimentación puede proporcionarse al lado de transmisión de contenido (por ejemplo, el dispositivo 10000 de transmisión) y/o al proveedor de servicios. Según realizaciones, la información de retroalimentación puede utilizarse en el dispositivo 10004 de recepción así como en el dispositivo 10000 de transmisión, o puede no proporcionarse.

La información de orientación de la cabeza según las realizaciones es información sobre la posición, orientación, ángulo, movimiento de la cabeza del usuario y similares. El dispositivo 10004 de recepción según las

realizaciones puede calcular la información de la ventana de visualización basándose en la información de orientación de la cabeza. La información de la ventana de visualización puede ser información sobre una región de un vídeo de la nube de puntos que el usuario está viendo. Un punto de vista es un punto a través del cual el usuario está viendo el vídeo de la nube de puntos, y puede referirse a un punto central de la región de la ventana de visualización. Es decir, la ventana de visualización es una región centrada en el punto de vista, y el tamaño y la forma de la región pueden determinarse mediante un campo de visión (FOV). Por consiguiente, el dispositivo 10004 de recepción puede extraer la información de la ventana de visualización basándose en un FOV vertical u horizontal soportado por el dispositivo además de la información de orientación de la cabeza. Además, el dispositivo 10004 de recepción realiza análisis de mirada o similar para comprobar la manera en que el usuario consume una nube de puntos, una región que el usuario observa en el vídeo de la nube de puntos, un tiempo de mirada y similares. Según las realizaciones, el dispositivo 10004 de recepción puede transmitir información de retroalimentación que incluye el resultado del análisis de mirada al dispositivo 10000 de transmisión. La información de retroalimentación según las realizaciones puede adquirirse en el proceso de renderización y/o visualización. La información de retroalimentación según las realizaciones puede asegurarse mediante uno o más sensores incluidos en el dispositivo 10004 de recepción. Según las realizaciones, la información de retroalimentación puede asegurarse por el renderizador 10007 o un elemento externo separado (o dispositivo, componente o similar).

Las líneas de puntos en la fig. 1 representan un proceso de transmisión de la información de retroalimentación asegurada por el renderizador 10007. El sistema de suministro de contenido de la nube de puntos puede procesar (codificar/decodificar) datos de la nube de puntos basándose en la información de retroalimentación. Por consiguiente, el decodificador 10006 de vídeo de la nube de puntos puede realizar una operación de decodificación basándose en la información de retroalimentación. El dispositivo 10004 de recepción puede transmitir la información de retroalimentación al dispositivo 10000 de transmisión. El dispositivo 10000 de transmisión (o el codificador 10002 de vídeo de la nube de puntos) puede realizar una operación de codificación basándose en la información de retroalimentación. Por consiguiente, el sistema de suministro de contenido de la nube de puntos puede procesar de manera eficiente los datos necesarios (por ejemplo, datos de la nube de puntos correspondientes a la posición de la cabeza del usuario) basándose en la información de retroalimentación en lugar de procesar (codificar/decodificar) todos los datos de la nube de puntos, y proporcionar contenido de la nube de puntos al usuario.

Según las realizaciones, el dispositivo 10000 de transmisión puede denominarse codificador, dispositivo de transmisión, transmisor, sistema de transmisión o similar, y el dispositivo 10004 de recepción puede denominarse decodificador, dispositivo de recepción, receptor, sistema de recepción o similar.

Los datos de la nube de puntos procesados en el sistema de suministro de contenido de la nube de puntos de la fig. 1 según las realizaciones (a través de una serie de procesos de adquisición/codificación/transmisión/decodificación/renderización) pueden denominarse datos de contenido de la nube de puntos o datos de vídeo de la nube de puntos. Según realizaciones, los datos de contenido de la nube de puntos pueden utilizarse como un concepto que cubre metadatos o información de señalización relacionada con los datos de la nube de puntos.

Los elementos del sistema de suministro de contenido de la nube de puntos ilustrado en la fig. 1 pueden implementarse mediante hardware, software, un procesador y/o una combinación de los mismos.

La fig. 2 es un diagrama de bloques que ilustra una operación de suministro de contenido de la nube de puntos según las realizaciones.

El diagrama de bloques de la fig. 2 muestra la operación del sistema de suministro de contenido de la nube de puntos descrito en la fig. 1. Como se ha descrito anteriormente, el sistema de suministro de contenido de la nube de puntos puede procesar datos de la nube de puntos basándose en la codificación de compresión de la nube de puntos (por ejemplo, G-PCC).

El sistema de suministro de contenido de la nube de puntos según las realizaciones (por ejemplo, el dispositivo 10000 de transmisión de la nube de puntos o la unidad 10001 de adquisición de vídeo de la nube de puntos) puede adquirir un vídeo de la nube de puntos (20000). El vídeo de la nube de puntos está representado por una nube de puntos que pertenece a un sistema de coordenadas para expresar un espacio 3D. El vídeo de la nube de puntos según las realizaciones puede incluir un archivo Ply (formato de archivo de polígono o el formato de triángulo de Stanford). Cuando el vídeo de la nube de puntos tiene una o más tramas, el vídeo de la nube de puntos adquirido puede incluir uno o más archivos Ply. Los archivos Ply contienen datos de la nube de puntos, tales como geometría de puntos y/o atributos. La geometría incluye posiciones de puntos. La posición de cada punto puede representarse por parámetros (por ejemplo, valores de los ejes X, Y y Z) que representan un sistema de coordenadas tridimensional (por ejemplo, un sistema de coordenadas compuesto por los ejes X, Y y Z). Los atributos incluyen atributos de puntos (por ejemplo, información sobre textura, color (en YCbCr o RGB), reflectancia r , transparencia, etc. de cada punto). Un punto tiene uno o más atributos. Por ejemplo, un punto puede tener un atributo que es un color, o dos atributos que son color y reflectancia.

Según las realizaciones, la geometría puede denominarse posiciones, información de geometría, datos de geometría o similares, y el atributo puede denominarse atributos, información de atributo, datos de atributos o similares.

El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10000 de transmisión de la nube de puntos o la unidad 10001 de adquisición de vídeo de la nube de puntos) puede asegurar datos de la nube de puntos a partir de información (por ejemplo, información de profundidad, información de color, etc.) relacionada con el proceso de adquisición del vídeo de la nube de puntos.

El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10000 de transmisión o el codificador 10002 de vídeo de la nube de puntos) según las realizaciones puede codificar los datos (20001) de la nube de puntos. El sistema de suministro de contenido de la nube de puntos puede codificar los datos de la nube de puntos basándose en la codificación de compresión de la nube de puntos. Como se ha descrito anteriormente, los datos de la nube de puntos pueden incluir la geometría y atributos de un punto. Por consiguiente, el sistema de suministro de contenido de la nube de puntos puede realizar codificación geométrica de codificación de la geometría y emitir un flujo de bits de geometría. El sistema de suministro de contenido de la nube de puntos puede realizar codificación de atributos de codificación de atributos y emitir un flujo de bits de atributos. Según las realizaciones, el sistema de suministro de contenido de la nube de puntos puede realizar la codificación de atributos basándose en la codificación de geometría. El flujo de bits de geometría y el flujo de bits de atributos según las realizaciones pueden multiplexarse y emitirse como un flujo de bits. El flujo de bits según las realizaciones puede contener además información de señalización relacionada con la codificación de geometría y la codificación de atributos.

El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10000 de transmisión o el transmisor 10003) según las realizaciones puede transmitir los datos (20002) de la nube de puntos codificados. Como se ilustra en la fig. 1, los datos de la nube de puntos codificados pueden representarse por un flujo de bits de geometría y un flujo de bits de atributos. Además, los datos de la nube de puntos codificados pueden transmitirse en forma de un flujo de bits junto con información de señalización relacionada con la codificación de los datos de la nube de puntos (por ejemplo, información de señalización relacionada con la codificación de geometría y la codificación de atributos). El sistema de suministro de contenido de la nube de puntos puede encapsular un flujo de bits que transporta los datos de la nube de puntos codificados y transmitir los mismos en forma de un archivo o segmento.

El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10004 de recepción o el receptor 10005) según las realizaciones puede recibir el flujo de bits que contiene los datos codificados de la nube de puntos. Además, el sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10004 de recepción o el receptor 10005) puede demultiplexar el flujo de bits.

El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10004 de recepción o el decodificador 10005 de vídeo de la nube de puntos) puede decodificar los datos de la nube de puntos codificados (por ejemplo, el flujo de bits de geometría, el flujo de bits de atributos) transmitidos en el flujo de bits. El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10004 de recepción o el decodificador 10005 de vídeo de la nube de puntos) puede decodificar los datos de vídeo de la nube de puntos basándose en la información de señalización relacionada con la codificación de los datos de vídeo de la nube de puntos contenidos en el flujo de bits. El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10004 de recepción o el decodificador 10005 de vídeo de la nube de puntos) puede decodificar el flujo de bits de geometría para reconstruir las posiciones (geometría) de los puntos. El sistema de suministro de contenido de la nube de puntos puede reconstruir los atributos de los puntos decodificando el flujo de bits de atributos basándose en la geometría reconstruida. El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10004 de recepción o el decodificador 10005 de vídeo de la nube de puntos) puede reconstruir el vídeo de la nube de puntos basándose en las posiciones según la geometría reconstruida y los atributos decodificados.

El sistema de suministro de contenido de la nube de puntos según las realizaciones (por ejemplo, el dispositivo 10004 de recepción o el renderizador 10007) puede renderizar los datos (20004) de la nube de puntos decodificados. El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10004 de recepción o el renderizador 10007) puede renderizar la geometría y los atributos decodificados a través del proceso de decodificación, utilizando diversos métodos de renderización. Los puntos en el contenido de la nube de puntos pueden renderizarse en un vértice que tiene un cierto espesor, un cubo que tiene un tamaño mínimo específico centrado en la posición de vértice correspondiente, o un círculo centrado en la posición de vértice correspondiente. Todo o parte del contenido de la nube de puntos renderizado se proporciona al usuario a través de un dispositivo de visualización (por ejemplo, un dispositivo de visualización VR/AR, un dispositivo de visualización general, etc.).

El sistema de suministro de contenido de la nube de puntos (por ejemplo, el dispositivo 10004 de recepción) según las realizaciones puede asegurar información (20005) de retroalimentación. El sistema de suministro de

contenido de la nube de puntos puede codificar y/o decodificar datos de la nube de puntos basándose en la información de retroalimentación. La información de retroalimentación y la operación del sistema de suministro de contenido de la nube de puntos según las realizaciones son los mismos que la información de retroalimentación y la operación descrita con referencia a la fig. 1, y por lo tanto se omite la descripción detallada de los mismos.

La fig. 3 ilustra un proceso ejemplar de captura de un vídeo de la nube de puntos según las realizaciones.

La fig. 3 ilustra un proceso ejemplar de captura de vídeo de la nube de puntos del sistema de suministro de contenido de la nube de puntos descrito con referencia a las figs. 1 a 2.

El contenido de la nube de puntos incluye un vídeo de la nube de puntos (imágenes y/o vídeos) que representa un objeto y/o entorno ubicado en diversos espacios 3D (por ejemplo, un espacio 3D que representa un entorno real, un espacio 3D que representa un entorno virtual, etc.). Por consiguiente, el sistema de suministro de contenido de la nube de puntos según las realizaciones puede capturar un vídeo de la nube de puntos utilizando una o más cámaras (por ejemplo, una cámara infrarroja capaz de asegurar información de profundidad, una cámara RGB capaz de extraer información de color correspondiente a la información de profundidad, etc.), un proyector (por ejemplo, un proyector de patrones infrarrojos para asegurar información de profundidad), un LiDAR o similares. El sistema de suministro de contenido de la nube de puntos según las realizaciones puede extraer la forma de geometría compuesta de puntos en un espacio 3D a partir de la información de profundidad y extraer los atributos de cada punto a partir de la información de color para asegurar datos de la nube de puntos. Una imagen y/o vídeo según las realizaciones puede capturarse basándose en al menos una de la técnica orientada hacia dentro y la técnica orientada hacia fuera.

La parte izquierda de la fig. 3 ilustra la técnica orientada hacia dentro. La técnica orientada hacia dentro se refiere a una técnica de captura de imágenes de un objeto central con una o más cámaras (o sensores de cámara) colocadas alrededor del objeto central. La técnica orientada hacia dentro puede utilizarse para generar contenido de la nube de puntos que proporciona una imagen de 360 grados de un objeto clave al usuario (por ejemplo, contenido VR/AR que proporciona una imagen de 360 grados de un objeto (por ejemplo, un objeto clave tal como un personaje, jugador, objeto o actor) al usuario).

La parte derecha de la fig. 3 ilustra la técnica orientada hacia fuera. La técnica orientada hacia fuera se refiere a una técnica de captura de imágenes de un entorno de un objeto central en lugar del objeto central con una o más cámaras (o sensores de cámara) situadas alrededor del objeto central. La técnica orientada hacia fuera puede utilizarse para generar contenido de la nube de puntos para proporcionar un entorno circundante que aparece desde el punto de vista del usuario (por ejemplo, contenido que representa un entorno externo que puede proporcionarse a un usuario de un vehículo de autónomo).

Como se muestra en la fig. 3, el contenido de la nube de puntos puede generarse basándose en la operación de captura de una o más cámaras. En este caso, el sistema de coordenadas puede diferir entre las cámaras y, por consiguiente, el sistema de suministro de contenido de la nube de puntos pueden calibrar una o más cámaras para establecer un sistema de coordenadas global antes de la operación de captura. Además, el sistema de suministro de contenido de la nube de puntos puede generar contenido de la nube de puntos sintetizando una imagen y/o vídeo arbitrarios con una imagen y/o vídeo capturados por la técnica de captura descrita anteriormente. El sistema de suministro de contenido de la nube de puntos puede no realizar la operación de captura descrita en la fig. 3 cuando genera contenido de la nube de puntos que representa un espacio virtual. El sistema de suministro de contenido de la nube de puntos según las realizaciones puede realizar un tratamiento posterior en la imagen y/o vídeo capturados. En otras palabras, el sistema de suministro de contenido de la nube de puntos puede eliminar un área no deseada (por ejemplo, un fondo), reconocer un espacio al que se conectan las imágenes y/o vídeos capturados y, cuando hay un orificio espacial, realizar una operación de llenado del orificio espacial.

El sistema de suministro de contenido de la nube de puntos puede generar una pieza de contenido de la nube de puntos realizando una transformación de coordenadas en puntos del vídeo de la nube de puntos asegurado desde cada cámara. El sistema de suministro de contenido de la nube de puntos puede realizar una transformación de coordenadas en los puntos basándose en las coordenadas de la posición de cada cámara. Por consiguiente, el sistema de suministro de contenido de la nube de puntos puede generar contenido que representa un amplio intervalo, o puede generar contenido de la nube de puntos que tiene una alta densidad de puntos.

La fig. 4 ilustra un codificador de vídeo de la nube de puntos ejemplar según las realizaciones.

La fig. 4 muestra un ejemplo del codificador 10002 de vídeo de la nube de puntos de la fig. 1. El codificador de vídeo de la nube de puntos reconstruye y codifica datos de la nube de puntos (por ejemplo, posiciones y/o atributos de los puntos) para ajustar la calidad del contenido de la nube de puntos (a, por ejemplo, sin pérdidas, con pérdidas o casi sin pérdidas) según la condición o aplicaciones de red. Cuando el tamaño general del contenido de la nube de puntos es grande (por ejemplo, se da un contenido de la nube de puntos de 60 Gbps

durante 30 fps), el sistema de suministro de contenido de la nube de puntos puede no transmitir el contenido en tiempo real. Por consiguiente, el sistema de suministro de contenido de la nube de puntos puede reconstruir el contenido de la nube de puntos basándose en la tasa de bits máxima objetivo para proporcionar el mismo según el entorno de red o similar.

5 Como se describe con referencia a las figs. 1 a 2, el codificador de vídeo de la nube de puntos puede realizar codificación de geometría y codificación de atributos. La codificación de geometría se realiza antes de la codificación de atributos.

10 El codificador de vídeo de la nube de puntos según las realizaciones incluye un transformador de coordenadas (coordenadas de transformada) 40000, un cuantificador (cuantificar y retirar puntos (voxelizar)) 40001, un analizador de árbol octal (analizar árbol octal) 40002 y un analizador de aproximación de superficie (analizar aproximación de superficie) 40003, un codificador aritmético (codificador aritmético) 40004, un reconstructor geométrico (reconstruir geometría) 40005, un transformador de color (colores de transformada) 40006, un transformador de atributos (atributos de transformada) 40007, un transformador RAHT (RAHT) 40008, un generador LOD (generar LOD) 40009, un transformador de elevación (elevación) 40010, un cuantificador de coeficientes (coeficientes de cuantificación) 40011 y/o un codificador aritmético (codificación aritmética) 40012.

20 El transformador 40000 de coordenadas, el cuantificador 40001, el analizador 40002 de árbol octal, el analizador 40003 de aproximación de superficie, el codificador 40004 aritmético y el reconstructor 40005 de geometría pueden realizar codificación de geometría. La codificación geométrica según las realizaciones puede incluir codificación de geometría de árbol octal, codificación directa, codificación de geometría trisoup y codificación por entropía. La codificación directa y la codificación de geometría trisoup se aplican de manera selectiva o en combinación. La codificación geométrica no se limita al ejemplo descrito anteriormente.

25 Como se muestra en la figura, el transformador 40000 de coordenadas según las realizaciones recibe posiciones y las transforma en coordenadas. Por ejemplo, las posiciones pueden transformarse en información de posición en un espacio tridimensional (por ejemplo, un espacio tridimensional representado por un sistema de coordenadas XYZ). La información de posición en el espacio tridimensional según las realizaciones puede denominarse información de geometría.

30 El cuantificador 40001 según las realizaciones cuantifica la información de geometría. Por ejemplo, el cuantificador 40001 puede cuantificar los puntos basándose en un valor de posición mínimo de todos los puntos (por ejemplo, un valor mínimo en cada uno de los ejes X, Y y Z). El cuantificador 40001 realiza una operación de cuantificación de multiplicar la diferencia entre el valor de posición mínimo y el valor de posición de cada punto por un valor de escala de cuantificación preestablecido y, a continuación, encontrar el valor entero más cercano redondeando el valor obtenido a través de la multiplicación. Por lo tanto, uno o más puntos pueden tener la misma posición cuantificada (o valor de posición). El cuantificador 40001 según las realizaciones realiza voxelización basándose en las posiciones cuantificadas para reconstruir puntos cuantificados. La voxelización significa una unidad mínima que representa información de posición en el espacio 3D. Los puntos de contenido de la nube de puntos (o vídeo de la nube de puntos 3D) según las realizaciones pueden incluirse en uno o más vóxeles. El término vóxel, que es un compuesto de volumen y píxel, se refiere a un espacio cúbico 3D generado cuando un espacio 3D se divide en unidades (unidad=1,0) basándose en los ejes que representan el espacio 3D (por ejemplo, eje X, eje Y y eje Z). El cuantificador 40001 puede hacer coincidir grupos de puntos en el espacio 3D con vóxeles. Según las realizaciones, un vóxel puede incluir solo un punto. Según las realizaciones, un vóxel puede incluir uno o más puntos. Para expresar un vóxel como un punto, la posición del punto central de un vóxel puede establecerse basándose en las posiciones de uno o más puntos incluidos en el vóxel. En este caso, los atributos de todas las posiciones incluidas en un vóxel pueden combinarse y asignarse al vóxel.

45 El analizador 40002 de árbol octal según las realizaciones realiza codificación de geometría de árbol octal (o codificación de árbol octal) para presentar vóxeles en una estructura de árbol octal. La estructura de árbol octal representa puntos coincidentes con vóxeles, basándose en la estructura de árbol octal.

El analizador 40003 de aproximación de superficie según las realizaciones puede analizar y aproximar el árbol octal. El análisis y aproximación de árbol octal según las realizaciones es un proceso de análisis de una región que contiene una pluralidad de puntos para proporcionar eficientemente árbol octal y voxelización.

50 El codificador 40004 aritmético según las realizaciones realiza codificación por entropía en el árbol octal y/o el árbol octal aproximado. Por ejemplo, el esquema de codificación incluye codificación aritmética. Como resultado de la codificación, se genera un flujo de bits de geometría.

55 El transformador 40006 de color, el transformador 40007 de atributos, el transformador 40008 RAHT, el generador 40009 de LOD, el transformador 40010 de elevación, el cuantificador 40011 de coeficientes y/o el codificador 40012 aritmético realizan codificación de atributos. Como se ha descrito anteriormente, un punto puede tener uno o más atributos. La codificación de atributos según las formas de realización se aplica igualmente a los atributos que tienen un punto. Sin embargo, cuando un atributo (por ejemplo, color) incluye uno o más elementos, la codificación de atributos se aplica independientemente a cada elemento. La codificación de

- atributo según las realizaciones incluye la codificación por transformada de color, la codificación por transformada de atributos, la codificación por transformada jerárquica adaptativa de región (RAHT), la codificación de predicción jerárquica de contiguos más cercanos basada en interpolación (transformada de predicción) y la predicción jerárquica de contiguos más cercanos basada en interpolación con una codificación de etapa de actualización/elevación (transformada de elevación). Dependiendo del contenido de la nube de puntos, la codificación RAHT, la codificación por transformada de predicción y la codificación por transformada de elevación descritas anteriormente pueden utilizarse selectivamente, o puede utilizarse una combinación de uno o más de los esquemas de codificación. La codificación de atributos según las formas de realización no se limita al ejemplo descrito anteriormente.
- 10 El transformador 40006 de color según las realizaciones realiza la codificación por transformada de color de los valores de color (o texturas) de transformación incluidos en los atributos. Por ejemplo, el transformador 40006 de color puede transformar el formato de la información de color (por ejemplo, de RGB a YCbCr). La operación del transformador 40006 de color según las realizaciones puede aplicarse opcionalmente según los valores de color incluidos en los atributos.
- 15 El reconstructor 40005 de geometría según las realizaciones reconstruye (descomprime) el árbol octal y/o el árbol octal aproximado. El reconstructor 40005 de geometría reconstruye el árbol octal/vóxeles basándose en el resultado de analizar la distribución de puntos. El árbol octal/vóxeles reconstruidos pueden denominarse geometría reconstruida (geometría restaurada).
- 20 El transformador 40007 de atributos según las realizaciones realiza la transformación de atributos para transformar los atributos basándose en la geometría reconstruida y/o las posiciones en las que no se realiza la codificación de geometría. Como se ha descrito anteriormente, dado que los atributos dependen de la geometría, el transformador 40007 de atributos puede transformar los atributos basándose en la información de geometría reconstruida. Por ejemplo, basándose en el valor de posición de un punto incluido en un vóxel, el transformador 40007 de atributos puede transformar el atributo del punto en la posición. Como se ha descrito anteriormente,
- 25 cuando la posición del centro de un vóxel se establece basándose en las posiciones de uno o más puntos incluidos en el vóxel, el transformador 40007 de atributos transforma los atributos del uno o más puntos. Cuando se realiza la codificación de geometría trisoup, el transformador 40007 de atributos puede transformar los atributos basándose en la codificación de geometría trisoup.
- 30 El transformador 40007 de atributos puede realizar la transformación de atributos calculando el promedio de atributos o valores de atributos de puntos contiguos (por ejemplo, color o reflectancia de cada punto) dentro de una posición/radio específico desde la posición (o valor de posición) del centro de cada vóxel. El transformador 40007 de atributos puede aplicar un peso según la distancia desde el centro a cada punto en el cálculo del promedio. Por consiguiente, cada vóxel tiene una posición y un atributo calculado (o valor de atributo).
- 35 El transformador 40007 de atributos puede buscar puntos contiguos existentes dentro de una posición/radio específico desde la posición del centro de cada vóxel basándose en el árbol K-D o el código de Morton. El árbol K-D es un árbol de búsqueda binario y soporta una estructura de datos capaz de gestionar puntos basándose en las posiciones de tal manera que la búsqueda de contiguos más cercanos (NNS) puede realizarse rápidamente. El código de Morton se genera presentando coordenadas (por ejemplo, (x, y, z)) que representan posiciones 3D de todos los puntos como valores de bits y mezclando los bits. Por ejemplo, cuando las coordenadas que representan la posición de un punto son (5, 9, 1), los valores de bit para las coordenadas son (0101, 1001, 0001).
- 40 La mezcla de los valores de bit según el índice de bit en orden de z, y, y x produce 010001000111. Este valor se expresa como un número decimal de 1095. Es decir, el valor del código de Morton del punto que tiene coordenadas (5, 9, 1) es 1095. El transformador 40007 de atributos puede ordenar los puntos basándose en los valores de código de Morton y realizar NNS a través de un proceso de primer recorrido de profundidad. Después de la operación de transformación de atributos, el árbol K-D o el código de Morton se utiliza cuando el NNS es necesario en otro proceso de transformación para la codificación de atributos.
- 45 Como se muestra en la figura, los atributos transformados se introducen en el transformador 40008 RAHT y/o el generador 40009 de LOD.
- 50 El transformador 40008 RAHT según las realizaciones realiza codificación RAHT para predecir información de atributo basándose en la información de geometría reconstruida. Por ejemplo, el transformador 40008 RAHT puede predecir información de atributo de un nodo en un nivel superior en el árbol octal basándose en la información de atributo asociada con un nodo en un nivel inferior en el árbol octal.
- 55 El generador 40009 de LOD según las realizaciones genera un nivel de detalle (LOD). El LOD según las realizaciones es un grado de detalle del contenido de la nube de puntos. A medida que disminuye el valor de LOD, indica que se degrada el detalle del contenido de la nube de puntos. A medida que aumenta el valor de LOD, indica que se mejora el detalle del contenido de la nube de puntos. Los puntos pueden clasificarse por el LOD.

El transformador 40010 de elevación según las realizaciones realiza la codificación por transformada de elevación de transformar los atributos en una nube de puntos basándose en pesos. Como se ha descrito anteriormente, la codificación por transformada de elevación puede aplicarse opcionalmente.

5 El cuantificador 40011 de coeficientes según las realizaciones cuantifica los atributos codificados por atributo basándose en coeficientes.

El codificador 40012 aritmético según las realizaciones codifica los atributos cuantificados basándose en la codificación aritmética.

10 Aunque no se muestra en la figura, los elementos del codificador de vídeo de la nube de puntos de la fig. 4 pueden implementarse mediante hardware que incluye uno o más procesadores o circuitos integrados configurados para comunicarse con una o más memorias incluidas en el aparato de suministro de contenido de la nube de puntos, software, firmware o una combinación de los mismos. El uno o más procesadores pueden realizar al menos una de las operaciones y/o funciones de los elementos del codificador de vídeo de la nube de puntos de la fig. 4 descrito anteriormente. Adicionalmente, el uno o más procesadores pueden operar o ejecutar un conjunto de programas de software y/o instrucciones para realizar las operaciones y/o funciones de los elementos del codificador de vídeo de la nube de puntos de la fig. 4. La una o más memorias según las realizaciones pueden incluir una memoria de acceso aleatorio de alta velocidad, o incluir una memoria no volátil (por ejemplo, uno o más dispositivos de almacenamiento en disco magnético, dispositivos de memoria flash u otros dispositivos de memoria de estado sólido no volátiles).

La fig. 5 muestra un ejemplo de vóxeles según las realizaciones.

20 La fig. 5 muestra vóxeles colocados en un espacio 3D representado por un sistema de coordenadas compuesto por tres ejes, que son el eje X, el eje Y y el eje Z. Como se describe con referencia a la fig. 4, el codificador de vídeo de la nube de puntos (por ejemplo, el cuantificador 40001) puede realizar la voxelización. Vóxel se refiere a un espacio cúbico 3D generado cuando un espacio 3D se divide en unidades (unidad=1,0) basándose en los ejes que representan el espacio 3D (por ejemplo, eje X, eje Y y eje Z). La fig. 5 muestra un ejemplo de vóxeles generados a través de una estructura de árbol octal en la que una caja delimitadora alineada con eje cúbico definida por dos polos (0, 0, 0) y (2^d, 2^d, 2^d) está subdividida recursivamente. Un vóxel incluye al menos un punto. Las coordenadas espaciales de un vóxel pueden estimarse a partir de la relación posicional con un grupo de vóxeles. Como se ha descrito anteriormente, un vóxel tiene un atributo (tal como color o reflectancia) como píxeles de una imagen/vídeo 2D. Los detalles del vóxel son los mismos que los descritos con referencia a la fig. 4, y por lo tanto se omite una descripción de los mismos.

La fig. 6 muestra un ejemplo de un árbol octal y un código de ocupación según las realizaciones.

35 Como se describe con referencia a las figs. 1 a 4, el sistema de suministro de contenido de la nube de puntos (codificador 10002 de vídeo de la nube de puntos) o el analizador 40002 de árbol octal del codificador de vídeo de la nube de puntos realiza codificación de geometría de árbol octal (o codificación de árbol octal) basándose en una estructura de árbol octal para gestionar de manera eficiente la región y/o posición del vóxel.

La parte superior de la fig. 6 muestra una estructura de árbol octal. El espacio 3D del contenido de la nube de puntos según las realizaciones se representa por ejes (por ejemplo, eje X, eje Y y eje Z) del sistema de coordenadas. La estructura de árbol octal se crea subdividiendo recursivamente una zona de delimitación alineada con eje cúbico definida por dos polos (0, 0, 0) y (2^d, 2^d, 2^d). Aquí, 2^d puede establecerse a un valor que constituye la zona de delimitación más pequeña que rodea todos los puntos del contenido de la nube de puntos (o vídeo de la nube de puntos). Aquí, d indica la profundidad del árbol octal. El valor de d se determina en la ecuación 1. En la ecuación 1, (x^{int}_n, y^{int}_n, z^{int}_n) denota las posiciones (o valores de posición) de puntos cuantificados.

Ecuación 1

$$d = \text{Ceil} \left(\log_2 \left(\max(x_n^{\text{int}}, y_n^{\text{int}}, z_n^{\text{int}}, n = 1, \dots, N) + 1 \right) \right)$$

45 Como se muestra en el medio de la parte superior de la fig. 6, todo el espacio 3D puede dividirse en ocho espacios según la partición. Cada espacio dividido está representado por un cubo con seis caras. Como se muestra en la parte superior derecha de la fig. 6, cada uno de los ocho espacios se divide de nuevo basándose en los ejes del sistema de coordenadas (por ejemplo, eje X, eje Y y eje Z). Por consiguiente, cada espacio se divide en ocho espacios más pequeños. El espacio más pequeño dividido también está representado por un cubo con seis caras. Este esquema de partición se aplica hasta que el nodo hoja del árbol octal se convierte en un vóxel.

La parte inferior de la fig. 6 muestra un código de ocupación de árbol octal. El código de ocupación del árbol octal se genera para indicar si cada uno de los ocho espacios divididos generados dividiendo un espacio contiene al

menos un punto. Por consiguiente, un único código de ocupación está representado por ocho nodos secundarios. Cada nodo secundario representa la ocupación de un espacio dividido, y el nodo secundario tiene un valor en 1 bit. Por consiguiente, el código de ocupación se representa como un código de 8 bits. Es decir, cuando al menos un punto está contenido en el espacio correspondiente a un nodo secundario, al nodo se le asigna un valor de 1. Cuando no hay ningún punto contenido en el espacio correspondiente al nodo secundario (el espacio está vacío), al nodo se le asigna un valor de 0. Dado que el código de ocupación mostrado en la fig. 6 es 00100001, indica que los espacios correspondientes al tercer nodo secundario y el octavo nodo secundario entre los ocho nodos secundarios contienen cada uno al menos un punto. Como se muestra en la figura, cada uno del tercer nodo secundario y el octavo nodo secundario tiene ocho nodos secundarios, y los nodos secundarios están representados por un código de ocupación de 8 bits. La figura muestra que el código de ocupación del tercer nodo secundario es 10000111, y el código de ocupación del octavo nodo secundario es 01001111. El codificador de vídeo de la nube de puntos (por ejemplo, el codificador 40004 aritmético) según las realizaciones puede realizar codificación por entropía en los códigos de ocupación. Para aumentar la eficiencia de compresión, el codificador de vídeo de la nube de puntos puede realizar intra/intercodificación en los códigos de ocupación. El dispositivo de recepción (por ejemplo, el dispositivo 10004 de recepción o el decodificador 10006 de vídeo de la nube de puntos) según las realizaciones reconstruye el árbol octal basándose en los códigos de ocupación.

El codificador de vídeo de la nube de puntos (por ejemplo, el analizador 40002 de árbol octal) según las realizaciones puede realizar la voxelización y la codificación de árbol octal para almacenar las posiciones de los puntos. Sin embargo, los puntos no siempre se distribuyen uniformemente en el espacio 3D y, por consiguiente, puede haber una región específica en la que estén presentes menos puntos. Por consiguiente, es ineficaz realizar voxelización para todo el espacio 3D. Por ejemplo, cuando una región específica contiene pocos puntos, no es necesario realizar voxelización en la región específica.

Por consiguiente, para la región específica descrita anteriormente (o un nodo distinto del nodo hoja del árbol octal), el codificador de vídeo de la nube de puntos según las realizaciones puede omitir la voxelización y realizar una codificación directa para codificar directamente las posiciones de los puntos incluidos en la región específica. Las coordenadas de un punto de codificación directa según las realizaciones se denominan modo de codificación directa (DCM). El codificador de vídeo de la nube de puntos según las realizaciones también puede realizar codificación de geometría trisoup, que es para reconstruir las posiciones de los puntos en la región específica (o nodo) basándose en vóxeles, basándose en un modelo de superficie. La codificación de geometría trisoup es codificación de geometría que representa un objeto como una serie de mallas triangulares. Por consiguiente, el decodificador de vídeo de la nube de puntos puede generar una nube de puntos a partir de la superficie de malla. La codificación directa y la codificación de geometría trisoup según las realizaciones pueden realizarse de manera selectiva. Además, la codificación directa y la codificación de geometría trisoup según las realizaciones pueden realizarse en combinación con la codificación de geometría de árbol octal (o codificación de árbol octal).

Para realizar la codificación directa, debería activarse la opción de utilizar el modo directo para aplicar la codificación directa. Un nodo al que se ha de aplicar la codificación directa no es un nodo hoja, y puntos menores que un umbral deberían estar presentes dentro de un nodo específico. Además, el número total de puntos a los que se ha de aplicar la codificación directa no debería superar un umbral preestablecido. Cuando se cumplen las condiciones anteriores, el codificador de vídeo de la nube de puntos (o el codificador 40004 aritmético) según las realizaciones puede realizar codificación por entropía en las posiciones (o valores de posición) de los puntos.

El codificador de vídeo de la nube de puntos (por ejemplo, el analizador 40003 de aproximación de superficie) según las realizaciones puede determinar un nivel específico del árbol octal (un nivel menor que la profundidad del árbol octal), y el modelo de superficie puede utilizarse comenzando con ese nivel para realizar codificación de geometría trisoup para reconstruir las posiciones de puntos en la región del nodo basándose en vóxeles (modo trisoup). El codificador de vídeo de la nube de puntos según las realizaciones puede especificar un nivel al que se ha de aplicar la codificación de geometría trisoup. Por ejemplo, cuando el nivel específico es igual a la profundidad del árbol octal, el codificador de vídeo de la nube de puntos no opera en el modo trisoup. En otras palabras, el codificador de vídeo de la nube de puntos según las realizaciones puede operar en el modo trisoup solo cuando el nivel especificado es menor que el valor de profundidad del árbol octal. La región de cubo 3D de los nodos al nivel especificado según las realizaciones se denomina un bloque. Un bloque puede incluir uno o más vóxeles. El bloque o vóxel puede corresponder a un ladrillo. La geometría se representa como una superficie dentro de cada bloque. La superficie según las realizaciones puede intersectarse con cada borde de un bloque como máximo una vez.

Un bloque tiene 12 bordes, y por consiguiente hay al menos 12 intersecciones en un bloque. Cada intersección se denomina vértice (o ápice). Un vértice presente a lo largo de un borde se detecta cuando hay al menos un vóxel ocupado adyacente al borde entre todos los bloques que comparten el borde. El vóxel ocupado según las realizaciones se refiere a un vóxel que contiene un punto. La posición del vértice detectada a lo largo del borde es la posición promedio a lo largo del borde de todos los vóxeles adyacentes al borde entre todos los bloques que comparten el borde.

Una vez que se detecta el vértice, el codificador de vídeo de la nube de puntos según las realizaciones puede realizar codificación por entropía en el punto de inicio (x, y, z) del borde, el vector de dirección (Δx , Δy , Δz) del borde y el valor de posición del vértice (valor de posición relativa dentro del borde). Cuando se aplica la codificación de geometría trisoup, el codificador de vídeo en nube de puntos según las realizaciones (por ejemplo, el reconstructor 40005 de geometría) puede generar geometría restaurada (geometría reconstruida) realizando los procesos de reconstrucción de triángulo, muestreo ascendente y voxelización.

Los vértices situados en el borde del bloque determinan una superficie que pasa a través del bloque. La superficie según las realizaciones es un polígono no plano. En el proceso de reconstrucción de triángulos, una superficie representada por un triángulo se reconstruye basándose en el punto de partida del borde, el vector de dirección del borde y los valores de posición de los vértices. El proceso de reconstrucción de triángulos se realiza según la Ecuación 2 mediante: i) el cálculo del valor centroide de cada vértice, ii) la resta del valor central de cada valor de vértice, y iii) la estimación de la suma de los cuadrados de los valores obtenidos mediante la resta.

Ecuación 2

$$\textcircled{1} \begin{bmatrix} \mu_x \\ \mu_y \\ \mu_z \end{bmatrix} = \frac{1}{n} \sum_{i=1}^n \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \quad \textcircled{2} \begin{bmatrix} \bar{x}_i \\ \bar{y}_i \\ \bar{z}_i \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} - \begin{bmatrix} \mu_x \\ \mu_y \\ \mu_z \end{bmatrix} \quad \textcircled{3} \begin{bmatrix} \sigma_x^2 \\ \sigma_y^2 \\ \sigma_z^2 \end{bmatrix} = \sum_{i=1}^n \begin{bmatrix} \bar{x}_i^2 \\ \bar{y}_i^2 \\ \bar{z}_i^2 \end{bmatrix}$$

A continuación, se estima el valor mínimo de la suma, y el proceso de proyección se realiza según el eje con el valor mínimo. Por ejemplo, cuando el elemento x es el mínimo, cada vértice se proyecta en el eje x con respecto al centro del bloque, y se proyecta en el plano (y, z). Cuando los valores obtenidos a través de la proyección en el plano (y, z) son (ai, bi), el valor de θ se estima a través de $\text{atan2}(b_i, a_i)$, y los vértices se ordenan basándose en el valor de θ . La tabla 1 a continuación muestra una combinación de vértices para crear un triángulo según el número de vértices. Los vértices están ordenados de 1 a n. La tabla 1 a continuación muestra que para cuatro vértices, se pueden construir dos triángulos según combinaciones de vértices. El primer triángulo puede consistir en los vértices 1, 2 y 3 entre los vértices ordenados, y el segundo triángulo puede consistir en los vértices 3, 4 y 1 entre los vértices ordenados.

[Tabla 1] Triángulos formados a partir de vértices ordenados 1,..., n

n	Triángulos
3	(1,2,3)
4	(1,2,3), (3,4,1)
5	(1,2,3), (3,4,5), (5,1,3)
6	(1,2,3), (3,4,5), (5,6,1), (1,3,5)
7	(1,2,3), (3,4,5), (5,6,7), (7,1,3), (3,5,7)
8	(1,2,3), (3,4,5), (5,6,7), (7,8,1), (1,3,5), (5,7,1)
9	(1,2,3), (3,4,5), (5,6,7), (7,8,9), (9,1,3), (3,5,7), (7,9,3)
10	(1,2,3), (3,4,5), (5,6,7), (7,8,9), (9,10,1), (1,3,5), (5,7,9), (9,1,5)
11	(1,2,3), (3,4,5), (5,6,7), (7,8,9), (9,10,11), (11,1,3), (3,5,7), (7,9,11), (11,3,7)
12	(1,2,3), (3,4,5), (5,6,7), (7,8,9), (9,10,11), (11,12,1), (1,3,5), (5,7,9), (9,11,1), (1,5,9)

El proceso de muestreo ascendente se realiza para añadir puntos en el medio a lo largo del borde del triángulo y realizar la voxelización. Los puntos añadidos se generan basándose en el factor de muestreo ascendente y la anchura del bloque. Los puntos añadidos se denominan vértices refinados. El codificador de vídeo en nube de puntos según las realizaciones puede voxelizar los vértices refinados. Además, el codificador de vídeo de la nube de puntos puede realizar codificación de atributos basándose en las posiciones voxelizadas (o valores de

posición). La fig. 7 muestra un ejemplo de un patrón de nodo contiguo según las realizaciones. Con el fin de aumentar la eficiencia de compresión del vídeo de la nube de puntos, el codificador de vídeo de la nube de puntos según las realizaciones puede realizar codificación por entropía basándose en codificación aritmética adaptativa al contexto.

Como se describe con referencia a las figs. 1 a 6, el sistema de suministro de contenido de la nube de puntos o el codificador 10002 de vídeo de la nube de puntos de la fig. 1, o el codificador de vídeo de la nube de puntos o el codificador 40004 aritmético de la fig. 4 pueden realizar la codificación por entropía en el código de ocupación inmediatamente. Además, el sistema de suministro de contenido de la nube de puntos o el codificador de vídeo de la nube de puntos puede realizar codificación por entropía (intracodificación) basándose en el código de ocupación del nodo actual y la ocupación de nodos contiguos, o realizar codificación por entropía (intercodificación) basándose en el código de ocupación de la trama anterior. Una trama según las realizaciones representa un conjunto de vídeos de la nube de puntos generados al mismo tiempo. La eficiencia de compresión de intracodificación/intercodificación según las realizaciones puede depender del número de nodos contiguos a los que se hace referencia. Cuando los bits aumentan, la operación se complica, pero la codificación puede inclinarse hacia un lado, lo que puede aumentar la eficiencia de compresión. Por ejemplo, cuando se da un contexto de 3 bits, la codificación debe realizarse utilizando $2^3 = 8$ métodos. La parte dividida para la codificación afecta a la complejidad de la implementación. Por consiguiente, es necesario cumplir un nivel apropiado de eficiencia y complejidad de compresión.

La fig. 7 ilustra un proceso de obtención de un patrón de ocupación basándose en la ocupación de nodos contiguos. El codificador de vídeo de la nube de puntos según las realizaciones determina la ocupación de nodos contiguos de cada nodo del árbol octal y obtiene un valor de un patrón contiguo. El patrón de nodo contiguo se utiliza para inferir el patrón de ocupación del nodo. La parte superior de la fig. 7 muestra un cubo correspondiente a un nodo (un cubo situado en el medio) y seis cubos (nodos contiguos) que comparten al menos una cara con el cubo. Los nodos mostrados en la figura son nodos de la misma profundidad. Los números mostrados en la figura representan pesos (1, 2, 4, 8, 16 y 32) asociados con los seis nodos, respectivamente. Los pesos se asignan secuencialmente según las posiciones de los nodos contiguos.

La parte inferior de la fig. 7 muestra valores de patrones de nodos contiguos. Un valor de patrón de nodo contiguo es la suma de valores multiplicados por el peso de un nodo contiguo ocupado (un nodo contiguo que tiene un punto). Por consiguiente, los valores de patrón de nodo contiguo son de 0 a 63. Cuando el valor de patrón de nodo contiguo es 0, indica que no hay ningún nodo que tenga un punto (nodo no ocupado) entre los nodos contiguos del nodo. Cuando el valor de patrón de nodo contiguo es 63, indica que todos los nodos contiguos son nodos ocupados. Como se muestra en la figura, dado que los nodos contiguos a los que se asignan los pesos 1, 2, 4 y 8 son nodos ocupados, el valor del patrón de nodos contiguos es 15, la suma de 1, 2, 4 y 8. El codificador de vídeo de la nube de puntos puede realizar la codificación según el valor de patrón de nodo contiguo (por ejemplo, cuando el valor de patrón de nodo contiguo es 63, se pueden realizar 64 clases de codificación). Según las realizaciones, el codificador de vídeo de la nube de puntos puede reducir la complejidad de codificación cambiando un valor de patrón de nodo contiguo (por ejemplo, basándose en una tabla mediante la cual 64 se cambia a 10 o 6).

La fig. 8 ilustra un ejemplo de configuración de puntos en cada LOD según las realizaciones.

Como se describe con referencia a las figs. 1 a 7, la geometría codificada se reconstruye (descomprime) antes de realizar la codificación de atributos. Cuando se aplica la codificación directa, la operación de reconstrucción de geometría puede incluir cambiar la colocación de puntos codificados directamente (por ejemplo, colocar los puntos codificados directamente delante de los datos de la nube de puntos). Cuando se aplica la codificación de geometría trisoup, el proceso de reconstrucción de geometría se realiza a través de reconstrucción triangular, muestreo ascendente y voxelización. Dado que el atributo depende de la geometría, la codificación de atributos se realiza basándose en la geometría reconstruida.

El codificador de vídeo de la nube de puntos (por ejemplo, el generador 40009 de LOD) puede clasificar (reorganizar o agrupar) puntos por LOD. La fig. 8 muestra el contenido de la nube de puntos correspondiente a LOD. La imagen más a la izquierda en la fig. 8 representa el contenido de la nube de puntos original. La segunda imagen de la izquierda de la fig. 8 representa la distribución de los puntos en el LOD inferior, y la imagen de la derecha en la fig. 8 representa la distribución de los puntos en el LOD superior. Es decir, los puntos en el LOD inferior están distribuidos escasamente, y los puntos en el LOD superior están distribuidos densamente. Es decir, a medida que el LOD se eleva en la dirección apuntada por la flecha indicada en la parte inferior de la fig. 8, el espacio (o distancia) entre puntos se estrecha.

La fig. 9 ilustra un ejemplo de configuración de puntos para cada LOD según las realizaciones.

Como se describe con referencia a las figs. 1 a 8, el sistema de suministro de contenido de la nube de puntos, o el codificador de vídeo de la nube de puntos (por ejemplo, el codificador 10002 de vídeo de la nube de puntos de la fig. 1, el codificador de vídeo de la nube de puntos de la fig. 4, o el generador 40009 de LOD) puede generar un LOD. El LOD se genera reorganizando los puntos en un conjunto de niveles de refinamiento según un valor

de distancia de LOD establecido (o un conjunto de distancias euclidianas). El proceso de generación de LOD se realiza no solo por el codificador de vídeo de la nube de puntos, sino también por el decodificador de vídeo de la nube de puntos.

- 5 La parte superior de la fig. 9 muestra ejemplos (P0 a P9) de puntos del contenido de la nube de puntos distribuidos en un espacio 3D. En la fig. 9, el orden original representa el orden de los puntos P0 a P9 antes de la generación de LOD. En la fig. 9, el orden basado en LOD representa el orden de los puntos según la generación de LOD. Los puntos se reorganizan mediante LOD. Además, un LOD superior contiene los puntos que pertenecen a LOD inferiores. Como se muestra en la fig. 9, LOD0 contiene P0, P5, P4 y P2. LOD1 contiene los puntos de LOD0, P1, P6 y P3. LOD2 contiene los puntos de LOD0, los puntos de LOD1, P9, P8 y P7.
- 10 Como se describe con referencia a la fig. 4, el codificador de vídeo de la nube de puntos según las realizaciones puede realizar una codificación por transformada de predicción basada en LOD, una codificación por transformada de elevación basada en LOD y una codificación por transformada de RAHT selectivamente o en combinación.
- 15 El codificador de vídeo de la nube de puntos según las realizaciones puede generar un predictor para puntos para realizar la codificación por transformada de predicción basándose en el LOD para establecer un atributo predicho (o valor de atributo predicho) de cada punto. Es decir, se pueden generar N predictores para N puntos. El predictor según las realizaciones puede calcular un peso (= 1/distancia) basándose en el valor de LOD de cada punto, indexando la información sobre puntos contiguos presentes dentro de una distancia establecida para cada LOD y una distancia a los puntos contiguos.
- 20 El atributo predicho (o valor de atributo) según las realizaciones se establece en el promedio de valores obtenidos multiplicando los atributos (o valores de atributos) (por ejemplo, color, reflectancia, etc.) de puntos contiguos establecidos en el predictor de cada punto por un peso (o valor de peso) calculado basándose en la distancia a cada punto contiguo. El codificador de vídeo de la nube de puntos según las realizaciones (por ejemplo, el cuantificador 40011 de coeficientes) puede cuantificar y cuantificar de manera inversa el residuo de cada punto (que puede denominarse atributo residual, valor de atributo residual, valor residual de predicción de atributos o valor de atributo de error de predicción, etc.) obtenido restando un atributo predicho (o valor de atributo) a cada punto del atributo (es decir, valor de atributo original) de cada punto. El proceso de cuantificación realizado para un valor de atributo residual en un dispositivo de transmisión se configura como se muestra en la tabla 2. El proceso de cuantificación inversa realizado para un valor de atributo residual en un dispositivo de recepción se configura como se muestra en la tabla 3.
- 25
- 30

[Tabla 2]

int PCCQuantization(int value, int quantStep) {
if(value >=0) {
return floor(value / quantStep + 1.0 / 3.0);
} else {
return -floor(-value / quantStep + 1.0 / 3.0);
}
}

[Tabla 3]

int PCCInverseQuantization(int value, int quantStep) {
if(quantStep ==0) {
return value;
} else {
return value * quantStep;
}
}

Cuando el predictor de cada punto tiene puntos contiguos, el codificador de vídeo de la nube de puntos (por ejemplo, el codificador 40012 aritmético) según las realizaciones puede realizar codificación por entropía en los valores de atributos residual cuantificados e inversamente cuantificados como se ha descrito anteriormente. Cuando el predictor de cada punto no tiene punto contiguo, el codificador de vídeo de la nube de puntos según las realizaciones (por ejemplo, el codificador 40012 aritmético) puede realizar codificación por entropía en los atributos del punto correspondiente sin realizar la operación descrita anteriormente. El codificador de vídeo de la nube de puntos según las realizaciones (por ejemplo, el transformador 40010 de elevación) puede generar un predictor de cada punto, establecer el LOD calculado y registrar puntos contiguos en el predictor y establecer pesos según las distancias a puntos contiguos para realizar la codificación por transformada de elevación. La codificación por transformada de elevación según las realizaciones es similar a la codificación por transformada de predicción descrita anteriormente, pero difiere de la misma en que los pesos se aplican acumulativamente a valores de atributos. El proceso de aplicar de manera acumulativa pesos a los valores de atributos según las realizaciones se configura como sigue.

1) Crear una matriz de peso de cuantificación (QW) para almacenar el valor de peso de cada punto. El valor inicial de todos los elementos de QW es 1,0. Multiplicar los valores de QW de los índices de predictor de los nodos contiguos registrados en el predictor por el peso del predictor del punto actual, y sumar los valores obtenidos por la multiplicación.

2) Proceso de predicción de elevación: Restar el valor obtenido multiplicando el valor de atributo del punto por el peso del valor de atributo existente para calcular un valor de atributo predicho.

3) Crear matrices temporales llamadas actualizar peso y actualizar e inicializar las matrices temporales a cero.

4) Añadir acumulativamente las ponderaciones calculadas multiplicando las ponderaciones calculadas para todos los predictores por una ponderación almacenada en la QW correspondiente a un índice de predictor a la matriz de pesos de actualización como índices de nodos contiguos. Añadir acumulativamente, a la matriz de actualización, un valor obtenido multiplicando el valor de atributo del índice de un nodo contiguo por el peso calculado.

5) Proceso de actualización de elevación: Dividir los valores de atributos de la matriz de actualización para todos los predictores por el valor de peso de la matriz de actualización de peso del índice de predictor, y añadir el valor de atributo existente a los valores obtenidos por la división.

6) Calcular los atributos predichos multiplicando los valores de atributos actualizados a través del proceso de actualización de elevación por el peso actualizado a través del proceso de predicción de elevación (almacenado en el QW) para todos los predictores. El codificador de vídeo de la nube de puntos (por ejemplo, el cuantificador 40011 de coeficientes) según las realizaciones cuantifica los valores de atributos predichos. Además, el codificador de vídeo de la nube de puntos (por ejemplo, el codificador 40012 aritmético) realiza codificación por entropía en los valores de atributos cuantificados.

El codificador de vídeo de la nube de puntos (por ejemplo, el transformador 40008 RAHT) según las realizaciones puede realizar codificación por transformada RAHT en la que se predicen atributos de nodos de un nivel superior utilizando los atributos asociados con nodos de un nivel inferior en el árbol octal. La codificación por transformada de RAHT es un ejemplo de intracodificación de atributos a través de una exploración hacia atrás de árbol octal. El codificador de vídeo de la nube de puntos según las realizaciones explora toda la región desde el vóxel y repite el proceso de fusión de fusionar los vóxeles en un bloque más grande en cada etapa hasta que se alcanza el nodo raíz. El proceso de fusión según las realizaciones se realiza solo en los nodos ocupados. El proceso de fusión no se realiza en el nodo vacío. El proceso de fusión se realiza en un nodo superior inmediatamente por encima del nodo vacío.

La ecuación 3 siguiente representa una matriz de transformación RAHT. En la ecuación 3, $g_{l,x,y,z}$ indica el valor de atributo promedio de vóxeles a nivel l. $g_{l,x,y,z}$ puede calcularse basándose en $g_{l+1,2x,y,z}$ y $g_{l+1,2x+1,y,z}$. Los pesos para $g_{l+1,2x,y,z}$ y $g_{l+1,2x+1,y,z}$ are $w1 = w_{l,2x,y,z}$ y $w2 = w_{l,2x+1,y,z}$

Ecuación 3

$$\begin{bmatrix} g_{l-1,x,y,z} \\ h_{l-1,x,y,z} \end{bmatrix} = T_{w1 w2} \begin{bmatrix} g_{l,2x,y,z} \\ g_{l,2x+1,y,z} \end{bmatrix} \quad T_{w1 w2} = \frac{1}{\sqrt{w1+w2}} \begin{bmatrix} \sqrt{w1} & \sqrt{w2} \\ -\sqrt{w2} & \sqrt{w1} \end{bmatrix}$$

Aquí, $g_{l-1,x,y,z}$ es un valor de paso bajo y se utiliza en el proceso de fusión en el siguiente nivel superior.

$h_{l-1,x,y,z}$ indica coeficientes de paso alto. Los coeficientes de paso alto en cada etapa se cuantifican y se someten a codificación por entropía (por ejemplo, codificación por el codificador 40012 aritmético). Los pesos se

calculan como $w_{l-1,x,y,z} = w_{l2x,y,z} + w_{l2x+1,y,z}$. El nodo raíz se crea a través del nodo raíz $g_{10,0,0}$ y $g_{10,0,1}$ como ecuación 4.

Ecuación 4

$$\begin{bmatrix} gDC \\ h_{0,0,0} \end{bmatrix} = T_{w1000 w1001} \begin{bmatrix} g_{10,0,0} \\ g_{10,0,1} \end{bmatrix}$$

5 El valor de gDC también se cuantifica y se somete a codificación por entropía como los coeficientes de paso alto.

La fig. 10 ilustra un decodificador de vídeo de la nube de puntos según las realizaciones.

10 El decodificador de vídeo de la nube de puntos ilustrado en la fig. 10 es un ejemplo del decodificador 10006 de vídeo de la nube de puntos descrito en la fig. 1, y puede realizar las mismas o similares operaciones que las operaciones del decodificador 10006 de vídeo de la nube de puntos ilustrado en la fig. 1. Como se muestra en la figura, el decodificador de vídeo de la nube de puntos puede recibir un flujo de bits de geometría y un flujo de bits de atributos contenidos en uno o más flujos de bits. El decodificador de vídeo de la nube de puntos incluye un decodificador de geometría y un decodificador de atributo. El decodificador de geometría realiza decodificación de geometría en el flujo de bits de geometría y emite geometría decodificada. El decodificador de atributo realiza una decodificación de atributos en el flujo de bits de atributos basándose en la geometría decodificada, y envía atributos decodificados. La geometría decodificada y los atributos decodificados se utilizan para reconstruir el contenido de la nube de puntos (una nube de puntos decodificada).

La fig. 11 ilustra un decodificador de vídeo de la nube de puntos según las realizaciones.

20 El decodificador de vídeo de la nube de puntos ilustrado en la fig. 11 es un ejemplo del decodificador de vídeo de la nube de puntos ilustrado en la fig. 10, y puede realizar una operación de decodificación, que es un proceso inverso de la operación de codificación del codificador de vídeo de la nube de puntos ilustrado en las figs. 1 a 9.

Como se describe con referencia a las figs. 1 y 10, el decodificador de vídeo de la nube de puntos puede realizar decodificación de geometría y decodificación de atributos. La decodificación de geometría se realiza antes de la decodificación de atributo.

25 El decodificador de vídeo de la nube de puntos según las realizaciones incluye un decodificador aritmético (decodificador aritmético) 11000, un sintetizador de árbol octal (sintetizar árbol octal) 11001, un sintetizador de aproximación de superficie (sintetizar aproximación de superficie) 11002 y un reconstructor de geometría (reconstruir geometría) 11003, un transformador inverso de coordenadas (coordenadas de transformada inversa) 11004, un decodificador aritmético (decodificador aritmético) 11005, un cuantificador inverso (cuantificador inverso) 11006, un transformador RAHT 11007, un generador LOD (LOD generado) 11008, un elevador inverso (elevación inversa) 11009 y/o un transformador inverso de color (colores de transformada inversa) 11010.

35 El decodificador 11000 aritmético, el sintetizador 11001 de árbol octal, el sintetizador 11002 de aproximación de superficie y el reconstructor 11003 de geometría, y el transformador 11004 inverso de coordenadas pueden realizar decodificación de geometría. La decodificación de geometría según las realizaciones puede incluir decodificación directa y decodificación de geometría trisoup. La decodificación directa y la decodificación de geometría trisoup se aplican selectivamente. La decodificación de geometría no se limita al ejemplo descrito anteriormente, y se realiza como un proceso inverso de la codificación de geometría descrita con referencia a las figs. 1 a 9.

40 El decodificador 11000 aritmético según las realizaciones decodifica el flujo de bits de geometría recibido basándose en la codificación aritmética. La operación del decodificador 11000 aritmético corresponde al proceso inverso del codificador 40004 aritmético.

El sintetizador 11001 de árbol octal según las realizaciones puede generar un árbol octal adquiriendo un código de ocupación a partir del flujo de bits de geometría decodificado (o información sobre la geometría asegurada como resultado de la decodificación). El código de ocupación está configurado como se describe en detalle con referencia a las figs. 1 a 9.

45 Cuando se aplica la codificación de geometría trisoup, el sintetizador 11002 de aproximación de superficie según las realizaciones puede sintetizar una superficie basándose en la geometría decodificada y/o el árbol octal generado.

50 El reconstructor 11003 de geometría según las realizaciones puede regenerar geometría basándose en la superficie y/o la geometría decodificada. Como se describe con referencia a las figs. 1 a 9, se aplican selectivamente la codificación directa y la codificación de geometría triple. Por consiguiente, el reconstructor 11003 de geometría importa directamente y añade información de posición sobre los puntos a los que se aplica

codificación directa. Cuando se aplica la codificación de geometría trisoup, el reconstructor 11003 de geometría puede reconstruir la geometría realizando las operaciones de reconstrucción del reconstructor 40005 de geometría, por ejemplo, reconstrucción triangular, muestreo ascendente y voxelización. Los detalles son los mismos que los descritos con referencia a la fig. 6, y por tanto se omite su descripción. La geometría reconstruida puede incluir una imagen o trama de la nube de puntos que no contiene atributos.

El transformador 11004 inverso de coordenadas según las realizaciones puede adquirir posiciones de los puntos transformando las coordenadas basándose en la geometría reconstruida.

El decodificador 11005 aritmético, el cuantificador 11006 inverso, el transformador 11007 RAHT, el generador 11008 de LOD, el elevador 11009 inverso y/o el transformador 11010 inverso de color pueden realizar la decodificación de atributos descrita con referencia a la fig. 10. La decodificación de atributos según las realizaciones incluye decodificación por transformada jerárquica adaptativa de región (RAHT), decodificación por predicción jerárquica de contiguos más cercanos basada en interpolación (transformada de predicción) y predicción jerárquica de contiguos más cercanos basada en interpolación con una decodificación de etapa de actualización/elevación (transformada de elevación). Los tres esquemas de decodificación descritos anteriormente pueden utilizarse selectivamente, o puede utilizarse una combinación de uno o más esquemas de decodificación. La decodificación de atributos según las realizaciones no se limita al ejemplo descrito anteriormente.

El decodificador 11005 aritmético según las realizaciones decodifica el flujo de bits de atributos mediante codificación aritmética.

El cuantificador 11006 inverso según las realizaciones cuantifica inversamente la información sobre el flujo de bits de atributos decodificados o atributos asegurados como resultado de la decodificación, y emite los atributos cuantificados inversamente (o valores de atributos). La cuantificación inversa puede aplicarse selectivamente basándose en la codificación de atributos del codificador de vídeo de la nube de puntos.

Según las realizaciones, el transformador 11007 RAHT, el generador 11008 de LOD y/o el elevador 11009 inverso pueden procesar la geometría reconstruida y los atributos cuantificados inversamente. Como se ha descrito anteriormente, el transformador 11007 RAHT, el generador 11008 de LOD y/o el elevador 11009 inverso pueden realizar selectivamente una operación de decodificación correspondiente a la codificación del codificador de vídeo de la nube de puntos.

El transformador 11010 inverso de color según las realizaciones realiza codificación por transformada inversa para transformar inversamente un valor de color (o textura) incluido en los atributos decodificados. La operación del transformador 11010 inverso de color puede realizarse selectivamente basándose en la operación del transformador 40006 de color del codificador de vídeo de la nube de puntos.

Aunque no se muestra en la figura, los elementos del decodificador de vídeo de la nube de puntos de la fig. 11 pueden implementarse mediante hardware que incluye uno o más procesadores o circuitos integrados configurados para comunicarse con una o más memorias incluidas en el aparato de suministro de contenido de la nube de puntos, software, firmware o una combinación de los mismos. El uno o más procesadores pueden realizar al menos una o más de las operaciones y/o funciones de los elementos del decodificador de vídeo de la nube de puntos de la fig. 11 descrito anteriormente. Adicionalmente, el uno o más procesadores pueden operar o ejecutar un conjunto de programas de software y/o instrucciones para realizar las operaciones y/o funciones de los elementos del decodificador de vídeo de la nube de puntos de la fig. 11.

La fig. 12 ilustra un dispositivo de transmisión según las realizaciones.

El dispositivo de transmisión mostrado en la fig. 12 es un ejemplo del dispositivo 10000 de transmisión de la fig. 1 (o el codificador de vídeo de la nube de puntos de la fig. 4). El dispositivo de transmisión ilustrado en la fig. 12 puede realizar una o más de las operaciones y métodos iguales o similares a los del codificador de vídeo de la nube de puntos descrito con referencia a las figs. 1 a 9. El dispositivo de transmisión según las realizaciones puede incluir una unidad 12000 de entrada de datos, un procesador 12001 de cuantificación, un procesador 12002 de voxelización, un generador 12003 de código de ocupación de árbol octal, un procesador 12004 de modelo de superficie, un procesador 12005 de intra/intercodificación, un codificador 12006 aritmético, un procesador 12007 de metadatos, un procesador 12008 de transformada de color, un procesador 12009 de transformada de atributos, un procesador 12010 de transformada de predicción/elevación/RAHT, un codificador 12011 aritmético y/o un procesador 12012 de transmisión.

La unidad 12000 de entrada de datos según las realizaciones recibe o adquiere datos de la nube de puntos. La unidad 12000 de entrada de datos puede realizar un método de operación y/o adquisición igual o similar al método de operación y/o adquisición de la unidad 10001 de adquisición de vídeo de la nube de puntos (o el proceso 20000 de adquisición descrito con referencia a la fig. 2).

- La unidad 12000 de entrada de datos, el procesador 12001 de cuantificación, el procesador 12002 de voxelización, el generador 12003 de código de ocupación de árbol octal, el procesador 12004 de modelo de superficie, el procesador 12005 de intra/intercodificación y el codificador 12006 aritmético realizan codificación de geometría. La codificación de geometría según las realizaciones es la misma o similar a la codificación de geometría descrita con referencia a las figs. 1 a 9, y por lo tanto se omite una descripción detallada de la misma.
- El procesador 12001 de cuantificación según las realizaciones cuantifica la geometría (por ejemplo, valores de posición de puntos). La operación y/o cuantificación del procesador 12001 de cuantificación es la misma o similar a la operación y/o cuantificación del cuantificador 40001 descrito con referencia a la fig. 4. Los detalles son los mismos que los descritos con referencia a las figs. 1 a 9.
- El procesador 12002 de voxelización según las realizaciones voxeliza los valores de posición cuantificados de los puntos. El procesador 12002 de voxelización puede realizar una operación y/o procesar la misma o similar a la operación y/o el proceso de voxelización del cuantificador 40001 descrito con referencia a la fig. 4. Los detalles son los mismos que los descritos con referencia a las figs. 1 a 9.
- El generador 12003 de código de ocupación de árbol octal según las realizaciones realiza codificación de árbol octal en las posiciones voxelizadas de los puntos basándose en una estructura de árbol octal. El generador 12003 de código de ocupación de árbol octal puede generar un código de ocupación. El generador 12003 de código de ocupación de árbol octal puede realizar una operación y/o método igual o similar a la operación y/o método del codificador de vídeo de la nube de puntos (o el analizador 40002 de árbol octal) descrito con referencia a las figs. 4 y 6. Los detalles son los mismos que los descritos con referencia a las figs. 1 a 9.
- El procesador 12004 de modelo de superficie según las realizaciones puede realizar codificación de geometría de trisoup basándose en un modelo de superficie para reconstruir las posiciones de puntos en una región específica (o nodo) en una base de vóxel. El procesador 12004 de modelo de superficie puede realizar una operación y/o método igual o similar a la operación y/o método del codificador de vídeo en nube de puntos (por ejemplo, el analizador 40003 de aproximación de superficie) descrito con referencia a la fig. 4. Los detalles son los mismos que los descritos con referencia a las figs. 1 a 9.
- El procesador 12005 de intra/intercodificación según las realizaciones puede realizar la intra/intercodificación en datos de la nube de puntos. El procesador 12005 de intra/intercodificación puede realizar una codificación igual o similar a la intra/intercodificación descrita con referencia a la fig. 7. Los detalles son los mismos que los descritos con referencia a la fig. 7. Según las realizaciones, el procesador 12005 de intra/intercodificación puede incluirse en el codificador 12006 aritmético.
- El codificador 12006 aritmético según las realizaciones realiza codificación por entropía en un árbol octal de los datos de la nube de puntos y/o un árbol octal aproximado. Por ejemplo, el esquema de codificación incluye codificación aritmética. El codificador 12006 aritmético realiza una operación y/o método igual o similar a la operación y/o método del codificador 40004 aritmético.
- El procesador 12007 de metadatos, según las realizaciones, procesa metadatos sobre los datos de la nube de puntos, por ejemplo, un valor establecido, y los proporciona a un proceso de tratamiento necesario, tal como la codificación de geometría y/o la codificación de atributos. Además, el procesador 12007 de metadatos según las realizaciones puede generar y/o procesar información de señalización relacionada con la codificación de geometría y/o la codificación de atributos. La información de señalización según las realizaciones puede codificarse por separado de la codificación de geometría y/o la codificación de atributos. La información de señalización según las realizaciones puede intercalarse.
- El procesador 12008 de transformada de color, el procesador 12009 de transformada de atributos, el procesador 12010 de transformada de predicción/elevación/RAHT y el codificador 12011 aritmético realizan la codificación de atributos. La codificación de atributos según las realizaciones es la misma o similar a la codificación de atributos descrita con referencia a las figs. 1 a 9, y así se omite una descripción detallada de la misma.
- El procesador 12008 de transformada de color según las realizaciones realiza la codificación por transformada de color para transformar los valores de color incluidos en atributos. El procesador 12008 de transformada de color puede realizar la codificación por transformada de color basándose en la geometría reconstruida. La geometría reconstruida es la misma que la descrita con referencia a las figs. 1 a 9. Además, realiza una operación y/o método igual o similar a la operación y/o método del transformador 40006 de color descrito con referencia a la fig. 4. Se omite la descripción detallada de la misma.
- El procesador 12009 de transformada de atributos según las realizaciones realiza la transformación de atributos para transformar los atributos basándose en la geometría reconstruida y/o las posiciones en las que no se realiza la codificación de geometría. El procesador 12009 de transformada de atributos realiza una operación y/o método igual o similar a la operación y/o método del transformador 40007 de atributos descrito con referencia a la fig. 4. Se omite la descripción detallada de la misma. El procesador 12010 de transformada de predicción/elevación/RAHT según las realizaciones puede codificar los atributos transformados mediante una

cualquiera o una combinación de codificación RAHT, codificación por transformada de predicción y codificación por transformada de elevación. El procesador 12010 de transformada de predicción/elevación/RAHT realiza al menos una de las operaciones igual o similar a las operaciones del transformador 40008 de RAHT, el generador 40009 de LOD y el transformador 40010 de elevación descritos con referencia a la fig. 4. Además, la codificación por transformada de predicción, la codificación por transformada de elevación y la codificación por transformada de RAHT son las mismas que las descritas con referencia a las figs. 1 a 9, y por tanto se omite una descripción detallada de las mismas.

El codificador 12011 aritmético según las realizaciones puede codificar los atributos codificados basándose en la codificación aritmética. El codificador 12011 aritmético realiza una operación y/o método igual o similar a la operación y/o método del codificador 40012 aritmético.

El procesador 12012 de transmisión según las realizaciones puede transmitir cada flujo de bits que contiene geometría codificada y/o atributos y metadatos codificados, o transmitir un flujo de bits configurado con la geometría codificada y/o los atributos y los metadatos codificados. Cuando la geometría codificada y/o los atributos y los metadatos codificados según las realizaciones se configuran en un flujo de bits, el flujo de bits puede incluir uno o más subflujos de bits. El flujo de bits según las realizaciones puede contener información de señalización que incluye un conjunto de parámetros de secuencia (SPS) para la señalización de un nivel de secuencia, un conjunto de parámetros de geometría (GPS) para la señalización de la codificación de información de geometría, un conjunto de parámetros de atributo (APS) para la señalización de la codificación de información de atributo, y un conjunto de parámetros de mosaico (TPS o inventario de mosaicos) para la señalización de un nivel de mosaicos, y datos de segmento. Los datos de segmento pueden incluir información sobre uno o más segmentos. Un segmento según las realizaciones puede incluir un flujo de bits de geometría $Geom0^0$ y uno o más flujos de bits de atributos $Attr0^0$ y $Attr10$.

El segmento es una serie de un elemento sintáctico que representa en su totalidad o en parte la trama de la nube de puntos codificada.

El TPS según las realizaciones puede incluir información sobre cada mosaico (por ejemplo, información de coordenadas e información de altura/tamaño sobre una zona de delimitación) para uno o más mosaicos. El flujo de bits de geometría puede contener una cabecera y una carga útil. La cabecera del flujo de bits de geometría según las realizaciones puede contener un identificador de conjunto de parámetros ($geom_parameter_set_id$), un identificador de mosaico ($geom_tile_id$) y un identificador de segmento ($geom_slice_id$) incluidos en el GPS, e información sobre los datos contenidos en la carga útil. Como se ha descrito anteriormente, el procesador 12007 de metadatos según las realizaciones puede generar y/o procesar la información de señalización y transmitir la misma al procesador 12012 de transmisión. Según las realizaciones, los elementos para realizar la codificación de geometría y los elementos para realizar la codificación de atributos pueden compartir datos/información entre sí como se indica por líneas de puntos. El procesador 12012 de transmisión según las realizaciones puede realizar una operación y/o un método de transmisión igual o similar a la operación y/o el método de transmisión del transmisor 10003. Los detalles son los mismos que los descritos con referencia a las figs. 1 y 2, y por tanto se omite una descripción de los mismos.

La fig. 13 ilustra un dispositivo de recepción según las realizaciones.

El dispositivo de recepción ilustrado en la fig. 13 es un ejemplo del dispositivo 10004 de recepción de la fig. 1 (o el decodificador de vídeo de la nube de puntos de las figs. 10 y 11). El dispositivo de recepción ilustrado en la fig. 13 puede realizar una o más de las operaciones y métodos iguales o similares a los del decodificador de vídeo de la nube de puntos descrito con referencia a las figs. 1 a 11.

El dispositivo de recepción según la realización incluye un receptor 13000, un procesador 13001 de recepción, un decodificador 13002 aritmético, un procesador 13003 de reconstrucción de árbol octal basado en código de ocupación, un procesador 13004 de modelo de superficie (reconstrucción triangular, muestreo ascendente, voxelización), un procesador 13005 de cuantificación inversa, un analizador 13006 sintáctico de metadatos, un decodificador 13007 aritmético, un procesador 13008 de cuantificación inversa, un procesador 13009 de transformada inversa de predicción/elevación/RAHT, un procesador 13010 de transformada inversa de color y/o un renderizador 13011. Cada elemento para decodificar según las realizaciones puede realizar un proceso inverso de la operación de un elemento correspondiente para codificar según las realizaciones.

El receptor 13000 según las realizaciones recibe datos de la nube de puntos. El receptor 13000 puede realizar un método de operación y/o recepción igual o similar al método de operación y/o recepción del receptor 10005 de la fig. 1. Se omite la descripción detallada de la misma.

El procesador 13001 de recepción según las realizaciones puede adquirir un flujo de bits de geometría y/o un flujo de bits de atributos a partir de los datos recibidos. El procesador 13001 de recepción puede estar incluido en el receptor 13000.

El decodificador 13002 aritmético, el procesador 13003 de reconstrucción de árbol octal basado en código de ocupación, el procesador 13004 de modelo de superficie y el procesador 1305 de cuantificación inversa pueden realizar decodificación de geometría. La decodificación geométrica según las realizaciones es la misma o similar a la decodificación de geometría descrita con referencia a las figs. 1 a 10, y por lo tanto se omite una descripción detallada de la misma.

El decodificador 13002 aritmético según las realizaciones puede decodificar el flujo de bits de geometría basándose en la codificación aritmética. El decodificador 13002 aritmético realiza una operación y/o codificación igual o similar a la operación y/o codificación del decodificador 11000 aritmético.

El procesador 13003 de reconstrucción de árbol octal basado en código de ocupación según las realizaciones puede reconstruir un árbol octal adquiriendo un código de ocupación a partir del flujo de bits de geometría decodificado (o información sobre la geometría asegurada como resultado de la decodificación). El procesador 13003 de reconstrucción de árbol octal basado en código de ocupación realiza una operación y/o método igual o similar a la operación y/o método generación de árbol octal del sintetizador 11001 de árbol octal. Cuando se aplica la codificación de geometría trisoup, el procesador 1302 de modelo de superficie según las realizaciones puede realizar una decodificación de geometría trisoup y una reconstrucción de geometría relacionada (por ejemplo, reconstrucción triangular, muestreo ascendente, voxelización) basándose en el método de modelo de superficie. El procesador 1302 de modelo de superficie realiza una operación igual o similar a la del sintetizador 11002 de aproximación de superficie y/o el reconstructor 11003 de geometría.

El procesador 1305 de cuantificación inversa según las realizaciones puede cuantificar de manera inversa la geometría decodificada.

El analizador 1306 sintáctico de metadatos según las realizaciones puede analizar los metadatos contenidos en los datos de la nube de puntos recibidos, por ejemplo, un valor establecido. El analizador 1306 sintáctico de metadatos puede pasar los metadatos a decodificación de geometría y/o decodificación de atributos. Los metadatos son los mismos que los descritos con referencia a la fig. 12, y por lo tanto se omite una descripción detallada de los mismos.

El decodificador 13007 aritmético, el procesador 13008 de cuantificación inversa, el procesador 13009 de transformada inversa de predicción/elevación/RAHT y el procesador 13010 de transformada inversa de color realizan decodificación de atributos. La decodificación de atributos es la misma o similar a la decodificación de atributos descrita con referencia a las figs. 1 a 10, y por lo tanto se omite una descripción detallada de la misma.

El decodificador 13007 aritmético según las realizaciones puede decodificar el flujo de bits de atributos mediante codificación aritmética. El decodificador 13007 aritmético puede decodificar el flujo de bits de atributos basándose en la geometría reconstruida. El decodificador 13007 aritmético realiza una operación y/o codificación igual o similar a la operación y/o codificación del decodificador 11005 aritmético.

El procesador 13008 de cuantificación inversa según las realizaciones puede cuantificar de manera inversa el flujo de bits de atributos decodificados. El procesador 13008 de cuantificación inversa realiza una operación y/o un método iguales o similares a la operación y/o al método de cuantificación inversa del cuantificador 11006 inverso.

El transformador 13009 inverso de predicción/elevación/RAHT según las realizaciones puede procesar la geometría reconstruida y los atributos cuantificados inversamente. El procesador 1301 de transformada inversa de predicción/elevación/RAHT realiza una o más de las operaciones y/o decodificación de las mismas o similares a las operaciones y/o decodificación del transformador 11007 de RAHT, el generador 11008 de LOD y/o el elevador 11009 inverso. El procesador 13010 de transformada inversa de color según las realizaciones realiza la codificación por transformada inversa para transformar de manera inversa los valores de color (o texturas) incluidos en los atributos decodificados. El procesador 13010 de transformada inversa de color realiza una operación y/o codificación por transformada inversa igual o similar a la operación y/o codificación por transformada inversa del transformador 11010 inverso de color. El renderizador 13011 según las realizaciones puede renderizar los datos de la nube de puntos.

La fig. 14 muestra una estructura ejemplar conectable operativamente con un método/dispositivo para transmitir y recibir datos de la nube de puntos según las realizaciones.

La estructura de la fig. 14 representa una configuración en la que al menos uno de un servidor 17600, un robot 17100, un vehículo 17200 de autónomo, un dispositivo 17300 de XR, un teléfono inteligente 17400, un electrodoméstico 17500 y/o un dispositivo 17700 de presentación de montaje en la cabeza (HMD) está conectado a una red 17100 de la nube. El robot 17100, el vehículo 17200 autónomo, el dispositivo 17300 de XR, el teléfono inteligente 17400 o el electrodoméstico 17500 se denominan dispositivo. Además, el dispositivo 17300 de XR puede corresponder a un dispositivo de datos comprimidos de la nube de puntos (PCC) según las realizaciones o puede estar conectado operativamente al dispositivo de PCC.

La red 17000 de la nube puede representar una red que constituye parte de la infraestructura informática de la nube o está presente en la infraestructura informática de la nube. En este caso, la red 17000 de la nube puede configurarse utilizando una red 3G, una red 4G o una red de evolución a largo plazo (LTE), o una red 5G.

El servidor 17600 puede estar conectado a al menos uno del robot 17100, el vehículo 17200 autónomo, el dispositivo 17300 de XR, el teléfono inteligente 17400, el electrodoméstico 17500 y/o el HMD 17700 a través de la red 17000 de la nube y puede ayudar en al menos una parte del tratamiento de los dispositivos 17100 a 17700 conectados.

El HMD 17700 representa uno de los tipos de implementación del dispositivo de XR y/o el dispositivo de PCC según las realizaciones. El dispositivo de tipo HMD según las realizaciones incluye una unidad de comunicación, una unidad de control, una memoria, una unidad de E/S, una unidad de sensor y una unidad de suministro de energía.

En lo sucesivo, se describirán diversas realizaciones de los dispositivos 17100 a 17500 a los que se aplica la tecnología descrita anteriormente. Los dispositivos 17100 a 17500 ilustrados en la fig. 14 pueden estar conectados/acoplados operativamente a un dispositivo de transmisión de datos de la nube de puntos y recepción según las realizaciones descritas anteriormente.

<PCC+XR>

El dispositivo 17300 de XR/PCC puede emplear tecnología de PCC y/o tecnología de XR (AR+VR), y puede implementarse como un HMD, un dispositivo de presentación de cabeza erguida (HUD) provisto en un vehículo, una televisión, un teléfono móvil, un teléfono inteligente, un ordenador, un dispositivo que se puede llevar puesto, un electrodoméstico, una señalización digital, un vehículo, un robot estacionario o un robot móvil.

El dispositivo 17300 de XR/PCC puede analizar datos de la nube de puntos 3D o datos de imagen adquiridos a través de diversos sensores o desde un dispositivo externo y generar datos de posición y datos de atributos sobre puntos 3D. De este modo, el dispositivo 17300 de XR/PCC puede adquirir información sobre el espacio circundante o un objeto real, y renderizar y emitir un objeto de XR. Por ejemplo, el dispositivo 17300 de XR/PCC puede hacer coincidir un objeto de XR que incluye información auxiliar sobre un objeto reconocido con el objeto reconocido y emitir el objeto de XR coincidente.

<PCC+ Autónomo+XR>

El vehículo 17200 autónomo puede implementarse como un robot móvil, un vehículo, un vehículo aéreo no tripulado o similares aplicando la tecnología de PCC y la tecnología de XR.

El vehículo 17200 autónomo al que se aplica la tecnología de XR/PCC puede representar un vehículo autónomo provisto de medios para proporcionar una imagen de XR, o un vehículo autónomo que es un objetivo de control/interacción en la imagen de XR. En particular, el vehículo 17200 autónomo que es un objetivo de control/interacción en la imagen de XR puede distinguirse del dispositivo XR de 17300 y puede estar conectado operativamente al mismo.

El vehículo 17200 autónomo que tiene medios para proporcionar una imagen de XR/PCC puede adquirir información de sensor de sensores que incluyen una cámara, y emitir la imagen de XR/PCC generada basándose en la información de sensor adquirida. Por ejemplo, el vehículo 17200 autónomo puede tener un HUD y emitir una imagen de XR/PCC al mismo, proporcionando de este modo a un ocupante un objeto de XR/PCC correspondiente a un objeto real o un objeto presente en la pantalla.

Cuando el objeto XR/PCC se emite al HUD, al menos una parte del objeto de XR/PCC puede enviarse para superponerse al objeto real al que se dirigen los ojos del ocupante. Por otro lado, cuando el objeto de XR/PCC se emite en una pantalla proporcionada dentro del vehículo autónomo, al menos una parte del objeto de XR/PCC puede emitirse para superponerse a un objeto en la pantalla. Por ejemplo, el vehículo 17200 autónomo puede emitir objetos de XR/PCC correspondientes a objetos tales como una carretera, otro vehículo, un semáforo, una señal de tráfico, un vehículo de dos ruedas, un peatón y un edificio.

La tecnología de realidad virtual (VR), la tecnología de realidad aumentada (AR), la tecnología de realidad mixta (MR) y/o la tecnología de compresión de la nube de puntos (PCC) según las realizaciones son aplicables a diversos dispositivos.

En otras palabras, la tecnología de VR es una tecnología de visualización que proporciona solo imágenes de CG de objetos del mundo real, fondos y similares. Por otro lado, la tecnología de AR se refiere a una tecnología que muestra una imagen de CG creada virtualmente en la imagen de un objeto real. La tecnología de MR es similar a la tecnología de AR descrita anteriormente en que los objetos virtuales que han de mostrarse se mezclan y combinan con el mundo real. Sin embargo, la tecnología de MR difiere de la tecnología de AR en que la tecnología de AR hace una distinción clara entre un objeto real y un objeto virtual creado como una imagen de CG y utiliza objetos virtuales como objetos complementarios para objetos reales, mientras que la tecnología de

MR trata objetos virtuales como objetos que tienen características equivalentes como objetos reales. Más específicamente, un ejemplo de aplicaciones de tecnología de MR es un servicio de holograma.

Recientemente, las tecnologías de VR, AR y MR a veces se denominan tecnología de realidad extendida (XR) en lugar de distinguirse claramente entre sí. Por consiguiente, las realizaciones de la presente descripción son aplicables a cualquiera de las tecnologías de VR, AR, MR y XR. La codificación/decodificación basada en técnicas de PCC, V-PCC y G-PCC es aplicable a tales tecnologías.

El método/dispositivo de PCC según las realizaciones puede aplicarse a un vehículo que proporciona un servicio de conducción autónoma.

Un vehículo que proporciona el servicio de conducción autónoma está conectado a un dispositivo de PCC para comunicación por cable/inalámbrica.

Cuando el dispositivo de transmisión/recepción de datos de compresión de la nube de puntos (PCC) según las realizaciones está conectado a un vehículo para comunicación por cable/inalámbrica, el dispositivo puede recibir/procesar datos de contenido relacionados con un servicio de AR/VR/PCC, que pueden proporcionarse junto con el servicio de conducción autónoma, y transmitir los mismos al vehículo. En el caso donde el dispositivo de transmisión/recepción de PCC está montado en un vehículo, el dispositivo de transmisión/recepción de PCC puede recibir/procesar datos de contenido relacionados con el servicio de AR/VR/PCC según una entrada de señal de entrada de usuario a través de un dispositivo de interfaz de usuario y proporcionar los mismos al usuario. El vehículo o el dispositivo de interfaz de usuario según las realizaciones pueden recibir una señal de entrada de usuario. La señal de entrada de usuario según las realizaciones puede incluir una señal que indica el servicio de conducción autónoma.

Mientras tanto, el codificador de vídeo de la nube de puntos en el lado de transmisión puede realizar además un proceso de división en particiones espacial dividiendo en particiones de manera espacial (o dividiendo) los datos de la nube de puntos en uno o más bloques 3D antes de codificar los datos de la nube de puntos. Es decir, para que las operaciones de codificación y transmisión del dispositivo de transmisión y las operaciones de decodificación y renderización del dispositivo de recepción se realicen en tiempo real y se procesen con baja latencia, el dispositivo de transmisión puede dividir en particiones espacialmente los datos de la nube de puntos en una pluralidad de regiones. Además, el dispositivo de transmisión puede codificar de manera independiente o no independiente las regiones (o bloques) espacialmente divididas en particiones, permitiendo así el acceso aleatorio y la codificación paralela en el espacio tridimensional ocupado por los datos de la nube de puntos. Además, el dispositivo de transmisión y el dispositivo de recepción pueden realizar la codificación y la decodificación de manera independiente o no independiente para cada región (o bloque) espacialmente dividida en particiones, impidiendo de este modo que se acumulen errores en el proceso de codificación y decodificación.

La fig. 15 es un diagrama que ilustra otro ejemplo de un dispositivo de transmisión de la nube de puntos según las realizaciones, que incluye un divisor en particiones espacial.

El dispositivo de transmisión de la nube de puntos según las realizaciones puede incluir una unidad 51001 de entrada de datos, una unidad 51002 de transformación de coordenadas, un procesador 51003 de cuantificación, un divisor 51004 en particiones espacial, un procesador 51005 de señalización, un codificador 51006 de geometría, un codificador 51007 de atributos y un procesador 51008 de transmisión. Según las realizaciones, la unidad 51002 de transformación de coordenadas, el procesador 51003 de cuantificación, el divisor 51004 en particiones espacial, el codificador 51006 de geometría y el codificador 51007 de atributos pueden denominarse codificadores de vídeo de la nube de puntos.

La unidad 51001 de entrada de datos puede realizar algunas o todas las operaciones de la unidad 10001 de adquisición de vídeo de la nube de puntos de la fig. 1, o puede realizar algunas o todas las operaciones de la unidad 12000 de entrada de datos de la fig. 12. La unidad 51002 de transformación de coordenadas puede realizar algunas o todas las operaciones de la unidad 40000 de transformación de coordenadas de la fig. 4. Además, el procesador 51003 de cuantificación puede realizar algunas o todas las operaciones de la unidad 40001 de cuantificación de la fig. 4, o puede realizar algunas o todas las operaciones del procesador 12001 de cuantificación de la fig. 12.

El divisor 51004 en particiones espacial puede dividir en particiones espacialmente los datos de la nube de puntos cuantificados y emitidos desde el procesador 51003 de cuantificación en uno o más bloques 3D basándose en una zona de delimitación y/o una zona de subdelimitación. En este caso, el bloque 3D puede referirse a un grupo de mosaicos, un mosaico, un segmento, una unidad de codificación (CU), una unidad de predicción (PU) o una unidad de transformación (TU). En una realización, la información de señalización para la división en particiones espacial es codificada por entropía por el procesador 51005 de señalización y, a continuación, transmitida a través del procesador 51008 de transmisión en forma de un flujo de bits.

(a) de las figs. 16 a (c) de la fig. 16 ilustran una realización de división en particiones de una zona de delimitación en uno o más mosaicos. Como se muestra en (a) de la fig. 16, un objeto de la nube de puntos, que corresponde

a datos de la nube de puntos, puede expresarse en forma de una caja basada en un sistema de coordenadas, que se denomina zona de delimitación. En otras palabras, la zona de delimitación representa un cubo capaz de contener todos los puntos de la nube de puntos.

5 (b) de la fig. 16 y (c) de la fig. 16 ilustran un ejemplo en el que la zona de delimitación de (a) de la fig. 16 se divide en el mosaico 1# y el mosaico 2#, y el mosaico 2# se divide de nuevo en el segmento 1# y el segmento 2#.

10 En una realización, el contenido de la nube de puntos puede ser una persona tal como un actor, múltiples personas, un objeto o múltiples objetos. En un intervalo mayor, puede ser un mapa para la conducción autónoma o un mapa para la navegación interior de un robot. En este caso, el contenido de la nube de puntos puede ser una gran cantidad de datos conectados localmente. En este caso, el contenido de la nube de puntos no puede codificarse/decodificarse a la vez, y por consiguiente, la división en particiones de mosaicos puede realizarse antes de que se comprima el contenido de la nube de puntos. Por ejemplo, la habitación #101 en un edificio puede dividirse en un mosaico y la habitación #102 en el edificio puede dividirse en otro mosaico. Para soportar codificación/decodificación rápida aplicando paralelización a los mosaicos divididos en particiones, los mosaicos pueden dividirse en particiones (o dividirse) en segmentos de nuevo. Esta operación puede denominarse división en particiones (o división) de segmentos.

20 Es decir, un mosaico puede representar una región parcial (por ejemplo, un cubo rectangular) de un espacio 3D ocupado por datos de la nube de puntos según las realizaciones. Según las realizaciones, un mosaico puede incluir uno o más segmentos. El mosaico según las realizaciones puede dividirse en uno o más segmentos y, por lo tanto, el codificador de vídeo de la nube de puntos puede codificar datos de la nube de puntos en paralelo.

25 Un segmento puede representar una unidad de datos (o flujo de bits) que puede codificarse independientemente por el codificador de vídeo de la nube de puntos según las realizaciones y/o una unidad de datos (o flujo de bits) que puede decodificarse independientemente por el decodificador de vídeo de la nube de puntos. Un segmento puede ser un conjunto de datos en un espacio 3D ocupado por datos de la nube de puntos, o un conjunto de algunos datos entre los datos de la nube de puntos. Un segmento según las realizaciones puede representar una región o conjunto de puntos incluidos en un mosaico según las realizaciones. Según las realizaciones, un mosaico puede dividirse en uno o más segmentos basándose en el número de puntos incluidos en un mosaico. Por ejemplo, un mosaico puede ser un conjunto de puntos divididos en particiones por el número de puntos. Según las realizaciones, un mosaico puede dividirse en uno o más segmentos basándose en el número de puntos, y algunos datos pueden dividirse o fusionarse en el proceso de división en particiones. Es decir, un segmento puede ser una unidad que puede codificarse independientemente dentro de un mosaico correspondiente. De esta manera, un mosaico obtenido mediante división en particiones espacial puede dividirse en uno o más segmentos para un tratamiento rápido y eficiente.

35 El codificador de vídeo de la nube de puntos según las realizaciones puede codificar datos de la nube de puntos en una base de segmento a segmento o en una base de mosaico a mosaico, en donde un mosaico incluye uno o más segmentos. Además, el codificador de vídeo de la nube de puntos según las realizaciones puede realizar diferente cuantificación y/o transformación para cada mosaico o cada segmento.

40 Las posiciones de uno o más bloques 3D (por ejemplo, segmentos) espacialmente divididos en particiones por el divisor 51004 en particiones espacial se emiten al codificador 51006 de geometría, y la información de atributo (o atributos) se emite al codificador 51007 de atributos. Las posiciones pueden ser información de posición sobre los puntos incluidos en una unidad dividida en particiones (caja, bloque, mosaico, grupo de mosaicos o segmento), y se denominan información de geometría.

45 El codificador 51006 de geometría construye y codifica (es decir, comprime) un árbol octal basándose en las posiciones emitidas desde el divisor 51004 en particiones espacial para emitir un flujo de bits de geometría. El codificador 51006 de geometría puede reconstruir un árbol octal y/o un árbol octal aproximado y emitir el mismo al codificador 51007 de atributos. El árbol octal reconstruido puede denominarse geometría reconstruida (o geometría restaurada).

50 El codificador 51007 de atributos codifica (es decir, comprime) los atributos emitidos desde el divisor 51004 en particiones espacial basándose en la geometría reconstruida emitida desde el codificador 51006 de geometría, y emite un flujo de bits de atributos.

La fig. 17 es un diagrama de bloques detallado que ilustra otro ejemplo del codificador 51006 de geometría y el codificador 51007 de atributos según las realizaciones.

55 El procesador 53001 de voxelización, el generador 53002 de árbol octal, el predictor 53003 de información de geometría y el codificador 53004 aritmético del codificador 51006 de geometría de la fig. 17 pueden realizar algunas o todas las operaciones de la unidad 40002 de análisis de árbol octal, la unidad 40003 de análisis de aproximación de superficie, el codificador 40004 aritmético y la unidad 40005 de reconstrucción de geometría de la fig. 4, o pueden realizar algunas o todas las operaciones del procesador 12002 de voxelización, el generador

12003 de código de ocupación de árbol octal y el procesador 12004 de modelo de superficie, el procesador 12005 de intra/intercodificación y el codificador 12006 aritmético de la fig. 12

5 El codificador 51007 de atributos de la fig. 17 puede incluir un procesador 53005 de transformación de color, un procesador 53006 de transformación de atributos, una unidad 53007 de configuración de LOD, una unidad 53008 de configuración de grupo de puntos contiguos, una unidad 53009 de predicción de información de atributo, un procesador 53010 de cuantificación de información de atributo residual y un codificador 53011 aritmético.

10 En una realización, se puede proporcionar además un procesador de cuantificación entre el divisor 51004 en particiones espacial y el procesador 53001 de voxelización. El procesador de cuantificación cuantifica las posiciones de uno o más bloques 3D (por ejemplo, segmentos) espacialmente divididos en particiones por el divisor 51004 en particiones espacial. En este caso, el procesador de cuantificación puede realizar algunas o todas las operaciones de la unidad 40001 de cuantificación de la fig. 4, o realizar algunas o todas las operaciones del procesador 12001 de cuantificación de la fig. 12. Cuando el procesador de cuantificación se proporciona además entre el divisor 51004 en particiones espacial y el procesador 53001 de voxelización, el procesador 51003 de cuantificación de la fig. 15 puede o no omitirse.

15 El procesador 53001 de voxelización según las realizaciones realiza la voxelización basándose en las posiciones de uno o más bloques 3D espacialmente divididos en particiones (por ejemplo, cortes) o las posiciones cuantificadas de los mismos. Voxelización se refiere a la unidad mínima que expresa información de posición en un espacio 3D. Los puntos de contenido de la nube de puntos (o vídeo de la nube de puntos 3D) según las realizaciones pueden incluirse en uno o más vóxeles. Según las realizaciones, un vóxel puede incluir uno o más puntos. En una realización, en el caso en el que la cuantificación se realiza antes de realizar la voxelización, una pluralidad de puntos puede pertenecer a un vóxel.

En la presente especificación, cuando se incluyen dos o más puntos en un vóxel, los dos o más puntos se denominan puntos duplicados. Es decir, en el proceso de codificación de geometría, se pueden generar puntos superpuestos a través de cuantificación de geometría y voxelización.

25 El procesador 53001 de voxelización según las realizaciones puede emitir los puntos duplicados que pertenecen a un vóxel al generador 53002 de árbol octal sin fusionar los mismos, o puede fusionar los múltiples puntos en un punto y emitir el punto fusionado al generador 53002 de árbol octal.

El generador 53002 de árbol octal según las realizaciones genera un árbol octal basado en una salida de vóxel desde el procesador 53001 de voxelización.

30 El predictor 53003 de información de geometría según las realizaciones predice y comprime información de geometría basándose en el árbol octal generado por el generador 53002 de árbol octal, y emite la información predicha y comprimida al codificador 53004 aritmético. Además, el predictor 53003 de información de geometría reconstruye la geometría basándose en las posiciones cambiadas a través de compresión, y emite la geometría reconstruida (o decodificada) a la unidad 53007 de configuración de LOD del codificador 51007 de atributos. La reconstrucción de la información de geometría puede realizarse en un dispositivo o componente separado del predictor 53003 de información de geometría. En otra realización, la geometría reconstruida también puede proporcionarse al procesador 53006 de transformación de atributos del codificador 51007 de atributos.

40 El procesador 53005 de transformación de color del codificador 51007 de atributos corresponde a la unidad 40006 de transformación de color de la fig. 4 o al procesador 12008 de transformación de color de la fig. 12. El procesador 53005 de transformación de color según las realizaciones realiza la codificación por transformación de color de los valores de color (o texturas) de transformación incluidos en los atributos proporcionados desde la unidad 51001 de entrada de datos y/o el divisor 51004 en particiones espacial. Por ejemplo, el procesador 53005 de transformación de color puede transformar el formato de la información de color (por ejemplo, de RGB a YCbCr). La operación del procesador 53005 de transformación de color según las realizaciones puede aplicarse opcionalmente según los valores de color incluidos en los atributos. En otra realización, el procesador 53005 de transformación de color puede realizar la codificación por transformación de color basándose en la geometría reconstruida. Para más detalles sobre la reconstrucción de geometría, véase la descripción de las figs. 1 a 9.

50 El procesador 53006 de transformación de atributos según las realizaciones puede realizar la transformación de atributos de atributos de transformación basándose en posiciones en las que no se ha realizado la codificación de geometría y/o la geometría reconstruida.

El procesador 53006 de transformación de atributos puede denominarse unidad de coloreado de nuevo.

55 La operación del procesador 53006 de transformación de atributos según las realizaciones puede aplicarse opcionalmente según si los puntos duplicados se fusionan. Según una realización, la fusión de los puntos duplicados puede realizarse por el procesador 53001 de voxelización o el generador 53002 de árbol octal del codificador 51006 de geometría.

En la presente especificación, cuando los puntos que pertenecen a un vóxel se fusionan en un punto en el procesador 53001 de voxelización o el generador 53002 de árbol octal, el procesador 53006 de transformación de atributos realiza una transformación de atributos. Tomemos un ejemplo.

- 5 El procesador 53006 de transformación de atributos realiza una operación y/o un método idéntico o similar a la operación y/o el método de la unidad 40007 de transformación de atributos de la fig. 4 o el procesador 12009 de transformación de atributos de la fig. 12.

Según las realizaciones, la información de geometría reconstruida por el predictor 53003 de información de geometría y la información de atributo emitida desde el procesador 53006 de transformación de atributos se proporcionan a la unidad 53007 de configuración de LOD para la compresión de atributos.

- 10 Según las realizaciones, la salida de información de atributo del procesador 53006 de transformación de atributos puede comprimirse mediante una o una combinación de dos o más de codificación RAHT, codificación por transformada predictiva basada en LOD y codificación por transformada de elevación basada en la información de geometría reconstruida.

- 15 En lo sucesivo, se supone que la compresión de atributos se realiza mediante una o una combinación de la codificación por transformada predictiva basada en LOD y la codificación por transformada de elevación como una realización. Por lo tanto, se omitirá una descripción de la codificación RAHT. Para detalles de la codificación por transformada RAHT, véase las descripciones de las figs. 1 a 9.

La unidad 53007 de configuración de LOD según las realizaciones genera un nivel de detalle (LOD).

- 20 El LOD es un grado que representa el detalle del contenido de la nube de puntos. A medida que disminuye el valor de LOD, indica que el detalle del contenido de la nube de puntos se degrada. A medida que aumenta el valor de LOD, indica que se mejora el detalle del contenido de la nube de puntos. Los puntos de la geometría reconstruida (es decir, las posiciones reconstruidas) pueden clasificarse según el LOD.

En una realización, en la codificación por transformada predictiva y la codificación por transformada de elevación, los puntos pueden dividirse en LOD y agruparse.

- 25 Esta operación puede denominarse proceso de generación de LOD, y un grupo que tiene diferentes LOD puede denominarse un conjunto LOD₁. Aquí, 1 indica el LOD y es un número entero que comienza desde 0. LOD₀ es un conjunto que consiste en puntos con la mayor distancia entre ellos. A medida que 1 aumenta, disminuye la distancia entre puntos que pertenecen a LOD₁.

- 30 Cuando el conjunto de LOD₁ se genera por la unidad 53007 de configuración de LOD, la unidad 53008 de configuración de puntos contiguos según la realización puede encontrar X (>0) puntos contiguos más cercanos (NN) en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto LOD₁ y registrar lo mismo en el predictor que un conjunto contiguo. X es el número máximo de puntos que pueden establecerse como contiguos. X puede introducirse como un parámetro de usuario, o señalizarse en información de señalización a través del procesador 51005 de señalización (por ejemplo, el campo lifting_num_pred_nearest_neighbours de elevación señalizado en el APS).

- 35 Con referencia a la fig. 9 como ejemplo, contiguos de P3 que pertenecen a LOD₁ se encuentran en LOD₀ y LOD₁. Por ejemplo, cuando el número máximo (X) de puntos que pueden establecerse como contiguos es 3, tres nodos contiguos más cercanos de P3 pueden ser P2 P4 P6. Estos tres nodos se registran como en el predictor de P3 como un conjunto contiguo. En una realización, entre los nodos registrados, el nodo contiguo P4 es el más cercano a P3 en términos de distancia, a continuación, P6 es el siguiente nodo más cercano, y P2 es el nodo más lejano entre los tres nodos. Aquí, X=3 es simplemente una realización configurada para proporcionar la comprensión de la presente descripción. El valor de X puede variar.

Como se ha descrito anteriormente, todos los puntos de los datos de la nube de puntos pueden tener un predictor, respectivamente.

- 45 La unidad 53009 de predicción de información de atributo según las realizaciones predice valores de atributos a partir de puntos contiguos registrados en un predictor y obtiene un valor de atributo residual de un punto correspondiente basándose en los valores de atributos predichos. El valor de atributo residual se emite al procesador 53010 de cuantificación de información de atributo residual.

A continuación, se describirán en detalle la generación de LOD y la búsqueda de puntos contiguos.

- 50 Como se ha descrito anteriormente, para la compresión de atributos, el codificador de atributo realiza un proceso de generación de LOD basándose en puntos de geometría reconstruida y buscando los puntos contiguos más cercanos de un punto que se ha de codificar basándose en los LOD generados. Según las realizaciones, incluso el decodificador de atributo del dispositivo de recepción realiza un proceso de generación de LOD y de búsqueda de los puntos contiguos más cercanos de un punto que se ha de decodificar basándose en los LOD generados.

Según las realizaciones, la unidad 53007 de configuración de LOD puede configurar uno o más LOD utilizando uno o más métodos de generación de LOD (o métodos de configuración de LOD).

Según las realizaciones, un método de generación de LOD utilizado en la unidad 53007 de configuración de LOD puede introducirse como parámetro de usuario o puede señalizarse en la información de señalización a través del procesador 51005 de señalización. Por ejemplo, el método de generación de LOD puede señalizarse en un APS.

Según las realizaciones, los métodos de generación de LOD pueden clasificarse en un método de generación de LOD basado en árbol octal, un método de generación de LOD basado en distancia y un método de generación de LOD basado en muestreo.

La fig. 18 es un diagrama que ilustra un ejemplo de generación de LOD basándose en un árbol octal según las realizaciones.

Según las realizaciones, cuando se generan LOD basándose en un árbol octal, cada nivel de profundidad del árbol octal puede coincidir con cada LOD como se ilustra en la fig. 18. Es decir, el método de generación de LOD basado en árbol octal es un método de generación de LOD utilizando características en las que un detalle que representa datos de la nube de puntos aumenta gradualmente a medida que aumenta un nivel de profundidad (es decir, una dirección desde una raíz hasta una hoja) en una estructura de árbol octal. Según las realizaciones, se puede realizar una configuración de LOD basada en árbol octal desde un nodo raíz a un nodo hoja o, a la inversa, desde el nodo hoja al nodo raíz.

El método de generación de LOD basado en distancia según las realizaciones es un método para disponer puntos utilizando un código de Morton y generar LOD basándose en una distancia entre los puntos.

El método de generación de LOD basado en muestreo según las realizaciones es un método para disponer puntos utilizando un código de Morton y clasificar cada k-ésimo punto como un LOD inferior que puede ser un candidato de un conjunto contiguo. Es decir, durante el muestreo, cada k-ésimo punto puede seleccionarse y clasificarse como un conjunto candidato inferior que pertenece a LOD₀ a LOD₁₋₁ y los puntos restantes pueden registrarse en un conjunto LOD₁. Es decir, los puntos seleccionados se convierten en un grupo candidato que puede seleccionarse como un conjunto de puntos contiguos de los puntos registrados en el conjunto LOD₁. k puede diferir según el contenido de la nube de puntos. Aquí, los puntos pueden ser puntos de una nube de puntos capturada o puntos de geometría reconstruida.

En la presente descripción, un conjunto candidato inferior correspondiente a LOD₀ a LOD₁₋₁ basado en el conjunto LOD₁ se denominará conjunto retenido o conjunto retenido de LOD. LOD₀ es un conjunto que consiste en puntos con la mayor distancia entre ellos. A medida que 1 aumenta, disminuye la distancia entre puntos que pertenecen a LOD₁.

Según las realizaciones, los puntos en el conjunto LOD₁ también están dispuestos basándose en una orden de código de Morton. Además, se incluyeron puntos en cada uno de los conjuntos LOD₀ a LOD₁₋₁ dispuestos también basándose en una orden de código de Morton.

La fig. 19 es un diagrama que ilustra un ejemplo de disposición de puntos de una nube de puntos en un orden de código de Morton según las realizaciones.

Es decir, se genera un código de Morton de cada punto basándose en los valores de posición x, y, y z de cada punto de una nube de puntos. Cuando se generan códigos de Morton de puntos de la nube de puntos a través de este proceso, los puntos de la nube de puntos pueden disponerse en un orden de código de Morton. Según las realizaciones, los puntos de la nube de puntos pueden disponerse en orden ascendente de los códigos de Morton. El orden de los puntos dispuestos en orden ascendente de los códigos de Morton puede denominarse orden de Morton.

La unidad 53007 de configuración de LOD según las realizaciones puede generar LOD realizando un muestreo en puntos dispuestos en orden de Morton.

La unidad 53007 de configuración de LOD genera el conjunto LOD₁ aplicando al menos uno del método de generación de LOD basado en árbol octal, el método de generación de LOD basado en distancia o el método de generación de LOD basado en muestreo, la unidad 53008 de configuración de conjunto de puntos contiguos puede buscar X(>0) puntos contiguos más cercanos (NN) en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basado en el conjunto LOD₁ y registrar los X puntos NN como un conjunto de puntos contiguos en un predictor.

En este caso, dado que se consume mucho tiempo para buscar todos los puntos con el fin de configurar el conjunto de puntos contiguos, se proporciona una realización de búsqueda de puntos contiguos dentro de un intervalo de búsqueda de puntos contiguos que incluye solo puntos parciales. El intervalo de búsqueda se refiere al número de puntos y puede ser 128, 256 u otro valor. Según las realizaciones, la información sobre el intervalo

de búsqueda puede establecerse en la unidad 53008 de configuración de conjunto de puntos contiguos o puede introducirse como el parámetro de usuario. Alternativamente, la información sobre el intervalo de búsqueda puede señalizarse en la información de señalización a través del procesador 51005 de señalización (por ejemplo, un campo `lifting_search_range` señalado en el APS).

- 5 La fig. 20 es un diagrama que ilustra un ejemplo de búsqueda de puntos contiguos en un intervalo de búsqueda basado en LOD según las realizaciones. Las flechas ilustradas en el dibujo indican un orden de Morton según las realizaciones.

- 10 En la fig. 20, en una realización, una lista de índices incluye un conjunto de LOD (es decir, LOD_1) a la que pertenecen los puntos que se han de codificar y una lista retenida incluye al menos un conjunto de LOD inferior (por ejemplo, LOD_0 a LOD_{1-1}) basado en el conjunto de LOD_1 .

Según las realizaciones, los puntos de la lista de índices están dispuestos en orden ascendente basándose en el tamaño de un código de Morton, y los puntos de la lista retenida también están dispuestos en orden ascendente basándose en el tamaño del código de Morton. Por lo tanto, el punto más destacado de los puntos dispuestos en orden de Morton en la lista de índices y la lista retenida tiene el tamaño más pequeño del código de Morton.

- 15 La unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones puede buscar un punto que tiene un código de Morton más cercano a un código de Morton de un punto Px entre puntos ubicados antes del punto Px en orden (es decir, puntos que tienen códigos de Morton menores que o iguales al código de Morton del punto Px) entre puntos que pertenecen al conjunto de LOD_0 a LOD_{1-1} y/o puntos pertenecientes al conjunto de LOD_1 , para generar un conjunto de puntos contiguos del punto Px (es decir, un punto que se ha de codificar o un punto actual) que pertenece al conjunto de LOD_1 . En la presente descripción, el punto buscado se denominará Pi o un punto central.
- 20

- Según las realizaciones, cuando se busca el punto central Pi, la unidad 53008 de configuración de conjunto de puntos contiguos puede buscar el punto Pi que tiene un código de Morton más cercano a un código de Morton del punto Px entre todos los puntos ubicados antes del punto Px o buscar el punto Pi que tiene el código de Morton más cercano al código de Morton del punto Px entre puntos en un intervalo de búsqueda. En la presente descripción, el intervalo de búsqueda puede configurarse por la unidad 53008 de configuración de conjunto de puntos contiguos o puede introducirse como un parámetro de usuario. Además, la información sobre el intervalo de búsqueda puede señalizarse en la información de señalización a través del procesador 51005 de señalización. La presente descripción proporciona una realización de búsqueda del punto central Pi en la lista retenida cuando el número de LOD es plural y búsqueda del punto central Pi en la lista de índices cuando el número de LOD es uno. Por ejemplo, cuando el número de LOD es 1, el intervalo de búsqueda puede determinarse basándose en la posición de un punto actual en una lista dispuesta con códigos de Morton.
- 25
- 30

- La unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones compara valores de distancia entre el punto Px y puntos que pertenecen a un intervalo de búsqueda de puntos contiguos ubicado antes del punto central Pi buscado (o seleccionado) (es decir, en el lado izquierdo del punto central en la fig. 20) y después del punto central Pi buscado (o seleccionado) (es decir, en el lado derecho del punto central en la fig. 20). La unidad 53008 de configuración de conjunto de puntos contiguos puede registrar X (por ejemplo, 3) puntos NN como un conjunto de puntos contiguos. En una realización, el intervalo de búsqueda de puntos contiguos es el número de puntos. El intervalo de búsqueda de puntos contiguos según las realizaciones puede incluir uno o más puntos ubicados antes (es decir, delante) y/o después (es decir, detrás) del punto central Pi en orden de Morton. X es el número máximo de puntos que pueden registrarse como puntos contiguos.
- 35
- 40

- Según las realizaciones, la información sobre el intervalo de búsqueda de puntos contiguos y la información sobre el número máximo X de puntos que pueden registrarse como puntos contiguos pueden configurarse por la unidad 53008 de configuración de conjunto de puntos contiguos o pueden introducirse como un parámetro de usuario o señalizarse en la información de señalización a través del procesador 51005 de señalización (por ejemplo, un campo `lifting_search_range` y un campo `lifting_num_pred_nearest_neighbours` señalizados en un APS). Según las realizaciones, un intervalo de búsqueda real puede ser un valor obtenido multiplicando el valor del campo `lifting_search_range` por 2 y, a continuación, sumando el punto central al valor resultante (es decir, (valor del campo `lifting_search_range` * 2) + punto central), un valor obtenido sumando el valor central al valor del campo `lifting_search_range` (es decir, valor del campo `lifting_search_range` + punto central), o el valor del campo `lifting_search_range`. La presente descripción proporciona una realización de búsqueda de puntos contiguos del punto Px en la lista retenida cuando el número de LOD es plural y búsqueda de puntos contiguos del punto Px en la lista de índices cuando el número de LOD es uno.
- 45
- 50

- Por ejemplo, si el número de LOD es 2 o más y la información sobre el intervalo de búsqueda (por ejemplo, el campo `lifting_search_range`) es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto central Pi, 128 puntos después del punto central Pi y el punto central Pi en la lista retenida dispuesta con códigos de Morton. Como otro ejemplo, si el número de LOD es 1 y la información sobre el intervalo de búsqueda es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto central Pi y el punto central Pi en la lista de índices dispuesta con códigos de Morton. Como otro ejemplo, si el número de LOD es 1 y la información sobre el
- 55

intervalo de búsqueda es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto actual Px en la lista de índices dispuesta con códigos de Morton.

Según las realizaciones, la unidad 53008 de configuración de conjunto de puntos contiguos puede comparar valores de distancia entre puntos dentro del intervalo de búsqueda real y el punto Px y registrar X puntos NN como un conjunto de puntos contiguos del punto Px.

Por otro lado, las características del atributo del contenido pueden aparecer de manera diferente dependiendo de un objeto del que se capture el contenido, un esquema capturado o un equipo utilizado para capturar el contenido.

(a) de la fig. 21 y (b) de la fig. 21 son diagramas que ilustran ejemplos de contenido de la nube de puntos según las realizaciones.

(a) de la fig. 21 es un ejemplo de datos densos de la nube de puntos adquiridos capturando un objeto con un escáner 3D y una correlación de atributos del objeto puede ser alta. (b) de la fig. 21 es un ejemplo de datos dispersos de la nube de puntos adquiridos capturando un área amplia con un dispositivo LiDAR y una correlación de atributos entre puntos puede ser baja. Un ejemplo de los datos densos de la nube de puntos puede ser una imagen fija y un ejemplo de los datos dispersos de la nube de puntos puede ser una imagen capturada por un dron o un vehículo autónomo.

De esta manera, las características del atributo del contenido pueden aparecer de manera diferente dependiendo de un objeto del que se capture el contenido, un esquema capturado o un equipo utilizado para capturar el contenido. Si se realiza una búsqueda de puntos contiguos para configurar un conjunto de puntos contiguos cuando se codifican atributos sin tener en cuenta diferentes características del atributo, las distancias entre un punto correspondiente y los puntos contiguos registrados pueden no ser todas adyacentes. Por ejemplo, cuando X (por ejemplo, 3) puntos NN se registran como el conjunto de puntos contiguos después de comparar los valores de distancia entre el punto Px y los puntos que pertenecen al intervalo de búsqueda de puntos contiguos antes y después del punto central Pi, al menos uno de los X puntos contiguos registrados puede estar lejos del punto Px. En otras palabras, al menos uno de los X puntos contiguos registrados puede no ser un punto contiguo real del punto Px.

Es más probable que este fenómeno ocurra en los datos dispersos de la nube de puntos que en los datos densos de la nube de puntos. Por ejemplo, como se ilustra en (a) de la fig. 21, si hay un objeto en el contenido, el objeto tiene continuidad/similitud de color y se captura densamente, y un área capturada es pequeña, puede haber contenido que tenga una alta correlación de atributos entre puntos incluso cuando los puntos están un poco separados de un punto contiguo. Por el contrario, como se ilustra en (b) de la fig. 21, si un área considerablemente amplia se captura escasamente con un dispositivo LiDAR, puede haber contenido que tenga una correlación de color solo cuando la distancia al punto contiguo está dentro de un intervalo específico. Por lo tanto, cuando se codifican atributos de los datos dispersos de la nube de puntos como se ilustra en (b) de la fig. 21, algunos de los puntos contiguos registrados a través de la búsqueda de puntos contiguos pueden no ser puntos contiguos reales del punto que se ha de codificar.

Este fenómeno tiene una alta probabilidad de ocurrir cuando los puntos se disponen en orden de Morton cuando se genera LOD. Es decir, aunque la orden de Morton dispone rápidamente puntos contiguos, se produce una sección de salto (es decir, líneas largas intermitentes en la fig. 19) debido a una característica de exploración en zigzag del código de Morton y, a continuación, las ubicaciones de los puntos pueden cambiarse en gran medida. En este caso, algunos de los puntos contiguos registrados a través de la búsqueda de puntos contiguos dentro del intervalo de búsqueda pueden no ser los puntos NN reales del punto que se ha de codificar.

(a) de la fig. 22 y (b) de la fig. 22 son diagramas que ilustran ejemplos de una distancia promedio, una distancia mínima y una distancia máxima de puntos que pertenecen a un conjunto de puntos contiguos según las realizaciones.

(a) de la fig. 22 ilustra un ejemplo de la 100ª trama de contenido denominada FORD y (b) de la fig. 22 ilustra un ejemplo de la primera trama de contenido denominada QNX.

(a) de la fig. 22 y (b) de la fig. 22 son realizaciones ejemplares para ayudar a los expertos en la técnica a comprender. (a) de la fig. 22 y (b) de la fig. 22 pueden ser tramas diferentes del mismo contenido o tramas específicas de contenido diferente.

Por conveniencia de la explicación, la trama ilustrada en (a) de la fig. 22 se denomina primera trama y la trama ilustrada en (b) de la fig. 22 se denomina segunda trama.

En la primera trama de (a) de la fig. 22, el número total de puntos es 80265, hay 4 puntos en los que solo se registra un punto NN, hay 60 puntos en los que se registran 2 puntos NN, y hay 80201 puntos en los que se

registran 3 puntos NN. Es decir, el número máximo de puntos que pueden registrarse como puntos NN es 3 pero puede ser 1 o 2 dependiendo de las posiciones de los puntos en la trama.

En la segunda trama de (b) de la fig. 22, el número total de puntos es 31279, hay 3496 puntos en los que sólo se registra un NN, hay 219 puntos en los que se registran 2 puntos de NN, y hay 27564 puntos en los que se registran 3 puntos de NN.

En la primera trama de (a) de la fig. 22 y la segunda trama de (b) de la fig. 22, NN1 es el punto NN, NN2 es el punto NN siguiente, y NN3 es el tercer punto NN. La distancia máxima de NN2 en la primera trama de (a) de la fig. 22 es 1407616 y la distancia máxima de NN3 es 1892992, mientras que la distancia máxima de NN2 en la segunda trama de (b) de la fig. 22 es 116007040 y la distancia máxima de NN3 es 310887696.

Es decir, se puede apreciar que hay una diferencia considerable en la distancia según el contenido o la trama. Por consiguiente, algunos de los puntos contiguos registrados pueden no ser puntos contiguos reales.

De esta manera, si la diferencia de distancia entre un punto que se ha de codificar y un punto contiguo registrado se vuelve grande, es probable que una diferencia de atributos entre los dos puntos sea grande. Además, cuando se realiza predicción basándose en el conjunto de puntos contiguos que incluye estos puntos contiguos y se obtiene un valor de atributo residual, el valor de atributo residual puede aumentar, dando como resultado un aumento en el tamaño de flujo de bits.

En otras palabras, las características del contenido o una trama pueden tener un efecto sobre la configuración del conjunto de puntos contiguos para la predicción de atributos y esto puede afectar a la eficiencia de compresión de los atributos.

Por lo tanto, en esta descripción, cuando se configura el conjunto de puntos contiguos para codificar atributos de contenido de la nube de puntos, se seleccionan puntos contiguos en consideración de una correlación de atributos entre puntos del contenido, de modo que no se seleccionan puntos sin significado como el conjunto de puntos contiguos. A continuación, dado que un valor de atributo residual disminuye y el tamaño de flujo de bits se reduce de ese modo, se mejora la eficiencia de compresión de atributos. En otras palabras, la presente descripción mejora la eficiencia de compresión de atributos utilizando un método de restricción de puntos que puede seleccionarse como el conjunto de puntos contiguos en consideración de la correlación de atributos entre puntos de contenido.

La presente descripción proporciona una realización de aplicación de una distancia máxima de un punto contiguo (también denominada distancia NN máxima) para considerar una correlación de atributos entre puntos de contenido cuando se configura un conjunto de puntos contiguos en un proceso de codificación de atributos de contenido de la nube de puntos.

La presente descripción proporciona una realización de configuración del conjunto de puntos contiguos aplicando un intervalo de búsqueda y/o una distancia NN máxima cuando se codifican atributos de contenido de la nube de puntos.

Por ejemplo, como resultado de comparar valores de distancia entre un punto Px y puntos pertenecientes a un intervalo de búsqueda de puntos contiguos antes y después de un punto central Pi, los puntos más alejados que la distancia NN máxima entre X (por ejemplo, 3) puntos NN no se registran en el conjunto de puntos contiguos del punto Px. En otras palabras, solo los puntos dentro de la distancia NN máxima entre X (por ejemplo, 3) puntos NN se registran en el conjunto de puntos contiguos del punto Px como resultado de comparar los valores de distancia entre el punto Px y los puntos pertenecientes al intervalo de búsqueda de puntos contiguos antes y después del punto central Pi. Por ejemplo, si uno de los tres puntos que tiene la distancia más cercana está más alejado que la distancia NN máxima, el punto más alejado puede excluirse y solo los dos puntos restantes pueden registrarse en el conjunto de puntos contiguos del punto Px.

La unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones puede aplicar la distancia NN máxima según las características del atributo del contenido para lograr una eficiencia de compresión óptima de atributos independientemente de las características del atributo del contenido. Aquí, la distancia NN máxima puede calcularse de manera diferente según el método de generación de LOD (el método de generación de LOD basado en árbol octal, basado en distancia o basado en muestreo). Es decir, la distancia NN máxima puede aplicarse de manera diferente a cada LOD.

En la presente descripción, la distancia NN máxima se utiliza indistintamente con una distancia máxima de un punto contiguo o una distancia máxima entre puntos contiguos. En otras palabras, en la presente descripción, la distancia NN máxima, la distancia máxima entre puntos contiguos y la distancia máxima entre puntos contiguos tienen el mismo significado.

En la presente descripción, la distancia NN máxima puede obtenerse multiplicando una distancia base entre puntos contiguos (denominada distancia base o distancia de referencia) por el NN_range como se ilustra en la Ecuación 5 a continuación.

[Ecuación 5]

5 Distancia máxima entre puntos contiguos = distancia base entre puntos contiguos * NN_range

En la ecuación 5, NN_range es un intervalo dentro del cual se puede seleccionar un punto contiguo y se puede denominar intervalo máximo de puntos contiguos, intervalo de puntos contiguos o intervalo de NN.

10 La unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones puede establecer NN_range automática o manualmente según las características del contenido o establecer NN_range a través de la entrada como un parámetro de usuario. Además, la información sobre el NN_range puede señalizarse en la información de señalización a través del procesador 51005 de señalización. La información de señalización que incluye la información sobre el NN_range puede ser al menos uno de un SPS, un APS, un conjunto de parámetros de mosaico o una cabecera de segmento de atributo. El decodificador de atributo del dispositivo de recepción puede generar el conjunto de puntos contiguos basándose en la información de señalización. En este caso, los puntos contiguos que se han utilizado en la predicción de atributos de codificación de atributos pueden restaurarse y aplicarse a la decodificación de atributos.

20 Según las realizaciones, la unidad 53008 de configuración de conjunto de puntos contiguos puede calcular/configurar la distancia base entre puntos contiguos combinando uno o más de un método basado en árbol octal, un método basado en distancia, un método basado en muestreo, un método basado en diferencias promedio de códigos de Morton para cada LOD, y un esquema basado en diferencias de distancia promedio para cada LOD. Según las realizaciones, el método de cálculo/configuración de la distancia base entre puntos contiguos puede señalizarse en la información de señalización (por ejemplo, APS) a través del procesador 51005 de señalización.

25 A continuación, se describirán realizaciones de adquisición de una distancia base de un punto contiguo y una distancia máxima de un punto contiguo cuando se generan LOD basándose en un árbol octal.

Como se ilustra en la fig. 18, cuando la unidad 53007 de configuración de LOD genera LOD basados en árbol octal, cada nivel de profundidad del árbol octal puede coincidir con cada LOD.

La unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones puede obtener una distancia máxima de un punto contiguo basándose en los LOD basados en árbol octal.

30 Según las realizaciones, cuando la unidad 53007 de configuración de LOD genera los LOD basados en árbol octal, puede soportarse la escalabilidad espacial. Con escalabilidad espacial, puede accederse a una nube de puntos de resolución más baja, como una miniatura con menos complejidad de decodificador y/o menos ancho de banda, cuando una nube de puntos de origen es densa. Por ejemplo, una función de escalabilidad espacial de geometría puede proporcionarse mediante un proceso de codificación o decodificación de bits de ocupación solo hasta un nivel de profundidad seleccionado ajustando un nivel de profundidad del árbol octal durante la codificación/decodificación de geometría. Además, incluso durante la codificación/decodificación de un atributo, la función de escalabilidad espacial del atributo puede proporcionarse mediante un proceso de generación de LOD a partir de un nivel de profundidad seleccionado del árbol octal y de configuración de puntos en los que ha de realizarse la codificación/decodificación del atributo.

40 Según las realizaciones, para la escalabilidad espacial, la unidad 53007 de configuración de LOD puede generar los LOD basados en árbol octal y la unidad 53008 de configuración de conjunto de puntos contiguos puede realizar una búsqueda de puntos contiguos basándose en los LOD generados basados en árbol octal.

45 Cuando la unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones busca puntos contiguos en LOD anteriores basándose en puntos que pertenecen a un LOD generado basado en árbol octal actual, puede obtenerse una distancia máxima entre puntos contiguos multiplicando una distancia base entre puntos contiguos (denominada distancia base o distancia de referencia) por NN_range (es decir, distancia base entre puntos contiguos * NN_range).

50 Según las realizaciones, una distancia máxima (es decir, una distancia diagonal) de un nodo en un LOD generado basado en árbol octal específico puede definirse como la distancia base (o una distancia base entre puntos contiguos).

Según las realizaciones, cuando se buscan puntos contiguos en LOD anteriores (por ejemplo, LOD₀ a LOD₁₋₁ en la lista retenida) basándose en puntos que pertenecen a un LOD actual (por ejemplo, LOD₁ en la lista de índices), una distancia diagonal de un nodo superior (nodo padre) de un nodo de árbol octal del LOD actual puede ser una distancia base para adquirir la distancia máxima entre puntos contiguos.

Según una realización, la distancia base, que es la distancia diagonal (es decir, la distancia máxima) de un nodo en un LOD específico, puede obtenerse como en la Ecuación 6 a continuación.

[Ecuación 6]

$$\text{Distancia base} = \sqrt{2^{lod^2} \times 3}$$

Según otra realización, la distancia base, que es la distancia diagonal (es decir, la distancia máxima) de un nodo en un LOD específico, puede obtenerse basándose en L2 como se muestra en la Ecuación 7 a continuación.

[Ecuación 7]

$$\text{distancia base basada en L2} = 2^{lod^2} \times 3$$

En general, cuando se calcula la distancia entre dos puntos, se utiliza el método de cálculo de distancia Manhattan o el método de cálculo de distancia euclidiana. La distancia Manhattan se denominará distancia L1 y la distancia euclidiana se denominará distancia L2. Un espacio euclidiano puede definirse utilizando la distancia euclidiana y una norma correspondiente a esta distancia se denominará norma euclidiana o norma L2. La norma es un método (función) de medición de la longitud o tamaño de un vector.

(a) de la fig. 23 ilustra un ejemplo de adquisición de una distancia diagonal de un nodo de árbol octal de LOD₀. Según las realizaciones, cuando el ejemplo de (a) de la fig. 23 se aplica a la Ecuación 6, la distancia base se convierte en $\sqrt{3}$ y, cuando el ejemplo de (a) de la fig. 23 se aplica a la Ecuación 7, la distancia base se convierte en 3.

(b) de la fig. 23 ilustra un ejemplo de cálculo de una distancia diagonal de un nodo de árbol octal de LOD₁. Según las realizaciones, cuando el ejemplo de (b) de la fig. 23 se aplica a la Ecuación 6, la distancia base se convierte en $\sqrt{12}$ y, cuando el ejemplo de (b) de la fig. 23 se aplica a la ecuación 7, la distancia base se convierte en 12.

(a) de la fig. 24 y (b) de la fig. 24 ilustran ejemplos de un intervalo NN_range que puede seleccionarse como un punto contiguo en cada LOD. Más específicamente, (a) de la fig. 24 ilustra un ejemplo cuando el NN_range es 1 en LOD₀ y (b) de la fig. 24 ilustra un ejemplo cuando el NN_range es 3 en LOD₀. Según las realizaciones, el NN_range puede establecerse automática o manualmente según las características del contenido o puede establecerse a través de la entrada como un parámetro de usuario. La información sobre el NN_range puede señalizarse en la información de señalización. La información de señalización que incluye la información sobre el NN_range puede ser al menos uno de un SPS, un APS, un conjunto de parámetros de mosaico o una cabecera de segmento de atributo. Según las realizaciones, el intervalo NN_range que puede establecerse como un punto contiguo puede establecerse a un valor arbitrario (por ejemplo, un múltiplo de la distancia base) independientemente de un intervalo de nodos de árbol octal.

Según las realizaciones, cuando el ejemplo de (a) de la fig. 24 se aplica a las ecuaciones 5 y 6, la distancia máxima entre puntos contiguos se convierte en $\sqrt{3}(=\sqrt{3} \times 1)$ y, cuando el ejemplo de (b) de la fig. 24 se aplica a la ecuación 5 y a la ecuación 6, la distancia máxima entre puntos contiguos se convierte en $\sqrt{3}(=\sqrt{3} \times 3)$.

Según las realizaciones, cuando el ejemplo de (a) de la fig. 24 se aplica a la Ecuación 5 y la Ecuación 7, la distancia máxima entre puntos contiguos se convierte en 3 ($=3 \times 1$) y, cuando el ejemplo de (b) de la fig. 24 se aplica a la Ecuación 5 y la Ecuación 7, la distancia máxima entre puntos contiguos es 9 ($=3 \times 3$).

A continuación, se describirán realizaciones de adquisición de la distancia base entre puntos contiguos y la distancia máxima entre puntos contiguos cuando se generan LOD basados en la distancia.

Según las realizaciones, cuando se configuran los LOD basados en la distancia, la distancia base entre puntos contiguos puede establecerse como una distancia utilizada para configurar puntos en un LOD actual (es decir, LOD₁). Es decir, la distancia base para cada LOD puede establecerse a una distancia (dist_{2L}) aplicada a la generación de LOD a un nivel L de un LOD.

La fig. 25 ilustra ejemplos de una distancia base entre puntos contiguos base pertenece a cada LOD según las realizaciones.

Por lo tanto, cuando se configuran LOD basados en la distancia, la distancia máxima entre puntos contiguos puede obtenerse multiplicando una distancia dist_{2L} base de punto contiguo basada en la distancia mediante NN_range.

Según las realizaciones, el NN_range puede establecerse automática o manualmente según las características del contenido mediante la unidad 53008 de configuración de conjunto de puntos contiguos o puede establecerse a través de la entrada como un parámetro de usuario. La información sobre el NN_range puede señalizarse en la información de señalización. La información de señalización que incluye la información sobre NN_range puede ser al menos uno de un SPS, un APS, un conjunto de parámetros de mosaico o una cabecera de segmento de atributo. Según las realizaciones, NN_range que puede establecerse como un punto contiguo puede establecerse en un valor arbitrario (por ejemplo, un múltiplo de la distancia base). Según las realizaciones, NN_range puede ajustarse según un método de cálculo de una distancia al punto contiguo.

A continuación, se describirán realizaciones de adquisición de una distancia base entre puntos contiguos y una distancia máxima entre puntos contiguos cuando se generan LOD basados en muestreo.

Según las realizaciones, cuando se generan LOD basados en muestreo (es decir, basados en diezmado), la distancia base puede determinarse según una tasa de muestreo. Según las realizaciones, en un método de configuración de LOD basado en muestreo, los códigos de Morton se generan basándose en los valores de posición de los puntos, los puntos se disponen según los códigos de Morton y, a continuación, los puntos que no corresponden al k-ésimo punto según el orden dispuesto se registran en un LOD actual. Por lo tanto, cada LOD puede tener un k diferente y puede expresarse como k_L .

Según una realización, cuando se generan LOD aplicando un método de muestreo a puntos dispuestos basándose en códigos de Morton, una distancia promedio de puntos k_L -ésimos consecutivos en cada LOD pueden establecerse a la distancia base entre puntos contiguos en cada LOD. Es decir, la distancia promedio de puntos k_L -ésimos contiguos en un LOD actual dispuestos basándose en los códigos de Morton pueden establecerse a la distancia base entre puntos contiguos del LOD actual. Según otra realización, cuando se generan LOD aplicando el método de muestreo a puntos dispuestos basándose en los códigos de Morton, puede estimarse un nivel de profundidad de un árbol octal a partir del nivel de LOD actual, y puede establecerse una longitud diagonal del nodo del nivel de profundidad de árbol octal estimado para la distancia base entre puntos contiguos. Es decir, la distancia base entre puntos contiguos puede obtenerse aplicando la ecuación 6 o 7 descritas anteriormente.

Por lo tanto, cuando se configuran LOD basados en muestreo, la distancia máxima entre puntos contiguos puede obtenerse multiplicando la distancia base entre puntos contiguos basada en muestreo por el NN_range.

Según las realizaciones, el NN_range puede establecerse automática o manualmente según las características del contenido por la unidad 53008 de configuración de conjunto de puntos contiguos o puede establecerse a través de la entrada como un parámetro de usuario. La información sobre el NN_range puede señalizarse en la información de señalización. La información de señalización que incluye la información sobre el NN_range puede ser al menos uno de un SPS, un APS, un conjunto de parámetros de mosaico o una cabecera de segmento de atributo. Según las realizaciones, el NN_range que puede establecerse como un punto contiguo puede establecerse en un valor arbitrario (por ejemplo, un múltiplo de la distancia base). Según las realizaciones, NN_range puede ajustarse según un método de cálculo de una distancia al punto contiguo.

En la presente descripción, la distancia base entre puntos contiguos puede obtenerse utilizando un método distinto del método descrito anteriormente de adquisición de la distancia base entre puntos contiguos.

Por ejemplo, cuando se configuran los LOD, la distancia base entre puntos contiguos puede configurarse calculando una diferencia promedio de códigos de Morton de puntos de muestreo. Además, la distancia máxima entre puntos contiguos puede obtenerse multiplicando la distancia base entre puntos contiguos configurada por el NN_range.

Como otro ejemplo, la distancia base entre puntos contiguos puede configurarse calculando una diferencia de distancia promedio para los LOD configurados actualmente independientemente de si se utiliza el método basado en distancia o se utiliza el método basado en muestreo. Además, la distancia máxima entre puntos contiguos puede obtenerse multiplicando la distancia base entre puntos contiguos configurada por NN_range.

Como otro ejemplo, la distancia base entre puntos contiguos para cada LOD puede recibirse como el parámetro de usuario y, a continuación, aplicarse cuando se adquiere la distancia máxima entre puntos contiguos. A continuación, la información sobre la distancia base entre puntos contiguos recibida puede señalizarse en la información de señalización.

Como se ha descrito anteriormente, la distancia base entre puntos contiguos puede obtenerse aplicando el mejor método de cálculo de distancia base entre puntos contiguos según el método de configuración de LOD. En otra realización, el método de cálculo de la distancia base entre puntos contiguos puede aplicarse seleccionando un método arbitrariamente deseado y señalizando el método seleccionado en la información de señalización para utilizar el método para restaurar puntos en el decodificador del dispositivo de recepción. Según las realizaciones, se puede utilizar un método para calcular la distancia máxima entre puntos contiguos que es adecuada para un atributo de contenido de la nube de puntos y es capaz de maximizar el rendimiento de compresión.

El método de configuración de LOD, el método de adquisición de distancia base entre puntos contiguos, el método de adquisición de NN_range y el método de adquisición de distancia máxima entre puntos contiguos, descritos anteriormente, pueden aplicarse de la misma manera o similar cuando se configura un LOD o un conjunto de puntos contiguos para transformación de predicción o transformación de elevación.

Según una realización, si se genera el conjunto de LOD₁ y se determina la distancia máxima entre puntos contiguos como se ha descrito anteriormente, la unidad 53008 de configuración de conjunto de puntos contiguos busca X (por ejemplo, 3) puntos NN entre puntos dentro de un intervalo de búsqueda en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto de LOD₁. A continuación, la unidad 53008 de configuración de conjunto de puntos contiguos puede registrar solo puntos NN dentro de la distancia máxima entre puntos contiguos entre los X (por ejemplo, 3) puntos NN como un conjunto de puntos contiguos. Por lo tanto, el número de puntos NN registrados como el conjunto de puntos contiguos es igual o menor que X. En otras palabras, los puntos NN que no están dentro de la distancia máxima entre puntos contiguos entre los X puntos NN no están registrados como el conjunto de puntos contiguos y están excluidos del conjunto de puntos contiguos. Por ejemplo, si dos de los tres puntos NN no están dentro de la distancia máxima entre puntos contiguos, se excluyen los dos puntos NN y sólo se registra el otro punto NN, es decir, un punto NN, como el conjunto de puntos contiguos.

Según otra realización, si se genera el conjunto de LOD₁ y se determina la distancia máxima entre puntos contiguos como se ha descrito anteriormente, la unidad 53008 de configuración de conjunto de puntos contiguos puede buscar X (por ejemplo, 3) puntos NN dentro de la distancia máxima entre puntos contiguos entre puntos dentro del intervalo de búsqueda en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto de LOD₁ y registrar los X puntos NN como el conjunto de puntos contiguos.

Es decir, el número de puntos NN que pueden registrarse como el conjunto de puntos contiguos puede variar según una temporización (o posición) de aplicación de la distancia máxima entre puntos contiguos. En otras palabras, el número de puntos NN registrados como el conjunto de puntos contiguos puede diferir según si la distancia máxima entre puntos contiguos se aplica después de buscar X puntos NN a través de la búsqueda de puntos contiguos o la distancia máxima entre puntos contiguos se aplica al calcular una distancia entre puntos. X es el número máximo que puede establecerse como un punto contiguo y puede introducirse como un parámetro de usuario o señalizarse en la información de señalización a través del procesador 51005 de señalización (por ejemplo, el campo `lifting_num_pred_nearest_neighbours` señalizado en el APS).

La fig. 26 es un diagrama que ilustra un ejemplo de búsqueda de puntos contiguos aplicando un intervalo de búsqueda y una distancia máxima entre puntos contiguos según las realizaciones. Las flechas ilustradas en el dibujo indican un orden de Morton según las realizaciones.

En la fig. 26, en una realización, una lista de índices incluye un conjunto de LOD (es decir, LOD₁) a la que pertenecen los puntos que se han de codificar y una lista retenida incluye al menos un conjunto de LOD inferior (por ejemplo, LOD₀ a LOD₁₋₁) basado en un conjunto de LOD₁.

Según las realizaciones, los puntos en la lista de índices y los puntos en la lista retenida están dispuestos en orden ascendente basándose en el tamaño de un código de Morton. Por lo tanto, el punto más destacado de los puntos dispuestos en orden de Morton en la lista de índices y en la lista retenida tiene el tamaño más pequeño del código de Morton.

La unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones puede buscar un punto central P_i que tiene un código de Morton más cercano a un código de Morton de un punto P_x entre puntos ubicados antes del punto P_x en orden (es decir, puntos que tienen códigos de Morton menores que o iguales al código de Morton del punto P_x) entre puntos que pertenecen a los conjunto de LOD₀ a LOD₁₋₁ y/o puntos pertenecientes al conjunto de LOD₁, para registrar un conjunto de puntos contiguos del punto P_x (es decir, un punto que se ha de codificar o un punto actual) perteneciente al conjunto de LOD₁.

Según las realizaciones, cuando se busca el punto central P_i, la unidad 53008 de configuración de conjunto de puntos contiguos puede buscar el punto P_i que tiene el código de Morton más cercano al código de Morton del punto P_x entre todos los puntos ubicados antes del punto P_x o buscar el punto P_i que tiene el código de Morton más cercano al código de Morton del punto P_x entre puntos en un intervalo de búsqueda. En la presente descripción, el intervalo de búsqueda puede configurarse mediante la unidad 53008 de configuración de conjunto de puntos contiguos o puede introducirse como un parámetro de usuario. Además, la información sobre el intervalo de búsqueda puede señalizarse en la información de señalización a través del procesador 51005 de señalización. La presente descripción proporciona una realización de búsqueda del punto central P_i en la lista retenida cuando el número de LOD es plural y búsqueda del punto central P_i en la lista de índices cuando el número de LOD es uno. Por ejemplo, cuando el número de LOD es 1, el intervalo de búsqueda puede determinarse basándose en la posición de un punto actual en una lista dispuesta con códigos de Morton.

La unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones compara valores de distancia entre el punto Px y puntos que pertenecen a un intervalo de búsqueda de puntos contiguos ubicado antes del punto central Pi buscado (o seleccionado) (es decir, en el lado izquierdo del punto central en la fig. 26) y después del punto central Pi buscado (o seleccionado) (es decir, en el lado derecho del punto central en la fig. 26). La unidad 53008 de configuración de conjunto de puntos contiguos puede seleccionar X (por ejemplo, 3) puntos NN y registrar solo puntos dentro de la distancia máxima entre puntos contiguos en un LOD al que pertenece el punto Px entre los X puntos seleccionados como un conjunto de puntos contiguos del punto Px. En una realización, el intervalo de búsqueda de puntos contiguos es el número de puntos. El intervalo de búsqueda de puntos contiguos según las realizaciones puede incluir uno o más puntos ubicados antes (es decir, delante) y/o después (es decir, detrás) del punto central Pi en orden de Morton. X es el número máximo de puntos que pueden registrarse como puntos contiguos.

Según las realizaciones, la información sobre el intervalo de búsqueda de puntos contiguos y la información sobre el número máximo X de puntos que pueden registrarse como puntos contiguos pueden configurarse mediante la unidad 53008 de configuración de conjunto de puntos contiguos o pueden introducirse como un parámetro de usuario o señalizarse en la información de señalización a través del procesador 51005 de señalización (por ejemplo, el campo `lifting_search_range` y el campo `lifting_num_pred_nearest_neighbours` señalizados en el APS). Según las realizaciones, un intervalo de búsqueda real puede ser un valor obtenido multiplicando el valor del campo `lifting_search_range` por 2 y, a continuación, sumando el punto central al valor resultante (es decir, (valor del campo `lifting_search_range` * 2) + punto central), un valor obtenido sumando el valor central al valor del campo `lifting_search_range` (es decir, valor del campo `lifting_search_range` + punto central), o el valor del campo `lifting_search_range`. La presente descripción proporciona una realización de búsqueda de puntos contiguos del punto Px en la lista retenida cuando el número de LOD es plural y búsqueda de puntos contiguos del punto Px en la lista de índices cuando el número de LOD es uno.

Por ejemplo, si el número de LOD es 2 o más y la información sobre el intervalo de búsqueda (por ejemplo, el campo `lifting_search_range`) es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto central Pi, 128 puntos después del punto central Pi y el punto central Pi en la lista retenida es dispuesta con códigos de Morton. Como otro ejemplo, si el número de LOD es 1 y la información sobre el intervalo de búsqueda es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto central Pi y el punto central Pi en la lista de índices dispuesta con códigos de Morton. Como otro ejemplo, si el número de LOD es 1 y la información sobre el intervalo de búsqueda es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto actual Px en la lista de índices dispuesta con códigos de Morton.

Según las realizaciones, la unidad 53008 de configuración de conjunto de puntos contiguos compara los valores de distancia entre puntos en el intervalo de búsqueda real y el punto Px para buscar X puntos NN y registrar solo puntos dentro de la distancia máxima entre puntos contiguos en un LOD al que pertenece el punto Px entre los X puntos como el conjunto de puntos contiguos del punto Px. Es decir, los puntos contiguos registrados como el conjunto de puntos contiguos del punto Px se limitan a puntos dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto Px entre los X puntos.

Como se ha descrito anteriormente, la presente descripción puede proporcionar un efecto de mejora de la eficiencia de compresión de atributos limitando los puntos que pueden seleccionarse como el conjunto de puntos contiguos en consideración de las características del atributo (correlación) entre puntos de contenido de la nube de puntos.

La fig. 27 es un diagrama que ilustra otro ejemplo de búsqueda de puntos contiguos aplicando un intervalo de búsqueda y una distancia máxima entre puntos contiguos según las realizaciones. Las flechas ilustradas en el dibujo indican un orden de Morton según las realizaciones.

Dado que el ejemplo de la fig. 27 es similar al de la fig. 26 excepto por una posición a la que se aplica la distancia máxima entre puntos contiguos, se omitirá en la presente memoria una descripción repetitiva del mismo. Por lo tanto, para las partes omitidas o no descritas en la fig. 27, se hace referencia a la descripción de la fig. 26.

Según las realizaciones, los puntos de la lista de índices y los puntos de la lista retenida están dispuestos en orden ascendente basándose en el tamaño de un código de Morton.

La unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones puede buscar un punto central Pi que tiene un código de Morton más cercano a un código de Morton de un punto Px entre puntos ubicados antes del punto Px en orden (es decir, puntos que tienen códigos de Morton menores que o iguales al código de Morton del punto Px) entre puntos que pertenecen a los conjuntos de LOD₀ a LOD₁₋₁ y/o puntos pertenecientes al conjunto de LOD₁, en orden para registrar un conjunto de puntos contiguos del punto Px (es decir, un punto que se ha de codificar o un punto actual) perteneciente al conjunto LOD₁.

La unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones compara valores de distancia entre el punto Px y puntos que pertenecen a un intervalo de búsqueda de puntos contiguos ubicado antes del punto central Pi buscado (o seleccionado) (es decir, en el lado izquierdo del punto central en la fig. 27)

y después del punto central P_i buscado (o seleccionado) (es decir, en el lado derecho del punto central en la fig. 27). La unidad 53008 de configuración de conjunto de puntos contiguos puede seleccionar X (por ejemplo, 3) puntos NN dentro de la distancia máxima entre puntos contiguos en un LOD al que pertenece el punto P_x y registrar los X puntos como el conjunto de puntos contiguos del punto P_x .

5 La presente descripción proporciona una realización de búsqueda de puntos contiguos del punto P_x en un intervalo de búsqueda real de la lista retenida cuando el número de LOD es plural y búsqueda de los puntos contiguos del punto P_x en el intervalo de búsqueda real de la lista de índices cuando el número de LOD es uno.

10 Por ejemplo, si el número de LOD es 2 o más y la información sobre el intervalo de búsqueda (por ejemplo, el campo `lifting_search_range`) es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto central P_i , 128 puntos después del punto central P_i y el punto central P_i en la lista retenida dispuesta con códigos de Morton. Como otro ejemplo, si el número de LOD es 1 y la información sobre el intervalo de búsqueda es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto central P_i y el punto central P_i en la lista de índices dispuesta con códigos de Morton. Como otro ejemplo, si el número de LOD es 1 y la información sobre el intervalo de búsqueda es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto actual P_x en la lista de índices dispuesta con códigos de Morton.

15 Tras comparar los valores de distancia entre puntos en el intervalo de búsqueda real y el punto P_x , la unidad 53008 de configuración de conjunto de puntos contiguos según las realizaciones selecciona X (por ejemplo, 3) puntos NN dentro de la distancia máxima entre puntos contiguos en un LOD al que pertenece el punto P_x de entre puntos en el intervalo de búsqueda real y registra los X puntos como el conjunto de puntos contiguos del punto P_x . Es decir, los puntos contiguos registrados como el conjunto de puntos contiguos del punto P_x se limitan a puntos dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto P_x entre los X puntos.

20 Como se ha descrito anteriormente, la presente descripción puede proporcionar un efecto de mejora de la eficiencia de compresión de atributos seleccionando el conjunto de puntos contiguos en consideración de características del atributo (correlación) entre puntos de contenido de la nube de puntos.

25 Según las realizaciones descritas anteriormente, cuando el conjunto de puntos contiguos se registra en cada predictor de puntos que se han de codificar en la unidad 53008 de configuración de conjunto de puntos contiguos, la unidad 53009 de predicción de información de atributo predice un valor de atributo de un punto correspondiente a partir de uno o más puntos contiguos registrados en el predictor. Como se ha descrito anteriormente, cuando se configura el conjunto de puntos contiguos, el número de puntos contiguos incluidos en el conjunto de puntos contiguos registrado en cada predictor es igual a o menor que X (por ejemplo, 3) aplicando la distancia máxima entre puntos contiguos. Según las realizaciones, un predictor del punto puede registrar una media distancia (= peso) basándose en un valor de distancia a cada punto contiguo mediante un conjunto de puntos contiguos registrados. Por ejemplo, el predictor del nodo P_3 calcula un peso basándose en el valor de distancia a cada punto contiguo en (P_2 P_4 P_6) como el conjunto de puntos contiguos. Según realizaciones, los pesos de puntos contiguos pueden ser $(1/\sqrt{(P_2 - P_3)^2}, 1/\sqrt{(P_4 - P_3)^2}, 1/\sqrt{(P_6 - P_3)^2})$.

30 Según las realizaciones, cuando el conjunto de puntos contiguos está registrado en el predictor, la unidad 53008 de configuración del conjunto de puntos contiguos o la unidad 53009 de predicción de información de atributo pueden normalizar un peso de cada punto contiguo en un peso total de puntos contiguos incluidos en el conjunto de puntos contiguos.

35 Por ejemplo, se añaden los pesos de todos los puntos contiguos en el conjunto contiguo del nodo P_3 ($\text{total_weight} = 1/\sqrt{(P_2 - P_3)^2} + 1/\sqrt{(P_4 - P_3)^2} + 1/\sqrt{(P_6 - P_3)^2}$), y el peso de cada punto contiguo se divide por la suma de los pesos ($1/\sqrt{(P_2 - P_3)^2}/\text{total_weight}$, $1/\sqrt{(P_4 - P_3)^2}/\text{total_weight}$, $1/\sqrt{(P_6 - P_3)^2}/\text{total_weight}$). De este modo, se normaliza el peso de cada punto contiguo

40 A continuación, la unidad 5301 de predicción de información de atributo puede predecir un valor de atributo a través del predictor.

45 Según las realizaciones, el promedio de los valores obtenidos multiplicando los atributos (por ejemplo, color, reflectancia, etc.) de puntos contiguos registrados en el predictor mediante un peso (o un peso normalizado) puede establecerse como un resultado predicho (es decir, un valor de atributo predicho). Alternativamente, un atributo de un punto específico puede establecerse como resultado predicho (es decir, un valor de atributo predicho). Según las realizaciones, el valor de atributo predicho puede denominarse información de atributo predicha. Además, se puede obtener un valor de atributo residual (o información de atributo residual o residual) restando el valor de atributo predicho (o información de atributo predicha) del punto del valor de atributo (es decir, el valor de atributo original) del punto.

50 Según las realizaciones, los valores de resultado comprimidos pueden calcularse previamente aplicando diversos modos de predicción (o índices de predictor), y, a continuación, un modo de predicción (es decir, un índice de predictor) que genera el flujo de bits más pequeño puede seleccionarse de entre los modos de predicción.

A continuación, se describirá en detalle un proceso de selección de un modo de predicción.

En esta especificación, un modo de predicción tiene el mismo significado que un índice de predicción (Preindex), y puede denominarse ampliamente como un método de predicción.

- 5 En una realización, el proceso de descubrimiento del modo de predicción más adecuado para cada punto y establecer el modo de predicción encontrado en el predictor del punto correspondiente puede ser realizado por la unidad 53009 de predicción de información de atributo.

Según las realizaciones, un modo de predicción en el que se calcula un valor de atributo predicho a través de una media ponderada (es decir, una media obtenida multiplicando los atributos de puntos contiguos establecidos en el predictor de cada punto por un peso calculado basándose en la distancia a cada punto contiguo) se denominará modo de predicción 0. Además, un modo de predicción en el que el atributo del primer punto contiguo se establece como un valor de atributo predicho se denominará modo de predicción 1, un modo de predicción en el que el atributo del segundo punto contiguo se establece como un valor de atributo predicho se denominará modo de predicción 2, y un modo de predicción en el que el atributo del tercer punto contiguo se establece como un valor de atributo predicho se denominará modo de predicción 3. En otras palabras, el valor del modo de predicción (o índice de predicción) igual a 0 puede indicar que el valor de atributo se predice a través de la media ponderada, y el valor igual a 1 puede indicar que el valor de atributo se predice a través del primer nodo contiguo (es decir, el punto contiguo). El valor igual a 2 puede indicar que el valor de atributo se predice a través del segundo nodo contiguo, y el valor igual a 3 puede indicar que el valor de atributo se predice a través del tercer nodo contiguo.

20 Según las realizaciones, se puede calcular un valor de atributo residual en el modo de predicción 0, un valor de atributo residual en el modo de predicción 1, un valor de atributo residual en el modo de predicción 2 y un valor de atributo residual en el modo de predicción 3, y se puede calcular una puntuación o doble puntuación basándose en cada valor de atributo residual. A continuación, se puede seleccionar un modo de predicción que tenga la puntuación menos calculada y establecer como un modo de predicción del punto correspondiente.

25 Según las realizaciones, el proceso de descubrimiento del modo de predicción más adecuado entre una pluralidad de modos de predicción y establecer el mismo como un modo de predicción de un punto correspondiente puede realizarse cuando se cumple una condición preestablecida. Por consiguiente, cuando no se cumple la condición preestablecida, un modo de predicción predefinido, por ejemplo, el modo de predicción 0, en el que se calcula un valor de atributo de predicción a través de una media ponderada, puede establecerse como el modo de predicción del punto sin realizar el proceso de descubrimiento del modo de predicción más adecuado. En una realización, este proceso se realiza para cada punto.

Según las realizaciones, la condición preestablecida puede cumplirse para un punto específico cuando la diferencia en los elementos de atributo (por ejemplo, R, G, B) entre puntos contiguos registrados en el predictor de un punto es mayor que o igual a un umbral preestablecido (por ejemplo, `lifting_adaptive_prediction_threshold`), o cuando se calculan las diferencias en los elementos de atributo (por ejemplo, R, G, B) entre los puntos contiguos registrados en el predictor del punto, y la suma de las mayores diferencias para los elementos es mayor que o igual a un umbral preestablecido. Por ejemplo, supóngase que el punto P3 es el punto específico, y los puntos P2, P4 y P6 se registran como puntos contiguos del punto P3. Adicionalmente, supóngase que, cuando se calculan las diferencias en los valores R, G y B entre los puntos P2 y P4, las diferencias en los valores R, G y B entre los puntos P2 y P6, y las diferencias en los valores R, G y B entre los puntos P4 y P6, se obtiene la mayor diferencia en el valor R entre los puntos P2 y P4, se obtiene la mayor diferencia en el valor G entre los puntos P4 y P6, y se obtiene la mayor diferencia en el valor B entre los puntos P2 y P6. Además, supóngase que la diferencia en el valor R entre los puntos P2 y P4 es la mayor entre la mayor diferencia en el valor R (entre P2 y P4) y la mayor diferencia en el valor G (entre P4 y P6), y la mayor diferencia en el valor B (entre P2 y P6).

En estas suposiciones, cuando la diferencia en el valor R entre los puntos P2 y P4 es mayor que o igual a un umbral preestablecido, o cuando la suma de la diferencia en el valor R entre los puntos P2 y P4, la diferencia en el valor G entre los puntos P4 y P6, y la diferencia en el valor B entre los puntos P2 y P6 es mayor que o igual a un umbral preestablecido, se puede realizar un proceso de descubrimiento del modo de predicción más adecuado entre una pluralidad de modos de predicción candidatos. Además, el modo de predicción (por ejemplo, `predIndex`) puede señalizarse solo cuando la diferencia en el valor R entre los puntos P2 y P4 es mayor que o igual a un umbral preestablecido, o la suma de la diferencia en el valor R entre los puntos P2 y P4, la diferencia en el valor G entre los puntos P4 y P6, y la diferencia en el valor B entre los puntos P2 y P6 es mayor que o igual a un umbral preestablecido.

55 Según otra realización, la condición preestablecida puede cumplirse para un punto específico cuando la diferencia máxima entre valores de un atributo (por ejemplo, reflectancia) de puntos contiguos registrados en el predictor del punto es mayor que o igual a un umbral preestablecido (por ejemplo, `lifting_adaptive_prediction_threshold`). Por ejemplo, supóngase que la diferencia en la reflectancia entre los

puntos P2 y P4 es la mayor entre la diferencia en la reflectancia entre los puntos P2 y P4, la diferencia en la reflectancia entre los puntos P2 y P6, y la diferencia en la reflectancia entre los puntos P4 y P6.

5 En esta suposición, el proceso de descubrimiento del modo de predicción más adecuado entre una pluralidad de modos de predicción candidatos puede realizarse cuando la diferencia en la reflectancia entre los puntos P2 y P4 es mayor que o igual a un umbral preestablecido. Además, el modo de predicción (por ejemplo, predIndex) puede señalizarse solo cuando la diferencia en la reflectancia entre los puntos P2 y P4 es mayor que o igual a el umbral preestablecido.

10 Según las realizaciones, el modo de predicción seleccionado (por ejemplo, predIndex) del punto correspondiente puede señalizarse en datos de segmento de atributo. En este caso, el lado de transmisión transmite un valor de atributo residual adquirido basándose en el modo de predicción seleccionado y el lado de recepción adquiere un valor de atributo predicho de un punto correspondiente basándose en el modo de predicción señalado y añade el valor de atributo predicho y el valor de atributo residual recibido para restaurar un valor de atributo del punto correspondiente.

15 En otra realización, cuando el modo de predicción no es señalado, el lado de transmisión puede calcular un valor de atributo predicho basándose en un modo de predicción (por ejemplo, modo de predicción 0) establecido como un modo por defecto, y calcular y transmitir un valor de atributo residual basándose en la diferencia entre el valor de atributo original y el valor de atributo predicho. El lado de recepción puede calcular un valor de atributo predicho basándose en un modo de predicción (por ejemplo, modo de predicción 0) establecido como un modo por defecto, y restaurar el valor de atributo añadiendo el valor al valor de atributo residual recibido.

20 Según las realizaciones, el umbral puede introducirse o señalizarse directamente en la información de señalización a través del procesador 51005 de señalización (por ejemplo, el campo lifting_adaptive_prediction_threshold señalado en el APS).

25 Según las realizaciones, cuando se cumple una condición preestablecida para un punto específico como se ha descrito anteriormente, se pueden generar candidatos de predictor. Los candidatos de predictor se denominan modos de predicción o índices de predictor.

Según las realizaciones, los modos de predicción 1 a 3 pueden incluirse en los candidatos de predictor. Según las realizaciones, el modo de predicción 0 puede o no incluirse en los candidatos de predictor. Según las realizaciones, al menos un modo de predicción no mencionado anteriormente puede incluirse además en los candidatos de predictor.

30 El conjunto de modos de predicción para cada punto a través del proceso descrito anteriormente y el valor de atributo residual en el modo de predicción establecido se emiten al procesador 53010 de cuantificación de información de atributo residual.

Según las realizaciones, el procesador 53010 de cuantificación de información de atributo residual puede aplicar codificación de longitud de ejecución cero a los valores de atributos residuales de entrada.

35 Según una realización de la presente descripción, la cuantificación y la codificación de longitud de ejecución cero pueden realizarse en los valores de atributos residuales.

Según las realizaciones, el codificador 53011 aritmético aplica codificación aritmética a los modos de predicción y valores de atributos residuales emitidos desde el procesador 53010 de cuantificación de información de atributo residual y emite el resultado como un flujo de bits de atributos.

40 El flujo de bits de geometría comprimido y emitido por el codificador 51006 de geometría y el flujo de bits de atributos comprimido y emitido por el codificador 51007 de atributos se emiten al procesador 51008 de transmisión.

45 El procesador 51008 de transmisión según las realizaciones puede realizar una operación y/o un método de transmisión idéntico o similar a la operación y/o el método de transmisión del procesador 12012 de transmisión de la fig. 12, y realizar una operación y/o un método de transmisión idéntico o similar a la operación y/o el método de transmisión del transmisor 10003 de la fig. 1. Para más detalles, se hará referencia a la descripción de las figs. 1 o 12.

50 El procesador 51008 de transmisión según las realizaciones puede transmitir el flujo de bits de geometría emitido desde el codificador 51006 de geometría, el flujo de bits de atributos emitido desde el codificador 51007 de atributos y el flujo de bits de señalización emitido desde el procesador 51005 de señalización, respectivamente, o puede multiplexar los flujos de bits en un flujo de bits que se ha de transmitir.

El procesador 51008 de transmisión según las realizaciones puede encapsular el flujo de bits en un archivo o segmento (por ejemplo, un segmento de transmisión en directo) y, a continuación, transmitir el flujo de bits encapsulado a través de diversas redes tales como una red de difusión y/o una red de banda ancha.

El procesador 51005 de señalización según las realizaciones puede generar y/o procesar información de señalización y emitir la misma al procesador 51008 de transmisión en forma de un flujo de bits. La información de señalización generada y/o procesada por el procesador 51005 de señalización se proporcionará al codificador 51006 de geometría, al codificador 51007 de atributos y/o al procesador 51008 de transmisión para la codificación de geometría, la codificación de atributos y el tratamiento de transmisión. Alternativamente, el procesador 51005 de señalización puede recibir información de señalización generada por el codificador 51006 de geometría, el codificador 51007 de atributos y/o el procesador 51008 de transmisión.

En la presente descripción, la información de señalización puede señalizarse y transmitirse en una base por conjunto de parámetros (conjunto de parámetros de secuencia (SPS), conjunto de parámetros de geometría (GPS), conjunto de parámetros de atributo (APS), conjunto de parámetros de mosaico (TPS) o similares). Además, puede señalizarse y transmitirse basándose en una unidad de codificación de cada imagen, tal como un segmento o mosaico. En la presente descripción, la información de señalización puede incluir metadatos (por ejemplo, valores establecidos) relacionados con datos de la nube de puntos, y puede proporcionarse al codificador 51006 de geometría, al codificador 51007 de atributos y/o al procesador 51008 de transmisión para codificación de geometría, codificación de atributos y tratamiento de transmisión. Dependiendo de la aplicación, la información de señalización también puede definirse en el lado del sistema, tal como un formato de archivo, transmisión dinámica adaptativa sobre HTTP (DASH) o transporte de medios MPEG (MMT), o en el lado de la interfaz cableada, tal como la interfaz multimedia de alta definición (HDMI), el puerto de visualización, la asociación de estándares de electrónica de vídeo (VESA) o CTA.

Un método/dispositivo según las realizaciones puede señalizar información relacionada para añadir/realizar una operación de las realizaciones. La información de señalización según las realizaciones puede utilizarse en un dispositivo de transmisión y/o un dispositivo de recepción.

Según una realización de la presente descripción, la información sobre el número máximo de predictores que se han de utilizar para la predicción de atributos (`lifting_max_num_direct_predictors`), la información de umbral para permitir la predicción adaptativa de un atributo (`lifting_adaptive_prediction_threshold`), la información de opción relacionada con la selección de puntos contiguos y similares se puede señalizar en al menos uno de un conjunto de parámetros de secuencia, un conjunto de parámetros de atributo, un conjunto de parámetros de mosaico o una cabecera de segmento de atributo. Además, según una realización, la información de índice de predictor (`predIndex`) que indica un modo de predicción correspondiente a un candidato de predictor seleccionado de entre una pluralidad de candidatos de predictor puede señalizarse en datos de segmento de atributo.

Según las realizaciones, la información de opción relacionada con la selección de puntos contiguos puede incluir información relacionada con el `NN_range` (por ejemplo, `nearest_neighbor_max_range`). Según las realizaciones, la información de opción relacionada con la selección de puntos contiguos puede incluir además al menos una de información sobre un número máximo que puede establecerse como un punto contiguo (por ejemplo, `lifting_num_pred_nearest_neighbours`), información relacionada con un intervalo de búsqueda (por ejemplo, `lifting_search_range`), información sobre un método de configuración de LOD y/o información relacionada con una distancia base entre puntos contiguos.

De manera similar al codificador de vídeo de la nube de puntos del dispositivo de transmisión descrito anteriormente, el decodificador de vídeo de la nube de puntos del dispositivo de recepción realiza el mismo proceso o un proceso similar de generación de un conjunto de `LOD1`, encontrando los puntos contiguos más cercanos basándose en el conjunto `LOD1` y registrar el mismo en el predictor que un conjunto de puntos contiguos, calcular un peso basándose en una distancia desde cada punto contiguo, y realizar normalización con el peso. A continuación, el decodificador decodifica un modo de predicción recibido y predice un valor de atributo del punto según el modo de predicción decodificado. Además, después de que se decodifique el valor de atributo residual recibido, el valor de atributo residual decodificado puede añadirse al valor de atributo predicho para restaurar el valor de atributo del punto.

La fig. 28 es un diagrama que muestra otro dispositivo de recepción de la nube de puntos ejemplar según las realizaciones.

El dispositivo de recepción de la nube de puntos según las realizaciones puede incluir un procesador 61001 de recepción, un procesador 61002 de señalización, un decodificador 61003 de geometría, un decodificador 61004 de atributos y un postprocesador 61005. Según las realizaciones, el decodificador 61003 de geometría y el decodificador 61004 de atributos pueden denominarse decodificador de vídeo de la nube de puntos. Según las realizaciones, el decodificador de vídeo de la nube de puntos puede denominarse decodificador de PCC, unidad de decodificación de PCC, decodificador de la nube de puntos, unidad de decodificación de la nube de puntos o similares.

El procesador 61001 de recepción según las realizaciones puede recibir un único flujo de bits, o puede recibir un flujo de bits de geometría, un flujo de bits de atributos y un flujo de bits de señalización, respectivamente. Cuando se recibe un archivo y/o segmento, el procesador 61001 de recepción según las realizaciones puede desencapsular el archivo y/o segmento recibido y emitir el archivo y/o segmento desencapsulado como un flujo de bits.

Cuando se recibe (o desencapsula) el flujo de bits único, el procesador 61001 de recepción según las realizaciones puede demultiplexar el flujo de bits de geometría, el flujo de bits de atributos y/o el flujo de bits de señalización del flujo de bits único. El procesador 61001 de recepción puede emitir el flujo de bits de señalización demultiplexado al procesador 61002 de señalización, el flujo de bits de geometría al decodificador 61003 de geometría y el flujo de bits de atributos al decodificador 61004 de atributos.

Cuando se recibe (o desencapsula) el flujo de bits de geometría, el flujo de bits de atributos y/o el flujo de bits de señalización, respectivamente, el procesador 61001 de recepción según las realizaciones puede entregar el flujo de bits de señalización al procesador 61002 de señalización, el flujo de bits de geometría al decodificador 61003 de geometría y el flujo de bits de atributos al decodificador 61004 de atributos.

El procesador 61002 de señalización puede analizar información de señalización, por ejemplo, información contenida en el SPS, GPS, APS, TPS, metadatos o similares a partir del flujo de bits de señalización de entrada, procesar la información analizada y proporcionar la información procesada al decodificador 61003 de geometría, al decodificador 61004 de atributos y al postprocesador 61005. En otra realización, la información de señalización contenida en la cabecera de segmento de geometría y/o en la cabecera de segmento de atributo también puede analizarse mediante el procesador 61002 de señalización antes de decodificar los datos de segmento correspondientes. Es decir, cuando los datos de la nube de puntos se dividen en mosaicos y/o segmentos en el lado de transmisión como se muestra en la fig. 16, el TPS incluye el número de segmentos incluidos en cada mosaico, y por consiguiente el decodificador de vídeo de la nube de puntos según las realizaciones puede comprobar el número de segmentos y analizar rápidamente la información para decodificación paralela.

Por consiguiente, el decodificador de vídeo de la nube de puntos según la presente descripción puede analizar rápidamente un flujo de bits que contiene datos de la nube de puntos a medida que recibe un SPS que tiene una cantidad reducida de datos. El dispositivo de recepción puede decodificar mosaicos al recibir los mosaicos y puede decodificar cada segmento basándose en el GPS y al APS incluidos en cada mosaico. De este modo, la eficiencia de decodificación puede maximizarse.

Es decir, el decodificador 61003 de geometría puede reconstruir la geometría realizando el proceso inverso de la operación del codificador 51006 de geometría de la fig. 15 en el flujo de bits de geometría comprimida basándose en información de señalización (por ejemplo, parámetros relacionados con la geometría). La geometría restaurada (o reconstruida) por el decodificador 61003 de geometría se proporciona al decodificador 61004 de atributos. El decodificador 61004 de atributos puede restaurar el atributo realizando el proceso inverso de la operación del codificador 51007 de atributos de la fig. 15 en el flujo de bits de atributos comprimido basándose en la información de señalización (por ejemplo, parámetros relacionados con atributos) y la geometría reconstruida. Según las realizaciones, cuando los datos de la nube de puntos se dividen en mosaicos y/o segmentos en el lado de transmisión como se muestra en la fig. 16, el decodificador 61003 de geometría y el decodificador 61004 de atributos realizan decodificación de geometría y decodificación de atributos en una base de mosaico por mosaico y/o en una base de segmento por segmento.

La fig. 29 es un diagrama de bloques detallado que ilustra otro ejemplo del decodificador 61003 de geometría y del decodificador 61004 de atributos según las realizaciones.

El decodificador 63001 aritmético, la unidad 63002 de reconstrucción de árbol octal, la unidad 63003 de predicción de información de geometría, el procesador 63004 de cuantificación inversa y la unidad 63005 de transformación inversa de coordenadas incluidas en el decodificador 61003 de geometría de la fig. 29 pueden realizar algunas o todas las operaciones del decodificador 11000 aritmético, la unidad 11001 de síntesis de árbol octal, la unidad 11002 de síntesis de aproximación de superficie, la unidad 11003 de reconstrucción de geometría y la unidad 11004 de transformación inversa de coordenadas de la fig. 11, o pueden realizar algunas o todas las operaciones del decodificador 13002 aritmético, el procesador 13003 de reconstrucción de árbol octal basado en código de ocupación, el procesador 13004 de modelo de superficie y el procesador 13005 de cuantificación inversa de la fig. 13. Las posiciones restauradas por el decodificador 61003 de geometría se emiten al postprocesador 61005.

Según las realizaciones, cuando la información sobre el número máximo de predictores que se han de utilizar para la predicción de atributos (lifting_max_num_direct_predictors), la información de umbral para permitir la predicción adaptativa de un atributo (lifting_adaptive_prediction_threshold), la información relacionada con la selección de puntos contiguos y similares se señalizan en al menos uno de un conjunto de parámetros de secuencia (SPS), un conjunto de parámetros de atributo (APS), un conjunto de parámetros de mosaico (TPS) o una cabecera de segmento de atributo, pueden adquirirse y proporcionarse al decodificador 61004 de atributos por el procesador 61002 de señalización, o pueden adquirirse directamente por el decodificador 61004 de atributos.

El decodificador 61004 de atributos, según las realizaciones, puede incluir un decodificador 63006 aritmético, una unidad 63007 de configuración de LOD, una unidad 63008 de configuración de punto contiguo, una unidad 63009

de predicción de información de atributo y un procesador 63010 de cuantificación inversa de información de atributo residual y un procesador 63011 de transformación inversa de color.

El decodificador 63006 aritmético según las realizaciones puede decodificar aritméticamente el flujo de bits de atributos de entrada. El decodificador 63006 aritmético puede decodificar el flujo de bits de atributos basándose en la geometría reconstruida. El decodificador 63006 aritmético realiza una operación y/o decodificación idéntica o similar a la operación y/o decodificación del decodificador 11005 aritmético de la fig. 11 o el decodificador 13007 aritmético de la fig. 13.

Según las realizaciones, el flujo de bits de atributos emitido desde el decodificador 63006 aritmético puede decodificarse mediante una o una combinación de dos o más de decodificación RAHT, decodificación por transformada predictiva basada en LOD y decodificación por transformada de elevación basada en la información de geometría reconstruida.

Se ha descrito como una realización que el dispositivo de transmisión realiza la compresión de atributos por una o una combinación de la codificación por transformada predictiva basada en LOD y la codificación por transformada de elevación. Por consiguiente, se describirá una realización en la que el dispositivo de recepción realiza decodificación de atributos mediante una o una combinación de la decodificación por transformada predictiva basada en LOD y la decodificación por transformada de elevación. Se omitirá la descripción de la decodificación RAHT para el dispositivo de recepción.

Según una realización, un flujo de bits de atributos que se decodifica aritméticamente por el decodificador 63006 aritmético se proporciona a la unidad 63007 de configuración de LOD. Según las realizaciones, el flujo de bits de atributos proporcionado desde el decodificador 63006 aritmético a la unidad 63007 de configuración de LOD puede contener modos de predicción y valores de atributos residuales.

La unidad 63007 de configuración de LOD según las realizaciones genera uno o más LOD de la misma manera o similar que la unidad 53007 de configuración de LOD del dispositivo de transmisión, y emite el uno o más LOD generados a la unidad 63008 de configuración de punto contiguo.

Según las realizaciones, la unidad 63007 de configuración de LOD puede configurar uno o más LOD utilizando uno o más métodos de generación de LOD (o métodos de configuración de LOD). Según las realizaciones, el método de generación de LOD utilizado en la unidad 63007 de configuración de LOD puede proporcionarse a través del procesador 61002 de señalización. Por ejemplo, el método de generación de LOD puede señalizarse en un APS de información de señalización. Según las realizaciones, el método de generación de LOD puede clasificarse en un método de generación de LOD basado en árbol octal, un método de generación de LOD basado en distancia y un método de generación de LOD basado en muestreo.

Según las realizaciones, un grupo que tiene diferentes LOD se denomina conjunto de LOD₁. Aquí, 1 representa un LOD como un número entero que comienza a partir de 0. LOD₀ es un conjunto que consiste en puntos que tienen la mayor distancia entre ellos. A medida que 1 aumenta, la distancia entre puntos que pertenecen a LOD₁ disminuye.

Según las realizaciones, los modos de predicción y los valores de atributos residuales, codificados por el dispositivo de transmisión, pueden proporcionarse para cada LOD o solo para un nodo hoja.

En una realización, cuando el conjunto de LOD₁ se genera por la unidad 63007 de configuración de LOD, la unidad 63008 de configuración de conjunto de puntos contiguos puede buscar puntos contiguos iguales o menores que X (por ejemplo, 3) puntos en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto de LOD₁ y registrar los puntos contiguos buscados como un conjunto de puntos contiguos en el predictor.

Según una realización, la unidad 63008 de configuración de conjunto de puntos contiguos configura el conjunto de puntos contiguos aplicando un intervalo de búsqueda y/o una distancia máxima entre puntos contiguos basándose en la información de señalización.

La unidad 63008 de configuración de conjunto de puntos contiguos según las realizaciones puede obtener la distancia máxima entre puntos contiguos multiplicando una distancia base entre puntos contiguos por NN_range. El NN_range es un intervalo dentro del cual se puede seleccionar un punto contiguo y se denominará intervalo máximo de puntos contiguos, intervalo de puntos contiguos o intervalo de NN.

La unidad 63008 de configuración de conjunto de puntos contiguos según las realizaciones puede establecer el NN_range automática o manualmente según las características del contenido o puede recibirse a través del procesador 61002 de señalización. Por ejemplo, la información sobre el NN_range puede señalizarse en al menos uno de un SPS, un APS, un conjunto de parámetros de mosaico o una cabecera de segmento de atributo de la información de señalización.

La unidad 63008 de configuración de conjunto de puntos contiguos según las realizaciones puede calcular/configurar la distancia base entre puntos contiguos combinando uno o más de un método basado en árbol octal, un método basado en distancia, un método basado en muestreo, un método basado en diferencias promedio de código de Morton para cada LOD, y un método basado en diferencias promedio de distancia para cada LOD. Según las realizaciones, la unidad 63008 de configuración de conjunto de puntos contiguos puede recibir información sobre la distancia base entre puntos contiguos a través del procesador 61002 de señalización, y la información sobre la distancia base entre puntos contiguos puede señalizarse en el APS de la información de señalización.

Dado que una descripción detallada del cálculo/configuración de la distancia base entre puntos contiguos combinando uno o más del método basado en árbol octal, el método basado en distancia, el método basado en muestreo, el método basado en diferencias promedio de código de Morton para cada LOD y el método basado en diferencias promedio de distancia para cada LOD en la unidad 63008 de configuración de conjunto de puntos contiguos según las realizaciones se ha dado en detalle en el proceso descrito anteriormente de codificación de un atributo del dispositivo de transmisión, detalles del mismo se omitirán en la presente memoria.

Según una realización, si la distancia máxima entre puntos contiguos se determina como se ha descrito anteriormente, la unidad 63008 de configuración de conjunto de puntos contiguos busca X (por ejemplo, 3) puntos NN entre puntos dentro del intervalo de búsqueda en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto LOD₁ como en la fig. 26. A continuación, la unidad 63008 de configuración de conjunto de puntos contiguos puede registrar solo puntos NN dentro de la distancia máxima entre puntos contiguos entre los X (por ejemplo, 3) puntos NN como un conjunto de puntos contiguos. Por lo tanto, el número de puntos NN registrados como el conjunto de puntos contiguos es igual o menor que X. En otras palabras, los puntos NN que no están dentro de la distancia máxima entre puntos contiguos entre los X puntos NN no están registrados como el conjunto de puntos contiguos y están excluidos del conjunto de puntos contiguos. Por ejemplo, si dos de los tres puntos NN no están dentro de la distancia máxima entre puntos contiguos, se excluyen los dos puntos NN y sólo se registra el otro punto NN, es decir, un punto NN, como el conjunto de puntos contiguos.

Según otra realización, si la distancia máxima entre puntos contiguos se determina como se ha descrito anteriormente, la unidad 53008 de configuración de conjunto de puntos contiguos puede buscar X (por ejemplo, 3) puntos NN dentro de la distancia máxima entre puntos contiguos entre puntos dentro del intervalo de búsqueda en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto de LOD₁ como en la fig. 27 y registra los X puntos NN como el conjunto de puntos contiguos.

Es decir, el número de puntos NN que pueden registrarse como el conjunto de puntos contiguos puede variar según una temporización (o posición) de aplicación de la distancia máxima entre puntos contiguos. En otras palabras, el número de puntos NN registrados como el conjunto de puntos contiguos puede diferir según si la distancia máxima entre puntos contiguos se aplica después de buscar X puntos NN a través de la búsqueda de puntos contiguos o la distancia máxima entre puntos contiguos se aplica al calcular una distancia entre puntos como en la fig. 27.

Con referencia a la fig. 26 como ejemplo, la unidad 63008 de configuración de conjunto de puntos contiguos compara los valores de distancia entre puntos dentro del intervalo de búsqueda real y el punto Px para buscar X puntos NN y registra solo puntos dentro de la distancia máxima entre puntos contiguos en un LOD al que pertenece el punto Px entre los X puntos como el conjunto de puntos contiguos del punto Px. Es decir, los puntos contiguos registrados como el conjunto de puntos contiguos del punto Px se limitan a puntos dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto Px entre los X puntos.

Con referencia a la fig. 27 como ejemplo, cuando se comparan los valores de distancia entre los puntos dentro del intervalo de búsqueda real y el punto Px, la unidad 63008 de configuración de conjunto de puntos contiguos selecciona X (por ejemplo, 3) puntos NN dentro de la distancia máxima entre puntos contiguos en un LOD al que pertenece el punto Px de entre puntos en el intervalo de búsqueda real y registra los puntos seleccionados como el conjunto de puntos contiguos del punto Px. Es decir, los puntos contiguos registrados como el conjunto de puntos contiguos del punto Px se limitan a puntos dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto Px.

En las figs. 26 y 27, el intervalo de búsqueda real puede ser un valor obtenido multiplicando el valor del campo `lifting_search_range` por 2 y, a continuación, sumando un punto central al valor resultante (es decir, (valor del campo `lifting_search_range` * 2) + punto central), un valor obtenido sumando el valor central al valor del campo `lifting_search_range` (es decir, valor del campo `lifting_search_range` + punto central), o el valor del campo `lifting_search_range`. En una realización, la unidad 63008 de configuración de conjunto de puntos contiguos busca puntos contiguos del punto Px en la lista retenida cuando el número de LOD es plural y busca puntos contiguos del punto Px en la lista de índices cuando el número de LOD es uno.

Por ejemplo, si el número de LOD es 2 o más y la información sobre el intervalo de búsqueda (por ejemplo, el campo `lifting_search_range`) es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto central Pi,

128 puntos después del punto central P_i y el punto central P_i en la lista retenida dispuesta con códigos de Morton. Como otro ejemplo, si el número de LOD es 1 y la información sobre el intervalo de búsqueda es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto central P_i y el punto central P_i en la lista de índices dispuesta con códigos de Morton. Como otro ejemplo, si el número de LOD es 1 y la información sobre el intervalo de búsqueda es 128, el intervalo de búsqueda real incluye 128 puntos antes del punto actual P_x en la lista de índices dispuesta con códigos de Morton.

Como ejemplo, se supone que la unidad 63008 de configuración de conjunto de puntos contiguos selecciona puntos P_2 P_4 P_6 como puntos contiguos de un punto P_3 (es decir, nodo) que pertenece a LOD_1 y registra los puntos seleccionados como el conjunto de puntos contiguos en el predictor de P_3 (véase la fig. 9).

Según las realizaciones, cuando el conjunto de puntos contiguos se registra en cada predictor de puntos que se ha de decodificar en la unidad 63008 de configuración de conjunto de puntos contiguos, la unidad 63009 de predicción de información de atributo predice un valor de atributo de un punto correspondiente a partir de uno o más puntos contiguos registrados en cada predictor. Según las realizaciones, el predictor 63009 de información de atributo realiza un proceso de predicción de un valor de atributo del punto basándose en un modo de predicción del punto. Este proceso de predicción de atributos se realiza en todos los puntos o al menos en algunos puntos de geometría reconstruida.

El modo de predicción de un punto específico según las realizaciones puede ser uno del modo de predicción 0 al modo de predicción 3.

Según las realizaciones, el modo de predicción 0 es un modo que calcula un valor de atributo de predicción a través de una media ponderada, el modo de predicción 1 es un modo que determina un atributo de un primer punto contiguo como el valor de atributo de predicción, el modo de predicción 2 es un modo que determina un atributo de un segundo punto contiguo como el valor de atributo de predicción, y el modo de predicción 3 es un modo que determina un atributo de un tercer punto contiguo como el valor de atributo de predicción.

Según las realizaciones, cuando la diferencia máxima entre valores de atributos de los puntos contiguos registrados en el predictor del punto es menor que un umbral preestablecido, el codificador de atributos del lado de transmisión establece el modo de predicción 0 como el modo de predicción del punto. Cuando la diferencia máxima es mayor o igual que el umbral preestablecido, el codificador de atributo aplica el método RDO a una pluralidad de modos de predicción candidatos, y establece uno de los modos de predicción candidatos como un modo de predicción del punto. En una realización, este proceso se realiza para cada punto.

Según las realizaciones, un modo de predicción (predIndex) del punto seleccionado aplicando el método de RDO puede señalizarse en datos de segmento de atributo. Por consiguiente, el modo de predicción del punto puede obtenerse a partir de los datos de segmento de atributo.

Según las realizaciones, la unidad 63009 de predicción de información de atributo puede predecir el valor de atributo de cada punto basándose en el modo de predicción de cada punto que se establece como se ha descrito anteriormente.

Por ejemplo, cuando se supone que el modo de predicción del punto P_3 es el modo de predicción 0, puede calcularse el promedio de los valores obtenidos multiplicando los atributos de los puntos P_2 , P_4 y P_6 , que son puntos contiguos registrados en el predictor del punto P_3 , por un peso (o peso normalizado). El promedio calculado puede determinarse como el valor de atributo predicho del punto.

Como otro ejemplo, cuando se supone que el modo de predicción del punto P_3 es el modo de predicción 1, el valor de atributo del punto P_4 , que es un punto contiguo registrado en el predictor del punto P_3 , puede determinarse como el valor de atributo predicho del punto P_3 .

Como otro ejemplo, cuando se supone que el modo de predicción del punto P_3 es el modo de predicción 2, el valor de atributo del punto P_6 , que es un punto contiguo registrado en el predictor del punto P_3 , puede determinarse como el valor de atributo predicho del punto P_3 .

Como otro ejemplo, cuando se supone que el modo de predicción del punto P_3 es el modo de predicción 3, el valor de atributo del punto P_2 , que es un punto contiguo registrado en el predictor del punto P_3 , puede determinarse como el valor de atributo predicho del punto P_3 .

Una vez que la unidad 63009 de predicción de información de atributo obtiene el valor de atributo predicho del punto basándose en el modo de predicción del punto, el procesador 63010 de cuantificación inversa de información de atributo residual restaura el valor de atributo del punto añadiendo el valor de atributo predicho del punto predicho mediante el predictor 63009 al valor de atributo residual recibido del punto, y a continuación realiza la cuantificación inversa como un proceso inverso al proceso de cuantificación del dispositivo de transmisión.

En una realización, en el caso en el que el lado de transmisión aplica codificación de longitud de ejecución cero a los valores de atributos residuales de puntos, el procesador 63010 de cuantificación inversa de información de atributo residual realiza decodificación de longitud de ejecución cero sobre los valores de atributos residuales de los puntos, y a continuación realiza cuantificación inversa.

5 Los valores de atributos restaurados por el procesador 63010 de cuantificación inversa de información de atributo residual se envían al procesador 63011 de transformación inversa de color.

El procesador 63011 de transformación inversa de color realiza codificación por transformada inversa para transformación inversa de los valores (o texturas) de color incluidos en los valores de atributos restaurados, y a continuación, emite los atributos al postprocesador 61005. El procesador 63011 de transformación inversa de color realiza una operación y/o codificación por transformada inversa idéntica o similar a la operación y/o codificación por transformada inversa de la unidad 11010 de transformación inversa de color de la fig. 11 o el procesador 13010 de transformación inversa de color de la fig. 13.

10 El postprocesador 61005 puede reconstruir datos de la nube de puntos haciendo coincidir las posiciones restauradas y emitidas por el decodificador 61003 de geometría con los atributos restaurados y emitidos por el decodificador 61004 de atributos. Además, cuando los datos de la nube de puntos reconstruidos están en una unidad de mosaico y/o segmento, el postprocesador 61005 puede realizar un proceso inverso a la división en particiones espacial del lado de transmisión basándose en la información de señalización. Por ejemplo, cuando la zona de delimitación mostrada en (a) de la fig. 16 se divide en mosaicos y segmentos como se muestra en (b) de la fig. 16 y (c) de la fig. 16, los mosaicos y/o segmentos pueden combinarse basándose en la información de señalización para restaurar la zona de delimitación como se muestra en (a) de la fig. 16.

La fig. 30 ilustra un ejemplo de una estructura de flujo de bits de datos de la nube de puntos para la transmisión/recepción según las realizaciones.

La información relacionada puede señalizarse para añadir/realizar las realizaciones descritas anteriormente. La información de señalización según las realizaciones puede utilizarse en el codificador de vídeo de la nube de puntos de una extremidad de transmisión o en el decodificador de vídeo de la nube de puntos de una extremidad de recepción.

El codificador de vídeo de la nube de puntos según las realizaciones puede generar un flujo de bits como se ilustra en la fig. 32 codificando la información de geometría y la información de atributo como se ha descrito anteriormente. Además, la información de señalización sobre los datos de la nube de puntos puede generarse y procesarse por al menos uno del codificador de geometría, el codificador de atributo o el procesador de señalización del codificador de vídeo de la nube de puntos y puede incluirse en un flujo de bits.

La información de señalización según las realizaciones puede ser recibida/obtenida por al menos uno del decodificador de geometría, el decodificador de atributo y el procesador de señalización del decodificador de vídeo de la nube de puntos.

35 El flujo de bits según las realizaciones puede dividirse en un flujo de bits de geometría, un flujo de bits de atributos y un flujo de bits de señalización que se ha de transmitir/recibir o puede combinarse en un flujo de bits y transmitirse/recibirse.

Cuando un flujo de bits de geometría, un flujo de bits de atributos y un flujo de bits de señalización según las realizaciones se configuran como un flujo de bits, el flujo de bits puede incluir uno o más subflujos de bits. El flujo de bits según las realizaciones puede incluir un conjunto de parámetros de secuencia (SPS) para la señalización a nivel de secuencia, un conjunto de parámetros de geometría (GPS) para la señalización de la codificación de información de geometría, uno o más conjuntos de parámetros de atributo (APS) (APS0, APS1) para la señalización de la codificación de información de atributo, un conjunto de parámetros de mosaico (TPS) para la señalización a nivel de mosaico y uno o más segmentos (segmento 0 a segmento n). Es decir, un flujo de bits de datos de la nube de puntos según las realizaciones puede incluir uno o más mosaicos, y cada uno de los mosaicos puede ser un grupo de segmentos que incluyen uno o más segmentos (segmento 0 a segmento n). El TPS según las realizaciones puede contener información sobre cada uno de los uno o más mosaicos (por ejemplo, información de valor de coordenadas e información de altura/tamaño sobre la zona de delimitación). Cada segmento puede incluir un flujo de bits de geometría (Geom0) y uno o más flujos de bits de atributos (Attr0 y Attr1). Por ejemplo, un primer segmento (segmento 0) puede incluir un flujo de bits de geometría (Geom00) y uno o más flujos de bits de atributos (Attr00, Attr10).

El flujo de bits de geometría (o segmento de geometría) en cada segmento puede estar compuesto por una cabecera de segmento de geometría (geom_slice_header) y datos de segmento de geometría (geom_slice_data). Según las realizaciones, geom_slice_header puede incluir información de identificación (geom_parameter_set_id), un identificador de mosaico (geom_tile_id) y un identificador de segmento (geom_slice_id) para un conjunto de parámetros incluido en el GPS, e información (geomBoxOrigin, geom_box_log2_scale, geom_max_node_size_log2, geom_num_points) sobre datos contenidos en los datos de

segmento de geometría (`geom_slice_data`). `geomBoxOrigin` es información de origen de zona de geometría que indica el origen de la zona de los datos de segmento de geometría, `geom_box_log2_scale` es información que indica la escala logarítmica de los datos de segmento de geometría, `geom_max_node_size_log2` es información que indica el tamaño del nodo de árbol octal de geometría raíz, y `geom_num_points` es información relacionada con el número de puntos de los datos de segmento de geometría. Según las realizaciones, el `geom_slice_data` puede incluir información de geometría (o datos de geometría) sobre los datos de la nube de puntos en un segmento correspondiente.

Cada flujo de bits de atributos (o segmento de atributo) en cada segmento puede estar compuesto por una cabecera de segmento de atributo (`attr_slice_header`) y datos de segmento de atributo (`attr_slice_data`). Según las realizaciones, el `attr_slice_header` puede incluir información sobre los datos de segmento de atributo correspondientes. Los datos de segmento de atributo pueden contener información de atributo (o datos de atributo o valor de atributo) sobre los datos de la nube de puntos en el segmento correspondiente. Cuando hay una pluralidad de flujos de bits de atributos en un segmento, cada uno de los flujos de bits puede contener diferente información de atributo. Por ejemplo, un flujo de bits de atributos puede contener información de atributo correspondiente al color, y otro flujo de atributos puede contener información de atributo correspondiente a la reflectancia.

La fig. 31 muestra una estructura de flujo de bits ejemplar para datos de la nube de puntos según las realizaciones.

La fig. 32 ilustra una relación de conexión entre componentes en un flujo de bits de datos de la nube de puntos según las realizaciones.

La estructura de flujo de bits para los datos de la nube de puntos ilustrada en las figs. 31 y 32 puede representar la estructura de flujo de bits para los datos de la nube de puntos mostrada en la fig. 30.

Según las realizaciones, el SPS puede incluir un identificador (`seq_parameter_set_id`) para identificar el SPS, y el GPS puede incluir un identificador (`geom_parameter_set_id`) para identificar el GPS y un identificador (`seq_parameter_set_id`) que indica un SPS activo al que pertenece el GPS. El APS puede incluir un identificador (`attr_parameter_set_id`) para identificar el APS y un identificador (`seq_parameter_set_id`) que indica un SPS activo al que pertenece el APS.

Según las realizaciones, un flujo de bits de geometría (o segmento de geometría) puede incluir una cabecera de segmento de geometría y datos de segmento de geometría. La cabecera de segmento de geometría puede incluir un identificador (`geom_parameter_set_id`) de un GPS activo al que ha de hacer referencia un segmento de geometría correspondiente. Además, la cabecera de segmento de geometría puede incluir adicionalmente un identificador (`geom_slice_id`) para identificar un segmento de geometría correspondiente y/o un identificador (`geom_tile_id`) para identificar un mosaico correspondiente. Los datos de segmento de geometría pueden incluir información de geometría que pertenece a un segmento correspondiente.

Según las realizaciones, un flujo de bits de atributos (o segmento de atributo) puede incluir una cabecera de segmento de atributo y datos de segmento de atributo. La cabecera de segmento de atributo puede incluir un identificador (`attr_parameter_set_id`) de un APS activo al que ha de hacer referencia un segmento de atributo correspondiente y un identificador (`geom_slice_id`) para identificar un segmento de geometría relacionado con el segmento de atributo. Los datos de segmento de atributo pueden incluir información de atributo que pertenece a un segmento correspondiente.

Es decir, el segmento de geometría se refiere al GPS, y el GPS se refiere al SPS. Además, el SPS enumera atributos disponibles, asigna un identificador a cada uno de los atributos e identifica un método de decodificación. El segmento de atributo es hecho corresponder a atributos de salida según el identificador. El segmento de atributo tiene una dependencia del segmento de geometría (decodificado) anterior y el APS. El APS se refiere al SPS.

Según las realizaciones, los parámetros necesarios para la codificación de los datos de la nube de puntos pueden definirse nuevamente en un conjunto de parámetros de los datos de la nube de puntos y/o una cabecera de segmento correspondiente. Por ejemplo, cuando se realiza la codificación de la información de atributo, los parámetros pueden añadirse al APS. Cuando se realiza la codificación basada en mosaico, los parámetros pueden añadirse al mosaico y/o cabecera de segmento.

Como se muestra en las FIGS. 30, 31 y 32, el flujo de bits de los datos de la nube de puntos proporciona mosaicos o segmentos de tal manera que los datos de la nube de puntos pueden dividirse en particiones y procesarse por regiones. Según realizaciones, las regiones respectivas del flujo de bits pueden tener diferentes importancias. Por consiguiente, cuando los datos de la nube de puntos se dividen en mosaicos, se pueden aplicar un filtro diferente (método de codificación) y una unidad de filtro diferente a cada mosaico. Cuando los datos de la nube de puntos se dividen en segmentos, se pueden aplicar un filtro diferente y una unidad de filtro diferente a cada segmento.

Cuando los datos de la nube de puntos se dividen en particiones y se comprimen, el dispositivo de transmisión y el dispositivo de recepción según las realizaciones pueden transmitir y recibir un flujo de bits en una estructura sintáctica de alto nivel para la transmisión selectiva de información de atributo en las regiones divididas.

5 El dispositivo de transmisión según las realizaciones puede transmitir datos de la nube de puntos según la estructura de flujo de bits como se muestra en las figs. 30, 31 y 32. Por consiguiente, puede proporcionarse un método para aplicar diferentes operaciones de codificación y utilizar un método de codificación de buena calidad para una región importante. Además, se puede soportar una codificación y transmisión eficientes según las características de los datos de la nube de puntos, y se pueden proporcionar valores de atributo según los requisitos del usuario.

10 El dispositivo de recepción según las realizaciones puede recibir los datos de la nube de puntos según la estructura de flujo de bits como se muestra en las figs. 30, 31 y 32. Por consiguiente, se pueden aplicar diferentes métodos de filtrado (decodificación) a las regiones respectivas (regiones divididas en mosaicos o en segmentos), en lugar de aplicar un método de decodificación (filtrado) compleja a todos los datos de la nube de puntos. Por lo tanto, se proporciona al usuario una mejor calidad de imagen en una región importante y se puede garantizar una latencia apropiada para el sistema.

15 Como se ha descrito anteriormente, se proporciona un mosaico o un segmento para procesar los datos de la nube de puntos dividiendo en particiones los datos de la nube de puntos por región. Al dividir en particiones los datos de la nube de puntos por región, se puede establecer una opción para generar un conjunto diferente de puntos contiguos para cada región. De este modo, se puede proporcionar un método de selección que tiene baja complejidad y una fiabilidad ligeramente menor, o un método de selección que tiene alta complejidad y alta fiabilidad.

La información de opción relacionada con la selección de punto contiguo para una secuencia requerida en un proceso de codificación/decodificación de información de atributo según las realizaciones puede señalizarse en un SPS y/o un APS.

25 Según las realizaciones, si un mosaico o segmento que tiene diferentes características de atributo está presente en la misma secuencia, la información de opción relacionada con la selección de punto contiguo para la secuencia puede señalizarse en un TPS y/o una cabecera de segmento de atributo para cada segmento.

30 Según las realizaciones, cuando los datos de la nube de puntos se dividen por regiones, las características de atributo de una región específica pueden ser diferentes de las características de atributo de la secuencia y, por lo tanto, pueden configurarse de manera diferente a través de una función de configuración de intervalo máximo de puntos contiguos diferente.

Por lo tanto, cuando los datos de la nube de puntos se dividen por mosaicos, se puede aplicar un intervalo máximo de puntos contiguos diferente a cada mosaico. Además, cuando los datos de la nube de puntos se dividen por segmentos, se puede aplicar un intervalo máximo de puntos contiguos diferente a cada segmento.

35 Según las realizaciones, al menos uno del SPS, el APS, el TPS o la cabecera de segmento de atributo para cada segmento puede incluir la información de opción relacionada con la selección de punto contiguo. Según las realizaciones, la información de opción relacionada con la selección de punto contiguo puede incluir información sobre el intervalo máximo de puntos contiguos (por ejemplo, un campo `nearest_neighbor_max_range`).

40 Según las realizaciones, la información de opción relacionada con la selección de punto contiguo puede incluir además al menos una de información sobre el número máximo de puntos que pueden establecerse como puntos contiguos (por ejemplo, el campo `lifting_num_pred_nearest_neighbours`), información sobre un intervalo de búsqueda (por ejemplo, el campo `lifting_search_range`) e información sobre un método de configuración de LOD o información sobre una distancia base entre puntos contiguos.

45 Un campo, que es el término utilizado en sintaxis de la presente descripción descrito más adelante, puede tener el mismo significado que un parámetro o un elemento.

La fig. 33 muestra una realización de una estructura sintáctica de un conjunto de parámetros de secuencia (SPS) (`seq_parameter_set_rbsp()`) según la presente descripción. El SPS puede incluir información de secuencia sobre un flujo de bits de datos de la nube de puntos. En particular, en este ejemplo, el SPS incluye información de opción relacionada con la selección de punto contiguo.

50 El SPS según las realizaciones puede incluir un campo `profile_idc`, un campo `profile_compatibility_flags`, un campo `level_idc`, un campo `sps_bounding_box_present_flag`, un campo `sps_source_scale_factor`, un campo `sps_seq_parameter_set_id`, un campo `sps_num_attribute_sets` y un campo `sps_extension_present_flag`.

El campo `profile_idc` indica un perfil al que se ajusta el flujo de bits.

El campo `profile_compatibility_flags` igual a 1 puede indicar que el flujo de bits se ajusta al perfil indicado por `profile_idc`.

El campo `level_idc` indica un nivel al que se ajusta el flujo de bits.

5 El campo `sps_bounding_box_present_flag` indica si la información de zona de delimitación de origen se señala en el SPS. La información de la zona de delimitación de origen puede incluir información de desplazamiento y tamaño sobre la zona de delimitación de origen. Por ejemplo, el campo `sps_bounding_box_present_flag` igual a 1 indica que la información de la zona de delimitación de origen se señala en el SPS. El campo `sps_bounding_box_present_flag` igual a 0 indica que la información de la zona de delimitación de origen no se señala. El campo `sps_source_scale_factor` indica el factor de escala de la nube de puntos de origen.

10 El campo `sps_seq_parameter_set_id` proporciona un identificador para el SPS para referencia por otros elementos sintácticos.

El campo `sps_num_attribute_sets` indica el número de atributos codificados en el flujo de bits.

15 El campo `sps_extension_present_flag` especifica si la estructura sintáctica `sps_extension_data` está presente en la estructura sintáctica SPS. Por ejemplo, el campo `sps_extension_present_flag` igual a 1 especifica que la estructura sintáctica `sps_extension_data` está presente en la estructura sintáctica SPS. El campo `sps_extension_present_flag` igual a 0 especifica que esta estructura sintáctica no está presente. Cuando no está presente, se deduce que el valor del campo `sps_extension_present_flag` es igual a 0.

20 Cuando el campo `sps_bounding_box_present_flag` es igual a 1, el SPS según las realizaciones puede incluir además un campo `sps_bounding_box_offset_x`, un campo `sps_bounding_box_offset_y`, un campo `sps_bounding_box_offset_z`, un campo `sps_bounding_box_scale_factor`, un campo `sps_bounding_box_size_width`, un campo `sps_bounding_box_size_height` y un campo `sps_bounding_box_size_depth`.

25 El campo `sps_bounding_box_offset_x` indica el desplazamiento x de la zona de delimitación de origen en las coordenadas cartesianas. Cuando el desplazamiento x de la zona de delimitación de origen no está presente, el valor de `sps_bounding_box_offset_x` es 0.

El campo `sps_bounding_box_offset_y` indica el desplazamiento y de la zona de delimitación de origen en las coordenadas cartesianas. Cuando el desplazamiento y de la zona de delimitación de origen no está presente, el valor de `sps_bounding_box_offset_y` es 0.

30 El campo `sps_bounding_box_offset_z` indica el desplazamiento z de la zona de delimitación de origen en las coordenadas cartesianas. Cuando el desplazamiento z de la zona de delimitación de origen no está presente, el valor de `sps_bounding_box_offset_z` es 0.

El campo `sps_bounding_box_scale_factor` indica el factor de escala de la zona de delimitación de origen en las coordenadas cartesianas. Cuando el factor de escala de la zona de delimitación de origen no está presente, el valor de `sps_bounding_box_scale_factor` puede ser 1.

35 El campo `sps_bounding_box_size_width` indica la anchura de la zona de delimitación de origen en las coordenadas cartesianas. Cuando la anchura de la zona de delimitación de origen no está presente, el valor del campo `sps_bounding_box_size_width` puede ser 1.

40 El campo `sps_bounding_box_size_height` indica la altura de la zona de delimitación de origen en las coordenadas cartesianas. Cuando la altura de la zona de delimitación de origen no está presente, el valor del campo `sps_bounding_box_size_height` puede ser 1.

El campo `sps_bounding_box_size_depth` indica la profundidad de la zona de delimitación de origen en las coordenadas cartesianas. Cuando la profundidad de la zona de delimitación de origen no está presente, el valor del campo `sps_bounding_box_size_depth` puede ser 1.

45 El SPS según las realizaciones incluye una instrucción de iteración repetida tantas veces como el valor del campo `sps_num_attribute_sets`. En una realización, i se inicializa a 0, y se incrementa en 1 cada vez que se ejecuta la instrucción de iteración. La instrucción de iteración se repite hasta que el valor de i se vuelve igual al valor del campo `sps_num_attribute_sets`. La instrucción de iteración puede incluir un campo `attribute_dimension[i]`, un campo `attribute_instance_id[i]`, un campo `attribute_bitdepth[i]`, un campo `attribute_cicp_colour_primaries[i]`, un campo `attribute_cicp_transfer_characteristics[i]`, un campo `attribute_cicp_matrix_coeffs[i]`, un campo `attribute_cicp_video_full_range_flag[i]` y un campo `known_attribute_label_flag[i]`.

50

El campo `attribute_dimension[i]` especifica el número de componentes del atributo i-ésimo.

El campo `attribute_instance_id[i]` especifica el ID de instancia del atributo *i*-ésimo.

El campo `attribute_bitdepth[i]` especifica la profundidad de bit de la *i*-ésima señal(es) de atributo.

El campo `attribute_cicp_colour primaries[i]` indica coordenadas de cromaticidad de los primarios de origen de atributo de color del atributo *i*-ésimo.

5 El campo `attribute_cicp_transfer_characteristics[i]`, bien indica la función característica de transferencia optoelectrónica de referencia del atributo de color en función de una intensidad óptica lineal de entrada de origen con un intervalo nominal de valor real de 0 a 1, o bien, indica la inversa de la función característica de transferencia electroóptica de referencia en función de una intensidad óptica lineal de salida.

10 El campo `attribute_cicp_matrix_coeffs[i]` describe los coeficientes de matriz utilizados para derivar señales de luma y croma de los primarios verde, azul y rojo, o Y, Z y X.

El campo `attribute_cicp_video_full_range_flag[i]` indica el nivel y el intervalo de negro de las señales de luma y croma como se deriva de las señales de componentes de valor real de E'Y, E'PB y E'PR o E'R, E'G y E'B.

15 El campo `known_attribute_label_flag[i]` especifica si un campo `known_attribute_label` o un campo `attribute_label_four_bytes` se señala para el atributo *i*-ésimo. Por ejemplo, el valor del campo `known_attribute_label_flag[i]` igual a 0 especifica que el campo `known_attribute_label` se señala para el atributo *i*-ésimo. El campo `known_attribute_label_flag[i]` igual a 1 especifica que el campo `attribute_label_four_bytes` se señala para el atributo *i*-ésimo.

20 El campo `known_attribute_label[i]` puede especificar un tipo de atributo. Por ejemplo, el campo `known_attribute_label[i]` igual a 0 puede especificar que el atributo *i*-ésimo es color. El campo `known_attribute_label[i]` igual a 1 especifica que el atributo *i*-ésimo es reflectancia. El campo `known_attribute_label[i]` igual a 2 puede especificar que el atributo *i*-ésimo es el índice de trama.

El campo `attribute_label_four_bytes` indica el tipo de atributo conocido con un código de 4 bytes.

En este ejemplo, el campo `attribute_label_four_bytes` indica color cuando es igual a 0 e indica reflectancia cuando es igual a 1.

25 Según las realizaciones, cuando el campo `sps_extension_present_flag` es igual a 1, el SPS puede incluir además un campo `sps_extension_data_flag`.

El campo `sps_extension_data_flag` puede tener cualquier valor.

30 El SPS según las realizaciones puede incluir además la información de opción relacionada con la selección de punto contiguo. Según las realizaciones, la información de opción relacionada con la selección de punto contiguo puede incluir información relacionada con el `NN_range`.

Según las realizaciones, la información de opción relacionada con la selección de punto contiguo puede incluirse en una instrucción de iteración repetida tantas veces como el valor del campo de `sps_num_attribute_sets` descrito anteriormente.

35 Es decir, la instrucción de iteración puede incluir además un campo `nearest_neighbour_max_range[i]` y un campo `nearest_neighbour_min_range[i]`.

40 El campo `nearest_neighbour_max_range[i]` puede especificar un intervalo máximo de puntos contiguos aplicado cuando se comprime el atributo *i*-ésimo de una secuencia correspondiente. Según las realizaciones, el valor del campo `nearest_neighbour_max_range[i]` puede utilizarse como el valor del `NN_range` de la Ecuación 5. Según las realizaciones, el campo `nearest_neighbour_max_range[i]` puede utilizarse para limitar la distancia de un punto registrado como contiguo. Por ejemplo, si se genera un LOD basándose en un árbol octal, el valor del campo `nearest_neighbour_max_range[i]` puede ser el número de nodos de árbol de octal alrededor del punto.

El campo `nearest_neighbour_min_range[i]` puede especificar un intervalo mínimo de puntos contiguos cuando se comprime el atributo *i*-ésimo de la secuencia.

45 La fig. 34 muestra una realización de una estructura sintáctica del conjunto de parámetros de geometría (GPS) (`geometry_parameter_set()`) según la presente descripción. El GPS según las realizaciones puede contener información sobre un método de codificación de información de geometría sobre datos de la nube de puntos contenidos en uno o más segmentos.

50 Según las realizaciones, el GPS puede incluir un campo `gps_geom_parameter_set_id`, un campo `gps_seq_parameter_set_id`, un campo `gps_box_present_flag`, un campo `unique_geometry_points_flag`, un campo `neighbour_context_restriction_flag`, un campo `inferred_direct_coding_mode_enabled_flag`, un campo

bitwise_occupancy_coding_flag, un campo adjacent_child_contextualization_enabled_flag, un campo log2_neighbour_avail_boundary, un campo log2_intra_pred_max_node_size, un campo log2_trisoup_nodo_size y un campo gps_extension_present_flag.

5 El campo gps_geom_parameter_set_id proporciona un identificador para el GPS para referencia por otros elementos sintácticos.

El campo gps_seq_parameter_set_id especifica el valor de sps_seq_parameter_set_id para el SPS activo.

10 El campo gps_box_present_flag especifica si se proporciona información de zona de delimitación adicional en una cabecera de segmento de geometría que referencia el GPS actual. Por ejemplo, el campo gps_box_present_flag igual a 1 puede especificar que se proporciona información de la zona de delimitación adicional en una cabecera de geometría que referencia el GPS actual. Por consiguiente, cuando el campo gps_box_present_flag es igual a 1, el GPS puede incluir además un campo gps_gsh_box_log2_scale_present_flag.

15 El campo gps_gsh_box_log2_scale_present_flag especifica si el campo de gps_gsh_box_log2_scale se señala en cada cabecera de segmento de geometría que referencia el GPS actual. Por ejemplo, gps_gsh_box_log2_scale_present_flag igual a 1 puede especificar que el campo gps_gsh_box_log2_scale se señala en cada cabecera de segmento de geometría que hace referencia al GPS actual. Como otro ejemplo, el campo gps_gsh_box_log2_scale_present_flag igual a 0 puede especificar que el campo gps_gsh_box_log2_scale no se señala en cada cabecera de segmento de geometría y una escala común para todos los segmentos se señala en el campo gps_gsh_box_log2_scale del GPS actual.

20 Cuando el campo gps_gsh_box_log2_scale_present_flag es igual a 0, el GPS puede incluir además un campo gps_gsh_box_log2_scale.

El campo gps_gsh_box_log2_scale indica el factor de escala común del origen de la zona de delimitación para todos los segmentos que se refieren al GPS actual.

25 El campo unique_geometry_points_flag indica si todos los puntos de salida tienen posiciones únicas. Por ejemplo, el campo unique_geometry_points_flag igual a 1 indica que todos los puntos de salida tienen posiciones únicas. El campo unique_geometry_points_flag igual a 0 indica que en todos los segmentos que se refieren al GPS actual, los dos o más de los puntos de salida pueden tener la misma posición.

30 El campo neighbor_context_restriction_flag indica contextos utilizados para la codificación de ocupación de árbol octal. Por ejemplo, el campo neighbour_context_restriction_flag igual a 0 indica que la codificación de ocupación de árbol octal utiliza contextos determinados a partir de seis nodos padre contiguos. El campo neighbour_context_restriction_flag igual a 1 indica que la codificación de ocupación de árbol octal utiliza contextos determinados a partir de nodos hermanos solamente.

35 El campo inferred_direct_coding_mode_enabled_flag indica si el campo direct_mode_flag está presente en la sintaxis de nodo de geometría. Por ejemplo, el campo inferred_direct_coding_mode_enabled_flag igual a 1 indica que el campo direct_mode_flag puede estar presente en la sintaxis de nodo de geometría. Por ejemplo, el campo inferred_direct_coding_mode_enabled_flag igual a 0 indica que el campo direct_mode_flag no está presente en la sintaxis de nodo de geometría.

40 El campo bitwise_occupancy_coding_flag indica si la ocupación del nodo de geometría se codifica utilizando contextualización a nivel de bits del mapa de ocupación del elemento sintáctico. Por ejemplo, el campo bitwise_occupancy_coding_flag igual a 1 indica que la ocupación del nodo de geometría se codifica utilizando contextualización a nivel de bits del elemento sintáctico occupancy_map. Por ejemplo, el campo bitwise_occupancy_coding_flag igual a 0 indica que la ocupación del nodo de geometría se codifica utilizando el elemento sintáctico objection_byte codificado en diccionario.

45 El campo adjacent_child_contextualization_enabled_flag indica si los hijos adyacentes de nodos de árbol octal contiguos se utilizan para la contextualización de ocupación a nivel de bits. Por ejemplo, el campo adjacent_child_contextualization_enabled_flag igual a 1 indica que los hijos adyacentes de nodos de árbol octal contiguos se utilizan para la contextualización de ocupación a nivel de bits. Por ejemplo, adjacent_child_contextualization_enabled_flag igual a 0 indica que los hijos de nodos de árbol octal contiguos no se utilizan para la contextualización de ocupación.

50 El campo log2_neighbour_avail_boundary especifica el valor de la variable NeighbNeighAvailRequest que se utiliza en el proceso de decodificación de la siguiente manera:

$$\text{NeighbAvailBoundary} = 2^{\log_2 \text{neighbour_avail_boundary}}$$

Por ejemplo, cuando el campo `neighbour_context_restriction_flag` es igual a 1, `NeighbMaskMask` puede establecerse igual a 1. Por ejemplo, cuando el campo `neighbour_context_restriction_flag` es igual a 0, `NeighbMaskMask` puede establecerse igual a $1 \ll \log_2 \text{neighbour_avail_boundary}$.

5 El campo `log2_intra_pred_max_node_size` especifica el tamaño de nodo de árbol octal elegible para la intrapredicción de ocupación.

El campo `log2_trisoup_node_size` especifica la variable `TrisoupNodeSize` como el tamaño de los nodos triangulares como sigue.

$\text{TrisoupNodeSize} = 1 \ll \log_2 \text{log2_trisoup_node_size}$

10 El campo `gps_extension_present_flag` especifica si la estructura sintáctica `gps_extension_data` está presente en la estructura sintáctica GPS. Por ejemplo, `gps_extension_present_flag` igual a 1 especifica que la estructura sintáctica `gps_extension_data` está presente en la sintaxis GPS. Por ejemplo, `gps_extension_present_flag` igual a 0 especifica que esta estructura sintáctica no está presente en la sintaxis del GPS.

Cuando el valor del campo `gps_extension_present_flag` es igual a 1, el GPS según las realizaciones puede incluir además un campo `gps_extension_data_flag`.

15 El campo `gps_extension_data_flag` puede tener cualquier valor. Su presencia y valor no afectan a la conformidad del decodificador con los perfiles.

20 La fig. 35 muestra una realización de una estructura sintáctica del conjunto de parámetros de atributo (APS) (`attribute_parameter_set()`) según la presente descripción. El APS según las realizaciones puede contener información sobre un método de codificación de información de atributo en datos de la nube de puntos contenidos en uno o más segmentos. Según las realizaciones, el APS puede incluir información de opción relacionada con la selección de punto contiguo.

El APS según las realizaciones puede incluir un campo `aps_attr_parameter_set_id`, un campo `aps_seq_parameter_set_id`, un campo `attr_coding_type`, un campo `aps_attr_initial_qp`, un campo `aps_attr_chroma_qp_offset`, un campo `aps_slice_qp_delta_present_flag` y un campo `aps_flag`.

25 El campo `aps_attr_parameter_set_id` proporciona un identificador para el APS para referencia por otros elementos sintácticos.

El campo `aps_seq_parameter_set_id` especifica el valor de `sps_seq_parameter_set_id` para el SPS activo.

El campo `attr_coding_type` indica el tipo de codificación para el atributo.

30 En este ejemplo, el campo `attr_coding_type` igual a 0 indica elevación de peso de predicción (o LOD con transformada de predicción) como el tipo de codificación. El campo `attr_coding_type` igual a 1 indica RAHT como el tipo de codificación. El campo `attr_coding_type` igual a 2 indica elevación fija de peso (o LOD con transformada de elevación).

35 El campo `aps_attr_initial_qp` especifica el valor inicial de la variable `SliceQp` para cada segmento que se refiere al APS. El valor inicial de `SliceQp` se modifica en la capa de fragmento de segmento de atributo cuando se decodifica un valor distinto de cero de `slice_qp_delta_luma` o `slice_qp_delta_chroma`.

El campo `aps_attr_chroma_qp_offset` especifica los desplazamientos al parámetro de cuantificación inicial señalado por el `aps_attr_initial_qp` sintáctico.

40 El campo `aps_slice_qp_delta_present_flag` especifica si los elementos sintácticos `ash_attr_qp_delta_luma` y `ash_attr_qp_delta_chroma` están presentes en la cabecera de segmento de atributo (ASH). Por ejemplo, el campo `aps_slice_qp_delta_present_flag` igual a 1 especifica que los elementos sintácticos `ash_attr_qp_delta_luma` y `ash_attr_qp_delta_chroma` están presentes en el ASH. Por ejemplo, el campo `aps_slice_qp_delta_present_flag` igual a 0 especifica que los elementos sintácticos `ash_attr_qp_delta_luma` y `ash_attr_qp_delta_chroma` no están presentes en el ASH.

45 Cuando el valor del campo `attr_coding_type` es 0 o 2, es decir, el tipo de codificación es la elevación de peso de predicción (o LOD con transformada de predicción) o elevación fija de peso (o LOD con transformada de elevación), el APS según las realizaciones puede incluir además un campo `lifting_num_pred_nearest_neighbours`, un campo `lifting_max_num_direct_predictors` 필드 , un campo `lifting_search_range`, un campo `lifting_lod_regular_sampling_enabled_flag`, un campo `lifting_num_detail_levels_minus1`.

50 El campo `lifting_num_pred_nearest_neighbours` especifica el número máximo (es decir, el valor de X) de contiguos más cercanos que se han de utilizar para la predicción.

El campo `lifting_max_num_direct_predictors` especifica el número máximo de predictores que se han de utilizar para la predicción directa. El valor de la variable `MaxNumPredictors` que se utiliza en el proceso de decodificación es el siguiente:

$$\text{MaxNumPredictors} = \text{lifting_max_num_direct_predictors_field} + 1$$

- 5 El campo `lifting_search_range` especifica un intervalo de búsqueda utilizado para determinar los contiguos más cercanos.

El campo `lifting_num_detail_levels_minus1` especifica el número de niveles de detalle para la codificación de atributos.

- 10 El campo `lifting_lod_regular_sampling_enabled_flag` especifica si se construyen niveles de detalle (LOD) utilizando una estrategia de muestreo regular. Por ejemplo, el `lifting_lod_regular_sampling_enabled_flag` igual a 1 especifica que los niveles de detalle (LOD) se construyen utilizando una estrategia de muestreo regular. El `lifting_lod_regular_sampling_enabled_flag` igual a 0 especifica que se utiliza en su lugar una estrategia de muestreo basada en distancia.

- 15 El APS según las realizaciones incluye una instrucción de iteración repetida tantas veces como el valor del campo `lifting_num_detail_levels_minus1`. En una realización, el índice (`idx`) se inicializa a 0 y se incrementa en 1 cada vez que se ejecuta la instrucción de iteración, y la instrucción de iteración se repite hasta que el índice (`idx`) es mayor que el valor del campo `lifting_num_detail_levels_minus1`. Esta instrucción de iteración puede incluir un campo `lifting_sampling_period[idx]` cuando el valor del campo `lifting_lod_decimation_enabled_flag` es verdadero (por ejemplo, 1), y puede incluir un campo `lifting_sampling_distance_squared[idx]` cuando el valor del campo `lod_decimation_enabled_flag` de elevación es falso (por ejemplo, 0).
- 20 El campo `lifting_sampling_period[idx]` especifica el periodo de muestreo para el `idx` de nivel de detalle.

El campo `lifting_sampling_distance_squared[idx]` especifica el cuadrado de la distancia de muestreo para el `idx` de nivel de detalle.

- 25 Cuando el valor del campo `attr_coding_type` es 0, es decir, cuando el tipo de codificación es elevación de peso de predicción (o LOD con transformada de predicción), el APS según las realizaciones puede incluir además un campo `lifting_adaptive_prediction_threshold` y un campo `lifting_intra_lod_prediction_num_layers`.

El campo `lifting_adaptive_prediction_threshold` especifica el umbral para permitir la predicción adaptativa.

- 30 El campo `lifting_intra_lod_prediction_num_layers` especifica el número de capas de LOD donde se podría hacer referencia a los puntos decodificados en la misma capa de LOD para generar un valor de predicción de un punto objetivo. Por ejemplo, el campo `lifting_intra_lod_prediction_num_layers` igual a `num_detail_levels_minus1` más 1 indica que el punto objetivo podría referirse a puntos decodificados en la misma capa de LOD para todas las capas de LOD. Por ejemplo, el campo `lifting_intra_lod_prediction_num_layers` igual a 0 indica que el punto objetivo no podría referirse a puntos decodificados en la misma capa de LoD para ninguna capa de LoD.

- 35 El campo `aps_extension_present_flag` especifica si la estructura sintáctica `aps_extension_data` está presente en la estructura sintáctica APS. Por ejemplo, el campo `aps_extension_present_flag` igual a 1 especifica que la estructura sintáctica `aps_extension_data` está presente en la estructura sintáctica APS. Por ejemplo, el campo `aps_extension_present_flag` igual a 0 especifica que esta estructura sintáctica no está presente en la estructura sintáctica APS.

- 40 Cuando el valor del campo `aps_extension_present_flag` es 1, el APS según las realizaciones puede incluir además un campo `aps_extension_data_flag`.

El campo `aps_extension_data_flag` puede tener cualquier valor. Su presencia y valor no afectan a la conformidad del decodificador con los perfiles.

El APS según las realizaciones puede incluir además la información de opción relacionada con la selección de punto contiguo.

- 45 Cuando el valor del campo `attr_coding_type` es 0 o 2, es decir, cuando un tipo de codificación es elevación de peso de predicción o LOD con transformada de predicción, o elevación fija de peso o LOD con transformada de elevación, el APS según las realizaciones puede incluir además un campo `nearest_neighbour_max_range`, un campo `nearest_neighbour_min_range` y un campo `different_nn_range_in_tile_flag`.

- 50 El campo `nearest_neighbour_max_range` puede especificar un intervalo máximo de puntos contiguos. Según las realizaciones, el campo `nearest_neighbour_max_range` puede utilizarse para limitar la distancia de un punto registrado como contiguo. Según las realizaciones, el valor del campo `nearest_neighbour_max_range` puede utilizarse como el valor del `NN_range` de la Ecuación 5. Por ejemplo, si se genera un LOD basándose en un árbol

octal, el valor del campo `nearest_neighbour_max_range` puede ser el número de nodos de árbol octal alrededor del punto.

El campo `nearest_neighbour_min_range` puede especificar un intervalo mínimo de puntos contiguos cuando se comprime un atributo.

- 5 El campo `different_nn_range_in_tile_flag` puede especificar si una secuencia correspondiente utiliza un intervalo máximo/mínimo de puntos contiguos diferente para un mosaico dividido de la secuencia.

Según las realizaciones, la información sobre el número máximo que puede establecerse como un punto contiguo (por ejemplo, el campo `lifting_num_pred_nearest_neighbours`) y la información sobre un intervalo de búsqueda (por ejemplo, el campo `lifting_search_range`) también pueden incluirse en la información de opción relacionada con la selección de punto contiguo.

- 10 Según las realizaciones, la información de opción relacionada con la selección de punto contiguo puede incluir además al menos una de información sobre un método de configuración de LOD y/o información sobre una distancia base entre puntos contiguos.

- 15 La fig. 36 ilustra otra realización de una estructura sintáctica de un conjunto de parámetros de atributo (APS) (`attribute_parameter_set()`) según las realizaciones. El APS según las realizaciones puede incluir información sobre un método de codificación de información de atributo de datos de la nube de puntos incluidos en uno o más segmentos. En particular, se ilustra un ejemplo que incluye la información de opción relacionada con la selección de punto contiguo.

- 20 La estructura sintáctica del APS (`attribute_parameter_set()`) de la fig. 36 es la misma o similar que la estructura sintáctica del APS (`attribute_parameter_set()`) de la fig. 35, excepto por la información de opción relacionada con la selección de punto contiguo. Por consiguiente, para las porciones omitidas o no descritas en la descripción de la fig. 36, véase la fig. 35.

Un campo `aps_attr_parameter_set_id` especifica un ID del APS para referencia por otros elementos sintácticos.

Un campo `aps_seq_parameter_set_id` especifica el valor de `sps_seq_parameter_set_id` para un SPS activo.

- 25 Un campo `attr_coding_type` representa un tipo de codificación para un atributo.

En una realización, si el valor del campo `attr_coding_type` es 0, el tipo de codificación puede indicar la elevación de peso de predicción o un LOD con transformada de predicción y, si el valor del campo `attr_coding_type` es 1, el tipo de codificación puede indicar RAHT. Si el valor del campo `attr_coding_type` es 2, el tipo de codificación puede indicar elevación fija de peso o un LOD con transformada de elevación.

- 30 Cuando el valor del campo `attr_coding_type` es 0 o 2, es decir, cuando el tipo de codificación es elevación de peso de predicción o LOD con transformada de predicción, o elevación fija de peso o LOD con transformada de elevación, el APS según las realizaciones puede incluir además un campo `lifting_num_pred_nearest_neighbours`, un campo `lifting_max_num_direct_predictors`, un campo `lifting_search_range`, un campo `lifting_lod_regular_sampling_enabled_flag` y un campo `lifting_num_detail_levels_minus1`,

- 35 El campo `lifting_num_pred_nearest_neighbours` representa el número máximo de NN (es decir, el valor de X) que se ha de utilizar para la predicción.

El campo `lifting_max_num_direct_predictors` indica el número máximo de predictores que se ha de utilizar para la predicción directa. El valor de una variable `MaxNumPredictors` utilizada en un proceso de decodificación de datos de la nube de puntos según las realizaciones puede representarse como sigue.

- 40
$$\text{MaxNumPredictors} = \text{lifting_max_num_direct_predictors field} + 1$$

El campo `lifting_search_range` indica un intervalo de búsqueda utilizado para determinar los NN.

El campo `lifting_lod_regular_sampling_enabled_flag` indica si se construye un LOD utilizando una estrategia de muestreo regular. Por ejemplo, el valor del campo `lifting_lod_regular_sampling_enabled_flag` igual a 1 especifica que los LOD se construyen utilizando la estrategia de muestreo regular, y el valor del campo `lifting_lod_regular_sampling_enabled_flag` igual a 0 especifica que se utiliza en su lugar una estrategia de muestreo basada en distancia.

- 45 El campo `lifting_lod_regular_sampling_enabled_flag` igual a 0 especifica que se utiliza en su lugar una estrategia de muestreo basada en distancia.

El campo `lifting_num_detail_levels_minus1` especifica el número de LOD para la codificación de atributos.

El APS según las realizaciones incluye una instrucción de iteración repetida tantas veces como el valor del campo `lifting_num_detail_levels_minus1`. En una realización, un índice `idx` se inicializa a 0 y aumenta en 1 cada vez que se ejecuta la instrucción de iteración. La instrucción de iteración se repite hasta que el índice `idx` es

- 50

mayor que el valor del campo `lifting_num_detail_levels_minus1`. Esta instrucción de iteración puede incluir un campo `lifting_sampling_period[idx]` cuando el valor de un campo `lifting_lod_decimation_enabled_flag` es verdadero (por ejemplo, 1) y puede incluir un campo `lifting_sampling_distance_squared[idx]` cuando el valor del campo `lifting_lod_decimation_enabled_flag` es falso (por ejemplo, 0).

5 El campo `lifting_sampling_period[idx]` representa un periodo de muestreo para un `idx` de LOD.

El campo `lifting_sampling_distance_squared[idx]` especifica el cuadrado de una distancia de muestreo para el `idx` de LOD.

10 Cuando el valor del campo `attr_coding_type` es 0, es decir, cuando el tipo de codificación es elevación de peso de predicción o un LOD con transformada de predicción, el APS según las realizaciones puede incluir además un campo `lifting_adaptive_prediction_threshold` y un campo `lifting_intra_lod_prediction_num_layers`,

El campo `lifting_adaptive_prediction_threshold` especifica un umbral para permitir la predicción adaptativa.

El campo `lifting_intra_lod_prediction_num_layers` especifica el número de capas de LOD a las que los puntos decodificados en la misma capa de LOD pueden referirse para generar un valor de predicción de un punto objetivo.

15 El APS según las realizaciones puede incluir además la información de opción relacionada con la selección de punto contiguo.

20 Cuando el valor del campo `attr_coding_type` es 0 o 2, es decir, cuando el tipo de codificación es elevación de peso de predicción o LOD con transformada de predicción, o elevación fija de peso o LOD con transformada de elevación, el APS según las realizaciones puede incluir además un campo `different_nn_range_in_tile_flag` y un campo `different_nn_range_per_lod_flag`.

El campo `different_nn_range_in_tile_flag` puede especificar si una secuencia correspondiente utiliza un intervalo máximo/mínimo de puntos contiguos diferente para un mosaico dividido de la secuencia.

El campo `different_nn_range_per_lod_flag` puede especificar si utilizar un intervalo máximo/mínimo de puntos contiguos diferente para cada LOD.

25 Por ejemplo, cuando el valor del campo `different_nn_range_per_lod_flag` es falso, el APS puede incluir además un campo `nearest_neighbour_max_range` y un campo `nearest_neighbour_min_range`.

30 El campo `nearest_neighbour_max_range` puede especificar un intervalo máximo de puntos contiguos. Según las realizaciones, el campo `nearest_neighbour_max_range` puede utilizarse para limitar la distancia de un punto registrado como contiguo. Según las realizaciones, el valor del campo `nearest_neighbour_max_range` puede utilizarse como el valor del `NN_range` de la Ecuación 5.

El campo `nearest_neighbour_min_range` puede especificar un intervalo mínimo de puntos contiguos.

35 Por ejemplo, si el valor del campo `different_nn_range_per_lod_flag` es verdadero, el APS incluye además una instrucción de iteración repetida tantas veces como el valor del campo `lifting_num_detail_levels_minus1`. En una realización, un índice `idx` se inicializa a 0 y aumenta en 1 cada vez que se ejecuta la instrucción de iteración. La instrucción de iteración se repite hasta que el índice `idx` es mayor que el valor del campo `lifting_num_detail_levels_minus1`. Esta instrucción de iteración puede incluir un campo `nearest_neighbour_max_range[idx]` y un campo `nearest_neighbour_min_range[idx]`.

40 El campo `nearest_neighbour_max_range[idx]` puede especificar un intervalo máximo de puntos contiguos para un `idx` de LOD. Según las realizaciones, el campo `nearest_neighbour_max_range[idx]` puede utilizarse para limitar la distancia de un punto registrado como contiguo para el `idx` de LOD. Según las realizaciones, el valor del campo `nearest_neighbour_max_range[idx]` puede utilizarse como el valor de `NN_range` para el `idx` de LOD.

El campo `nearest_neighbour_min_range[idx]` puede indicar un intervalo mínimo de puntos contiguos para el `idx` de LOD.

45 Según las realizaciones, la información sobre un número máximo que puede establecerse como un punto contiguo (por ejemplo, el campo `lifting_num_pred_nearest_neighbours`) y la información sobre un intervalo de búsqueda (por ejemplo, el campo `lifting_search_range`) también pueden incluirse en la información de opción relacionada con la selección de punto contiguo.

50 Según las realizaciones, la información de opción relacionada con la selección de punto contiguo puede incluir además al menos una de información sobre el método de configuración de LOD y/o información sobre la distancia base entre puntos contiguos.

La fig. 37 muestra una realización de una estructura sintáctica de un conjunto de parámetros de mosaico (TPS) (tile_parameter_set()) según la presente descripción. Según las realizaciones, un TPS puede denominarse inventario de mosaicos. El TPS según las realizaciones incluye información relacionada con cada mosaico. En particular, en este ejemplo, el TPS incluye información de opción relacionada con la selección de punto contiguo.

5 El TPS según las realizaciones incluye un campo num_tiles.

El campo num_tiles indica el número de mosaicos señalizados para el flujo de bits. Cuando no está presente, se deduce que num_tiles es 0.

10 El TPS según las realizaciones incluye una instrucción de iteración repetida tantas veces como el valor del campo num_tiles. En una realización, i se inicializa a 0, y se incrementa en 1 cada vez que se ejecuta la instrucción de iteración. La instrucción de iteración se repite hasta que el valor de i se vuelve igual al valor del campo num_tiles. La instrucción de iteración puede incluir un campo tile_bounding_box_offset_x[i], un campo tile_bounding_box_offset_y[i], un campo tile_bounding_box_offset_z[i], un campo tile_bounding_box_size_width[i], un campo tile_bounding_box_size_height[i] y un campo tile_bounding_box_size_depth[i].

15 El campo tile_bounding_box_offset_x[i] indica el desplazamiento x del i-ésimo mosaico en las coordenadas cartesianas.

El campo tile_bounding_box_offset_y[i] indica el desplazamiento y del i-ésimo mosaico en las coordenadas cartesianas.

20 El campo tile_bounding_box_offset_z[i] indica el desplazamiento z del i-ésimo mosaico en las coordenadas cartesianas.

El campo tile_bounding_box_size_width[i] indica la anchura del i-ésimo mosaico en las coordenadas cartesianas.

El campo tile_bounding_box_size_height[i] indica la altura del i-ésimo mosaico en las coordenadas cartesianas.

El campo tile_bounding_box_size_depth[i] indica la profundidad del i-ésimo mosaico en las coordenadas cartesianas.

25 El TPS según las realizaciones puede incluir además la información de opción relacionada con la selección de punto contiguo.

Según las realizaciones, la información de opción relacionada con la selección de punto contiguo puede incluirse en una instrucción de iteración repetida tantas veces como el valor del campo num_tiles como sigue.

30 En una realización, si el valor del campo different_nn_range_in_tile_flag es verdadero, la instrucción de iteración puede incluir además un campo nearest_neighbour_max_range [i], un campo nearest_neighbour_min_range [i] y un campo different_nn_range_in_slice_flag [i]. En una realización, el campo different_nn_range_in_tile_flag se señala en el APS.

35 El campo nearest_neighbour_max_range [i] puede especificar un intervalo máximo de puntos contiguos del i-ésimo mosaico. Según las realizaciones, el campo nearest_neighbour_max_range [i] puede utilizarse para limitar la distancia de un punto registrado como contiguo en el i-ésimo mosaico. Según realizaciones, el valor del campo nearest_neighbour_max_range [i] puede utilizarse como el valor del NN_range del i-ésimo mosaico.

El campo nearest_neighbour_min_range [i] puede especificar un intervalo mínimo de puntos contiguos del i-ésimo mosaico.

40 El campo different_nn_range_in_slice_flag [i] puede especificar si un mosaico correspondiente utiliza un intervalo máximo/mínimo de puntos contiguos diferente para un segmento dividido del mosaico.

Si el valor del campo different_nn_range_in_slice_flag [i] es verdadero, el TPS puede incluir además un campo nearest_neighbour_offset_range_in_slice_flag[i].

45 El campo nearest_neighbour_offset_range_in_slice_flag[i] puede indicar si el intervalo máximo/mínimo de puntos contiguos definido en el segmento está marcado como un desplazamiento de intervalo en el intervalo máximo/mínimo de puntos contiguos definido en el mosaico o como un valor absoluto.

La fig. 38 muestra una realización de una estructura sintáctica de un flujo de bits() de segmento de geometría según la presente descripción.

50 El flujo de bits de segmentos de geometría (geometry_slice_bitstream()) según las realizaciones puede incluir una cabecera de segmentos de geometría (geometry_slice_header()) y datos de segmento de geometría (geometry_slice_data()).

La fig. 39 muestra una realización de una estructura sintáctica de la cabecera de segmento de geometría (`geometry_slice_header()`) según la presente descripción.

Un flujo de bits transmitido por el dispositivo de transmisión (o un flujo de bits recibido por el dispositivo de recepción) según las realizaciones puede contener uno o más segmentos. Cada segmento puede incluir un segmento de geometría y un segmento de atributo. El corte de geometría incluye una cabecera de segmento de geometría (GSH). El segmento de atributo incluye una cabecera de segmento de atributo (ASH).

La cabecera de segmento de geometría (`geometry_slice_header()`) según las realizaciones puede incluir un campo `gsh_geom_parameter_set_id`, un campo `gsh_tile_id`, un campo `gsh_slice_id`, un campo `gsh_max_node_size_log2`, un campo `gsh_num_points` y un campo `byte_assignment()`.

Cuando el valor del campo `gps_box_present_flag` incluido en el GPS es "verdadero" (por ejemplo, 1), y el valor del campo `gps_gsh_box_log2_scale_present_flag` es "verdadero" (por ejemplo, 1), la cabecera de segmento de geometría (`geometry_slice_header()`) según las realizaciones puede incluir además un campo `gsh_box_log2_scale`, un campo `gsh_box_origin_x`, un campo `gsh_box_origin_y` y un campo `gsh_box_origin_z`.

El campo `gsh_geom_parameter_set_id` especifica el valor del `gps_geom_parameter_set_id` del GPS activo.

El campo `gsh_tile_id` especifica el valor del id de mosaico al que se hace referencia por el GSH.

El `gsh_slice_id` especifica el id del segmento para referencia por otros elementos sintácticos.

El campo `gsh_box_log2_scale` especifica el factor de escala del origen de la zona de delimitación para el segmento.

El campo `gsh_box_origin_x` especifica el valor x del origen de la zona de delimitación de escalado por el valor del campo `gsh_box_log2_scale`.

El campo `gsh_box_origin_y` especifica el valor y del origen de la zona de delimitación escalado por el valor del campo `gsh_box_log2_scale`.

El campo `gsh_box_origin_z` especifica el valor z del origen de la zona de delimitación escalado por el valor del campo `gsh_box_log2_scale`.

El campo `gsh_max_node_size_log2` especifica un tamaño de un nodo de árbol octal de geometría raíz.

El campo `gsh_points_number` especifica el número de puntos codificados en el segmento.

La fig. 40 muestra una realización de una estructura sintáctica de datos de segmento de geometría (`geometry_slice_data()`) según la presente descripción. Los datos de segmento de geometría (`geometry_slice_data()`) según las realizaciones pueden transportar un flujo de bits de geometría que pertenece a un segmento correspondiente.

El `geometry_slice_data()` según las realizaciones puede incluir una primera instrucción de iteración repetida tantas veces como por el valor de `MaxGeometryOctreeRequest`. En una realización, la profundidad se inicializa a 0 y se incrementa en 1 cada vez que se ejecuta la instrucción de iteración, y la primera instrucción de iteración se repite hasta que la profundidad se vuelve igual a `MaxGeometryOctreeRequest`. La primera instrucción de iteración puede incluir una segunda instrucción de bucle repetida tantas veces como el valor de `NumNodesAtDepth`. En una realización, `nodeidx` se inicializa a 0 y se incrementa en 1 cada vez que se ejecuta la instrucción de iteración. La segunda instrucción de iteración se repite hasta que `nodeidx` se vuelve igual a `NumNodesAtDepth`. La segunda instrucción de iteración puede incluir $xN = \text{NodeX}[\text{depth}][\text{nodeidx}]$, $yN = \text{NodeY}[\text{depth}][\text{nodeidx}]$, $zN = \text{NodeZ}[\text{depth}][\text{nodeidx}]$, y `geometry_node` (profundidad, `nodeidx`, xN , yN , zN). `MaxGeometryOctreeDepth` indica el valor máximo de la profundidad de árbol octal de geometría, y `NumNodesAtDepth[depth]` indica el número de nodos que se ha de decodificar en la profundidad correspondiente. Las variables `NodeX[depth][nodeidx]`, `NodeY[depth][nodeidx]` y `NodeZ[depth][nodeidx]` indican las coordenadas x, y, z del `nodeidx`-ésimo en orden de decodificación a una profundidad dada. El flujo de bits de geometría del nodo de la profundidad se transmite a través de `geometry_node` (profundidad, `nodeidx`, xN , yN , zN).

Los datos de segmento de geometría (`geometry_slice_data()`) según las realizaciones pueden incluir además `geometry_trisoup_data()` cuando el valor del campo `log2_trisoup_node_size` es mayor que 0. Es decir, cuando el tamaño de los nodos triangulares es mayor que 0, un flujo de bits de geometría sometido a codificación de geometría trisoup se transmite a través de `geometry_trisoup_data()`.

La fig. 41 muestra una modalidad de una estructura sintáctica de `attribute_slice_bitstream()` según la presente descripción.

El flujo de bits de segmento de atributo (`attribute_slice_bitstream()`) según las realizaciones puede incluir una cabecera de segmento de atributo (`attribute_slice_header()`) y datos de segmento de atributo (`attribute_slice_data()`).

- 5 La fig. 42 muestra una realización de una estructura sintáctica de una cabecera de segmento de atributo (`attribute_slice_header()`) según la presente descripción. La cabecera de segmento de atributo según las realizaciones incluye información de señalización para un segmento de atributo correspondiente. En particular, en este ejemplo, la cabecera de segmento de atributo incluye información de opción relacionada con la selección de punto contiguo.
- 10 La cabecera de segmento de atributo (`attribute_slice_header()`) según las realizaciones puede incluir un campo `ash_attr_parameter_set_id` un campo `ash_attr_sps_attr_idx` y un campo `ash_attr_geom_slice_id`.
- Cuando el valor del campo `aps_slice_qp_delta_present_flag` del APS es "verdadero" (por ejemplo, 1), la cabecera de segmento de atributo (`attribute_slice_header()`) según las realizaciones puede incluir además un campo `ash_qp_delta_luma` y un campo `ash_qp_delta_chroma`.
- 15 El campo `ash_attr_parameter_set_id` especifica un valor del campo `aps_attr_parameter_set_id` del APS activo actual (por ejemplo, el campo `aps_attr_parameter_set_id` incluido en el APS descrito en la fig. 35 o la fig. 36).
- El campo `ash_attr_sps_attr_idx` identifica un conjunto de atributos en el SPS activo actual. El valor del campo `ash_attr_sps_attr_idx` está en el intervalo de 0 al campo `sps_num_attribute_sets` incluido en el SPS activo actual.
- El campo `ash_attr_geom_slice_id` especifica el valor del campo `gsh_slice_id` de la cabecera de segmento de geometría actual.
- 20 El campo `ash_qp_delta_luma` especifica un parámetro de cuantificación delta de luma (qp) derivado del segmento inicial qp en el conjunto de parámetros de atributo activo.
- El campo `ash_qp_delta_chroma` especifica el delta de croma qp derivado del segmento inicial qp en el conjunto de parámetros de atributo activo.
- 25 Una cabecera de segmento de atributo (`attribute_slice_header()`) según las realizaciones puede incluir además la información de opción relacionada con la selección de punto contiguo de la siguiente manera.
- En una realización, si el valor del campo `different_nn_range_in_slice_flag` es verdadero y el valor del campo `nearest_neighbour_offset_range_in_slice_flag` es falso, la cabecera de segmento de atributo puede incluir además un campo `nearest_neighbour_absolute_max_range` y un campo `nearest_neighbour_absolute_min_range`.
- 30 El campo `nearest_neighbour_absolute_max_range` puede especificar un intervalo máximo de puntos contiguos cuando se comprime un atributo de un segmento correspondiente. Según las realizaciones, el campo `nearest_neighbour_absolute_max_range` puede utilizarse para limitar la distancia de un punto registrado como contiguo en el segmento. Según las realizaciones, el valor del campo `nearest_neighbour_absolute_max_range` puede utilizarse como el valor del `NN_range` del segmento.
- 35 El campo `nearest_neighbour_absolute_min_range` puede indicar un intervalo mínimo de puntos contiguos cuando se comprime un atributo del segmento.
- En una realización, el campo `different_nn_range_in_slice_flag` y el campo `nearest_neighbour_offset_range_in_slice_flag` se señalizan en el TPS.
- 40 En una realización, si el valor del campo `different_nn_range_in_slice_flag` es verdadero y el valor del campo `nearest_neighbour_offset_range_inslice_flag` es verdadero, la cabecera de segmento de atributo puede incluir además un campo `nearest_neighbour_max_range_offset` y un campo `nearest_neighbour_min_range_offset`.
- El campo `nearest_neighbour_max_range_offset` puede especificar un desplazamiento de intervalo máximo de puntos contiguos cuando se comprime un atributo del segmento. En una realización, la referencia del desplazamiento de intervalo máximo de puntos contiguos es un intervalo máximo de puntos contiguos de un mosaico al que pertenece el segmento. Según las realizaciones, el campo `nearest_neighbour_max_range_offset` puede utilizarse para limitar la distancia de un punto registrado como contiguo en el segmento. Según las realizaciones, el valor del campo `nearest_neighbour_max_range_offset` puede utilizarse como un desplazamiento del valor del `NN_range` del segmento.
- 45 El campo `nearest_neighbour_min_range_offset` puede indicar un desplazamiento de intervalo mínimo de puntos contiguos cuando se comprime un atributo del segmento. En una realización, la referencia del desplazamiento de intervalo mínimo de puntos contiguos es un intervalo mínimo de puntos contiguos de un mosaico al que pertenece el segmento.
- 50

La fig. 43 ilustra otra realización de una estructura sintáctica de una cabecera de segmento de atributo (`attribute_slice_header()`) según la presente descripción. La cabecera de segmento de atributo según las realizaciones incluye información de señalización para un segmento de atributo correspondiente. En particular, se ilustra un ejemplo que incluye la información de opción relacionada con la selección de punto contiguo.

5 La estructura sintáctica de la cabecera de segmento de atributo de la fig. 43 es la misma o similar a la estructura sintáctica de la cabecera de segmento de atributo de la fig. 42 excepto por la información de opción sobre la selección de punto contiguo. Por consiguiente, para las porciones que se omiten o no se describen en la descripción de la fig. 43, véase la fig. 42.

10 El campo `ash_attr_parameter_set_id` representa el valor de un campo `aps_attr_parameter_set_id` de un APS activo actual (por ejemplo, el campo `aps_attr_parameter_set_id` incluido en el APS descrito con referencia a la fig. 35 o la fig. 36).

El campo `ash_attr_sps_attr_idx` identifica un conjunto de atributos en un SPS activo actual. El valor del campo `ash_attr_sps_attr_idx` está en el intervalo de 0 a un campo `sps_num_attribute_sets` incluido en el SPS activo actual.

15 El campo `ash_attr_geom_slice_id` especifica el valor de un campo `gsh_slice_id` de una cabecera de segmento de geometría actual.

La cabecera de segmento de atributo (`attribute_slice_header()`) según las realizaciones puede incluir, además, la información de opción relacionada con la selección de punto contiguo de la siguiente manera.

20 En una realización, si el valor del campo `different_nn_range_in_slice_flag` es verdadero, la cabecera de segmento de atributo puede incluir además un campo `different_nn_range_per_lod_flag`.

El campo `different_nn_range_per_lod_flag` puede especificar si utilizar de manera diferente un intervalo máximo/mínimo de puntos contiguos para cada LOD.

25 Por ejemplo, si el valor del campo `different_nn_range_per_lod_flag` es falso y el valor del campo `nearest_neighbour_offset_range_in_slice_flag` es falso, la cabecera de segmento de atributo puede incluir además un campo `nearest_neighbour_neighbour_absolute_max_range` y un campo `nearest_neighbour_absolute_min_range`.

30 El campo `nearest_neighbour_absolute_max_range` puede especificar un intervalo máximo de puntos contiguos cuando se comprime un atributo de un segmento correspondiente. Según las realizaciones, el campo `nearest_neighbour_absolute_max_range` puede utilizarse para limitar la distancia de un punto registrado como contiguo en el segmento. Según realizaciones, el valor del campo `nearest_neighbour_absolute_max_range` puede utilizarse como el valor del `NN_range` del segmento.

El campo `nearest_neighbour_absolute_min_range` puede especificar un intervalo mínimo de puntos contiguos cuando se comprime un atributo del segmento.

35 Por ejemplo, si el valor del campo `different_nn_range_per_lod_flag` es falso y el valor del campo `nearest_neighbour_offset_range_in_slice_flag` es verdadero, la cabecera de segmento de atributo puede incluir además un campo `nearest_neighbour_max_range_offset` y un campo `nearest_neighbour_min_range_offset`.

40 El campo `nearest_neighbour_max_range_offset` puede especificar un desplazamiento de intervalo máximo de puntos contiguos cuando se comprime un atributo del segmento. En una realización, la referencia del desplazamiento de intervalo máximo de puntos contiguos es un intervalo máximo de puntos contiguos de un mosaico al que pertenece el segmento. Según las realizaciones, el campo `nearest_neighbour_absolute_max_range` puede utilizarse para limitar la distancia de un punto registrado como contiguo en el segmento. Según realizaciones, el valor del campo `nearest_neighbour_absolute_max_range` puede utilizarse como un desplazamiento del valor del `NN_range` del segmento.

45 El campo `nearest_neighbour_min_range_offset` puede especificar un desplazamiento de intervalo mínimo de puntos contiguos cuando se comprime un atributo del segmento. En una realización, la referencia del desplazamiento de intervalo mínimo de puntos contiguos es un intervalo mínimo de puntos contiguos de un mosaico al que pertenece el segmento.

50 Por ejemplo, si el valor del campo `different_nn_range_per_lod_flag` es verdadero y el valor del campo `nearest_neighbour_offset_range_in_slice_flag` es falso, la cabecera de segmento de atributo incluye además una instrucción de iteración repetida tantas veces como el valor del campo `lifting_num_detail_levels_minus1`. En una realización, un índice (`idx`) se inicializa a 0 y aumenta en 1 cada vez que se ejecuta la instrucción de iteración. La instrucción de iteración se repite hasta que el índice `idx` es mayor que el valor del campo `lifting_num_detail_levels_minus1`. Esta instrucción de iteración puede incluir un campo `nearest_neighbour_absolute_max_range[idx]` y un campo `nearest_neighbour_absolute_min_range[idx]`.

En una realización, el campo `lifting_num_detail_levels_minus1` indica el número de LOD para la codificación de atributos y se señala en el APS.

El campo `nearest_neighbour_absolute_max_range[idx]` puede indicar un intervalo máximo de puntos contiguos para un `idx` de LOD cuando se comprime un atributo del segmento. Según las realizaciones, el campo `nearest_neighbour_absolute_max_range [idx]` puede utilizarse para limitar la distancia de un punto registrado como contiguo para el `idx` de LOD en el segmento. Según las realizaciones, el valor del campo `nearest_neighbour_absolute_max_range [idx]` puede utilizarse como el valor de `NN_range` para el `idx` de LOD de un segmento correspondiente.

El campo `nearest_neighbour_absolute_min_range [idx]` puede indicar un intervalo mínimo de puntos contiguos para el `idx` de LOD cuando se comprime un atributo del segmento.

Por ejemplo, si el valor del campo `different_nn_range_per_lod_flag` es verdadero, y el valor del campo `nearest_neighbour_offset_range_in_slice_flag` es verdadero, la cabecera de segmento de atributo incluye además una instrucción de iteración repetida tantas veces como el valor del campo `lifting_num_detail_levels_minus1`. En una realización, el índice `idx` se inicializa a 0 y aumenta en 1 cada vez que se ejecuta la instrucción de iteración. La instrucción de iteración se repite hasta que el índice `idx` es mayor que el valor del campo `lifting_num_detail_levels_minus1`. Esta instrucción de iteración puede incluir un campo `nearest_neighbour_max_range_offset [idx]` y un campo `nearest_neighbour_max_range_offset [idx]`.

En una realización, el campo `lifting_num_detail_levels_minus1` indica el número de LOD para la codificación de atributos y se señala en el APS.

El campo `nearest_neighbour_max_range_offset [idx]` puede especificar un desplazamiento de intervalo máximo de puntos contiguos para el `idx` de LOD cuando se comprime un atributo del segmento. Según una realización, la referencia del desplazamiento de intervalo máximo de puntos contiguos es un intervalo máximo de puntos contiguos de un mosaico al que pertenece el segmento. Según las realizaciones, el campo `nearest_neighbour_max_range_offset [idx]` puede utilizarse para limitar la distancia de un punto registrado como contiguo para el `idx` de LOD en el segmento. Según las realizaciones, el valor del campo `nearest_neighbour_max_range_offset [idx]` puede utilizarse como un desplazamiento del valor de `NN_range` para el `idx` de LOD del segmento.

El campo `nearest_neighbour_min_range_offset [idx]` puede especificar un desplazamiento de intervalo mínimo de puntos contiguos para el `idx` de LOD cuando se comprime un atributo del segmento. En una realización, la referencia del desplazamiento de intervalo mínimo de puntos contiguos es un intervalo mínimo de puntos contiguos de un mosaico al que pertenece el segmento.

La fig. 44 muestra una realización de una estructura sintáctica de los datos de segmento de atributo (`attribute_slice_data()`) según la presente descripción. Los datos de segmento de atributo (`attribute_slice_data()`) según las realizaciones pueden transportar un flujo de bits de atributos que pertenece a un segmento correspondiente.

En los datos de segmento de atributo (`attribute_slice_data()`) de la fig. 44, `dimension=attribute_dimension[ash_attr_sps_attr_idx]` representa la dimensión de atributo (`attribute_dimension`) del conjunto de atributos identificado por el campo `ash_attr_sps_attr_idx` en la cabecera de segmento de atributo correspondiente. El `attribute_dimension` se refiere al número de componentes que constituyen un atributo. Un atributo según las realizaciones representa la reflectancia, el color o similares. Por consiguiente, el número de componentes varía entre atributos. Por ejemplo, un atributo correspondiente al color puede tener tres componentes de color (por ejemplo, RGB). Por consiguiente, un atributo correspondiente a la reflectancia puede ser un atributo mono-dimensional, y el atributo correspondiente al color puede ser un atributo tridimensional.

Los atributos según las realizaciones pueden ser codificados por atributo en una base dimensión a dimensión.

Por ejemplo, el atributo correspondiente a la reflectancia y el atributo correspondiente al color pueden codificarse por atributo, respectivamente. Según las realizaciones, los atributos pueden ser codificados por atributo juntos independientemente de las dimensiones. Por ejemplo, el atributo correspondiente a la reflectancia y el atributo correspondiente al color pueden codificarse conjuntamente por atributo.

En la fig. 44, la ejecución a cero especifica el número de 0 antes del residual).

En la fig. 44, `i` indica un `i`-ésimo valor de punto del atributo. Según una realización, el campo `attr_coding_type` y el campo `lifting_adaptive_prediction_threshold` se señalan en el APS.

`MaxNumPredictors` de la fig. 44 es una variable utilizada en el proceso de decodificación de datos de la nube de puntos, y puede adquirirse basándose en el valor del campo `lifting_adaptive_prediction_threshold` señalado en el APS como sigue.

MaxNumPredictors = campo lifting_max_num_direct_predictors + 1

En este caso, el campo lifting_max_num_direct_predictors indica el número máximo de predictores que se han de utilizar para la predicción directa.

5 Según las realizaciones, predIndex[i] especifica el índice predictor (o modo de predicción) para decodificar el i-ésimo valor de punto del atributo. El valor de predIndex[i] está en el intervalo de 0 al valor del campo lifting_max_num_direct_predictors.

La variable MaxPredDiff[i] según las realizaciones puede calcularse como sigue.

minValue = maxValue = $\tilde{a}_0$

for (j = 0; j < k; j++) {

10 minValue = Min(minValue, $\tilde{a}_j$)

maxValue = Max(maxValue, $\tilde{a}_j$)

}

MaxPredDiff[i] = maxValue - minValue;

15 Aquí, sea k_i el conjunto de los k contiguos más cercanos del punto actual i y sea $(\tilde{a}_j)_{j \in \mathbb{N}_i}$ sus valores de atributos decodificados/reconstruidos. El número de contiguos más cercanos, k_i estará en el intervalo de 1 a lifting_num_pred_nearest_neighbours. Según las realizaciones, los valores de atributos decodificados/reconstruidos de contiguos se obtienen según el proceso de decodificación de elevación predictiva.

20 El campo lifting_num_pred_nearest_neighbours se señala en el APS e indica el número máximo de contiguos más cercanos que se ha de utilizar para la predicción.

La fig. 45 es un diagrama de flujo de un método de transmisión de datos de la nube de puntos según las realizaciones.

25 El método de transmisión de datos de la nube de puntos según las realizaciones puede incluir una etapa 71001 de codificación de geometría contenida en los datos de la nube de puntos, una etapa 71002 de codificación de un atributo contenido en los datos de la nube de puntos basándose en geometría de entrada y/o reconstruida, y una etapa 71003 de transmisión de un flujo de bits que incluye la geometría codificada, el atributo codificado e información de señalización.

30 Las etapas 71001 y 71002 de codificación de la geometría y del atributo contenidos en los datos de la nube de puntos pueden realizar algunas o todas las operaciones del codificador 10002 de vídeo de la nube de puntos de la fig. 1, el proceso 20001 de codificación de la fig. 2, el codificador de vídeo de la nube de puntos de la fig. 4, el codificador de vídeo de la nube de puntos de la fig. 12, el proceso de codificación de la nube de puntos de la fig. 14, el codificador de vídeo de la nube de puntos de la fig. 15 o el codificador de geometría y el codificador de atributo de la fig. 17.

35 En una realización, la etapa 71002 de codificación del atributo puede incluir la generación de un conjunto de LOD_1 aplicando al menos uno de un método de generación de LOD basado en árbol octal, un método de generación de LOD basado en distancia o un método de generación de LOD basado en muestreo, buscando X (>0) puntos NN en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basado en el conjunto de LOD_1 , y registrar los X puntos NN como un conjunto de puntos contiguos en un predictor.

40 Según las realizaciones, la etapa 71002 de codificación del atributo puede configurar el conjunto de puntos contiguos aplicando un intervalo de búsqueda y/o una distancia máxima entre puntos contiguos.

Según las realizaciones, la etapa 71002 de codificación del atributo puede obtener la distancia máxima entre puntos contiguos multiplicando una distancia base entre puntos contiguos por NN_range. El NN_range es un intervalo dentro del cual se puede seleccionar un punto contiguo y se denomina intervalo máximo de puntos contiguos, intervalo de puntos contiguos o intervalo de NN.

45 El intervalo de búsqueda, la distancia base entre puntos contiguos y el NN_range son los mismos que los descritos con referencia a las figs. 15 a 27 y por tanto se omite una descripción detallada de los mismos en la presente memoria.

Según una realización, la etapa 71002 de codificación del atributo incluye la búsqueda de X (por ejemplo, 3) puntos NN entre puntos dentro del intervalo de búsqueda en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto de LOD₁ como se ilustra en la fig. 26. A continuación, solo los puntos NN dentro de la distancia máxima entre puntos contiguos entre los X (por ejemplo, 3) puntos NN pueden registrarse como el conjunto de puntos contiguos. Con referencia a la fig. 26 como ejemplo, la etapa 71002 de codificación del atributo incluye comparar los valores de distancia entre puntos dentro de un intervalo de búsqueda real y el punto Px para buscar X puntos NN y registrar solo puntos dentro de la distancia máxima entre puntos contiguos en un LOD al que pertenece el punto Px entre los X puntos como el conjunto de puntos contiguos del punto Px. Es decir, los puntos contiguos registrados como el conjunto de puntos contiguos del punto Px se limitan a puntos dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto Px entre los X puntos.

Según otra realización, la etapa 71002 de codificación del atributo puede incluir, como se ilustra en la fig. 27, buscar X (por ejemplo, 3) puntos NN dentro de la distancia máxima entre puntos contiguos entre puntos dentro del intervalo de búsqueda en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto de LOD₁ y registrar los X (por ejemplo, 3) puntos NN como el conjunto de puntos contiguos. Con referencia a la fig. 27 como ejemplo, la etapa 71002 de codificación del atributo incluye la comparación de los valores de distancia entre los puntos dentro del intervalo de búsqueda real y el punto Px para buscar X (por ejemplo, 3) puntos NN dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto Px de entre puntos en el intervalo de búsqueda real y registrar los X puntos NN como el conjunto de puntos contiguos del punto Px. Es decir, los puntos contiguos registrados como el conjunto de puntos contiguos del punto Px se limitan a puntos dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto Px.

Según las realizaciones, la etapa 71002 de codificación del atributo incluye la adquisición de un valor de atributo de predicción de cada punto aplicando uno de los modos de predicción 0 a 3 cuando uno o más puntos contiguos están registrados en el predictor de cada punto y la adquisición de valores de atributos residuales de puntos basándose en un valor de atributo original y el valor de atributo de predicción de cada punto.

Según las realizaciones, el modo de predicción 0 es un modo que calcula el valor de atributo de predicción a través de una media ponderada, el modo de predicción 1 es un modo que determina un atributo de un primer punto contiguo como el valor de atributo de predicción, el modo de predicción 2 es un modo que determina un atributo de un segundo punto contiguo como el valor de atributo de predicción, y el modo de predicción 3 es un modo que determina un atributo de un tercer punto contiguo como el valor de atributo de predicción.

Según las realizaciones, en la etapa 71002 de codificación del atributo, si un valor de diferencia máxima entre valores de atributos de puntos contiguos registrados en el predictor de un punto correspondiente es menor que un umbral preestablecido, el modo de predicción 0 se configura como un modo de predicción del punto. Si el valor de diferencia máxima es igual o mayor que el valor umbral preestablecido, se aplica un método RDO a una pluralidad de modos de predicción candidatos y uno de los modos de predicción candidatos se configura como el modo de predicción del punto. En una realización, este proceso se realiza en cada punto.

Según las realizaciones, un modo de predicción aplicado a cada punto puede transmitirse en datos de segmento de atributo.

Según las realizaciones, la etapa 71002 puede aplicar una cuantificación y una codificación de longitud de ejecución cero a los valores de atributos residuales.

En las etapas 71001 y 71002 según las realizaciones, la codificación puede realizarse sobre la base de un segmento o un mosaico que contiene uno o más segmentos.

La etapa 71003 puede ser realizada por el transmisor 10003 de la fig. 1, el proceso 20002 de transmisión de la fig. 2, el procesador 12012 de transmisión de la fig. 12 o el procesador 51008 de transmisión de la fig. 15.

La fig. 46 es un diagrama de flujo de un método de recepción de datos de la nube de puntos según las realizaciones.

Según las realizaciones, un método de recepción de datos de la nube de puntos puede incluir una etapa 81001 de recepción de geometrías codificadas, atributos codificados e información de señalización, una etapa 81002 de decodificación de las geometrías basándose en la información de señalización, una etapa 81003 de decodificación de los atributos basándose en la información de señalización y las geometrías decodificadas/reconstruidas, y una etapa 81004 de renderización de datos de la nube de puntos restaurados basándose en las geometrías decodificadas y los atributos decodificados.

La etapa 81001 según las realizaciones puede ser realizada por el receptor 10005 de la fig. 1, el proceso 20002 de transmisión o el proceso 20003 de decodificación de la fig. 2, el receptor 13000 o el procesador 13001 de recepción de la fig. 13 o el procesador 61001 de recepción de la fig. 20.

En las etapas 81002 y 81003 según las realizaciones, la decodificación se puede realizar sobre la base de un segmento o un mosaico que contiene uno o más segmentos.

Según las realizaciones, la etapa 81002 puede realizar algunas o todas las operaciones del decodificador 10006 de vídeo de la nube de puntos de la fig. 1, el proceso 20003 de decodificación de la fig. 2, el decodificador de vídeo de la nube de puntos de la fig. 11, el decodificador de vídeo de la nube de puntos de la fig. 13, el decodificador de geometría de la fig. 20 o el decodificador de geometría de la fig. 21.

Según las realizaciones, la etapa 81003 puede realizar algunas o todas las operaciones del decodificador 10006 de vídeo de la nube de puntos de la fig. 1, el proceso 20003 de decodificación de la fig. 2, el decodificador de vídeo de la nube de puntos de la fig. 11, el decodificador de vídeo de la nube de puntos de la fig. 13, el decodificador de atributo de la fig. 28 o el decodificador de atributo de la fig. 29.

Según las realizaciones, la información de señalización, por ejemplo, al menos uno de un SPS, un APS, un TPS o una cabecera de segmento de atributo, puede incluir información de opción relacionada con la selección de punto contiguo. Según las realizaciones, la información de opción relacionada con la selección de punto contiguo puede incluir información sobre el NN_range (por ejemplo, campo nearest_neighbor_max_range). Según las realizaciones, la información de opción relacionada con la selección de punto contiguo puede incluir además al menos una de información sobre el número máximo de puntos que pueden establecerse como puntos contiguos (por ejemplo, campo lifting_num_pred_nearest_neighbours), información sobre un intervalo de búsqueda (por ejemplo, campo lifting_search_range) e información sobre un método de configuración de LOD o información sobre una distancia base entre puntos contiguos.

En una realización, la etapa 81003 de decodificación del atributo puede incluir la generación de un conjunto de LOD₁ aplicando al menos uno de un método de generación de LOD basado en árbol octal, un método de generación de LOD basado en distancia o un método de generación de LOD basado en muestreo, buscando X (>0) puntos NN en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basado en el conjunto de LOD₁, y registrar los X puntos NN como un conjunto de puntos contiguos en el predictor. En una realización, el método de generación de LOD se señala en la información de señalización (por ejemplo, APS).

Según las realizaciones, la etapa 81003 de decodificación del atributo puede configurar el conjunto de puntos contiguos aplicando un intervalo de búsqueda y/o una distancia máxima entre puntos contiguos.

Según las realizaciones, la etapa 81003 de codificación del atributo puede obtener la distancia máxima entre puntos contiguos multiplicando una distancia base entre puntos contiguos por NN_range. El NN_range es un intervalo en el que se puede seleccionar un punto contiguo y se denomina intervalo máximo de punto contiguo, intervalo de puntos contiguos o intervalo de NN.

El intervalo de búsqueda, la distancia base entre puntos contiguos y el NN_range son los mismos que los descritos con referencia a las figs. 15 a 29 y por tanto se omite una descripción detallada de los mismos en la presente memoria.

Según una realización, la etapa 81003 de decodificación del atributo incluye la búsqueda de X (por ejemplo, 3) puntos NN entre puntos dentro del intervalo de búsqueda en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto de LOD₁ como se ilustra en la fig. 26. A continuación, solo los puntos NN dentro de la distancia máxima entre puntos contiguos entre los X (por ejemplo, 3) puntos NN pueden registrarse como el conjunto de puntos contiguos. Con referencia a la fig. 26 como ejemplo, la etapa 81003 de decodificación del atributo incluye comparar los valores de distancia entre puntos dentro de un intervalo de búsqueda real y el punto Px para buscar X puntos NN y registrar solo puntos dentro de la distancia máxima entre puntos contiguos en un LOD al que pertenece el punto Px entre los X puntos como el conjunto de puntos contiguos del punto Px. Es decir, los puntos contiguos registrados como el conjunto de puntos contiguos del punto Px se limitan a puntos dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto Px entre los X puntos.

Según otra realización, la etapa 81003 de decodificación del atributo puede incluir, como se ilustra en la fig. 27, buscar X (por ejemplo, 3) puntos NN dentro de la distancia máxima entre puntos contiguos entre puntos dentro del intervalo de búsqueda en un grupo que tiene el mismo o menor LOD (es decir, una gran distancia entre nodos) basándose en el conjunto de LOD₁ y registrar los X (por ejemplo, 3) puntos NN como el conjunto de puntos contiguos. Con referencia a la fig. 27 como ejemplo, la etapa 81003 de decodificación del atributo incluye comparar los valores de distancia entre los puntos dentro del intervalo de búsqueda real y el punto Px para buscar X (por ejemplo, 3) puntos NN dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto Px desde puntos en el intervalo de búsqueda real y registrar los X puntos NN como el conjunto de puntos contiguos del punto Px. Es decir, los puntos contiguos registrados como el conjunto de puntos contiguos del punto Px se limitan a puntos dentro de la distancia máxima entre puntos contiguos en el LOD al que pertenece el punto Px.

Según las realizaciones, la etapa 81003 de decodificación del atributo puede incluir la adquisición de un valor de atributo de predicción de un punto correspondiente aplicando uno de los modos de predicción 0 a 3 cuando uno o más puntos contiguos están registrados en el predictor de un punto específico.

5 Según las realizaciones, el modo de predicción 0 es un modo que calcula un valor de atributo de predicción a través de una media ponderada, el modo de predicción 1 es un modo que determina un atributo de un primer punto contiguo como el valor de atributo de predicción, el modo de predicción 2 es un modo que determina un atributo de un segundo punto contiguo como el valor de atributo de predicción, y el modo de predicción 3 es un modo que determina un atributo de un tercer punto contiguo como el valor de atributo de predicción.

10 Según las realizaciones, un modo de predicción de un punto específico puede configurarse como un defecto e información de índice de predictor (predIndex) que indica el modo de predicción que puede señalizarse en datos de segmento de atributo.

Según una realización, la etapa 81003 puede predecir un valor de atributo de un punto que se ha de decodificar basándose en el modo de predicción 0 cuando la información de índice de predictor del punto no se señala en los datos de segmento de atributo.

15 Según una realización, la etapa 81003 puede predecir un valor de atributo de un punto que se ha de decodificar basándose en un modo de predicción que se señala en los datos de segmento de atributo cuando la información de índice de predictor del punto se señala en los datos de segmento de atributo.

20 Según una realización, la etapa 81003 puede restaurar un valor de atributo del punto añadiendo el valor de atributo predicho del punto y el valor de atributo residual recibido del punto. En una realización, un valor de atributo de cada punto puede restaurarse realizando este proceso para cada punto.

Según una realización, la etapa 81003 puede realizar una decodificación de longitud de ejecución cero en los valores de atributos residuales recibidos, que es un proceso inverso de una codificación de longitud de ejecución cero de un lado de transmisión, antes de restaurar los valores de atributos cuando los valores de atributos residuales recibidos se codifican con longitud de ejecución cero.

25 En la etapa 81004 de renderización de los datos de la nube de puntos según las realizaciones, los datos de la nube de puntos pueden renderizarse según diversos métodos de renderización. Por ejemplo, los puntos del contenido de la nube de puntos pueden renderizarse en un vértice que tiene un cierto espesor, un cubo de un tamaño mínimo específico centrado en la posición del vértice, o un círculo centrado en la posición del vértice. Todo o parte del contenido de la nube de puntos renderizado se proporciona al usuario a través de un dispositivo de visualización (por ejemplo, un dispositivo de visualización de VR/AR, un dispositivo de visualización general, etc.).

30 La etapa 81004 según las realizaciones puede ser realizada por el renderizador 10007 de la fig. 1, el proceso 20004 de renderización de la fig. 2 o el renderizador 13011 de la fig. 13.

35 Como se ha descrito anteriormente, según la presente descripción, se excluye la selección de puntos contiguos sin significado y se pueden seleccionar puntos contiguos significativos cuando se configura un conjunto de puntos contiguos considerando una correlación de atributos entre puntos de contenido durante la codificación de atributos del contenido de la nube de puntos. A continuación, dado que se reduce un valor de atributo residual y se reduce de ese modo un tamaño de flujo de bits, se mejora la eficiencia de compresión de atributos. En otras palabras, en la presente descripción, la eficiencia de compresión de atributos puede mejorarse utilizando un método de restricción de puntos que puede seleccionarse como el conjunto de puntos contiguos en consideración de la correlación de atributos entre puntos de contenido.

40 Cada parte, módulo o unidad descritos anteriormente puede ser una parte de software, procesador o hardware que ejecuta procedimientos sucesivos almacenados en una memoria (o unidad de almacenamiento). Cada una de las etapas descritas en las realizaciones anteriores puede ser realizada por una parte de procesador, software o hardware. Cada módulo/bloque/unidad descrito en las realizaciones anteriores puede operar como un procesador, software o hardware. Además, los métodos presentados por las realizaciones pueden ejecutarse como código. Este código puede escribirse en un medio de almacenamiento legible por procesador y, por lo tanto, leerse por un procesador proporcionado por un aparato.

45 En la especificación, cuando una parte "comprende" o "incluye" un elemento, significa que la parte comprende o incluye además otro elemento a menos que se mencione lo contrario. Además, el término "... módulo(o unidad)" descrito en la especificación significa una unidad para el tratamiento de al menos una función u operación, y puede implementarse mediante hardware, software o combinación de hardware y software.

50 Aunque se han explicado realizaciones con referencia a cada uno de los dibujos adjuntos por simplicidad, es posible diseñar nuevas realizaciones combinando las realizaciones ilustradas en los dibujos adjuntos. Si un medio de grabación legible por un ordenador, en el que se graban programas para ejecutar las realizaciones

mencionadas en la descripción anterior, está diseñado por los expertos en la técnica, puede estar dentro del alcance de las reivindicaciones adjuntas y sus equivalentes.

Los aparatos y métodos pueden no estar limitados por las configuraciones y métodos de las realizaciones descritas anteriormente. Las realizaciones descritas anteriormente pueden configurarse combinándose selectivamente entre sí en su totalidad o en parte para permitir diversas modificaciones.

Aunque se han descrito realizaciones preferidas con referencia a los dibujos, los expertos en la técnica apreciarán que se pueden realizar diversas modificaciones y variaciones en las realizaciones sin apartarse del alcance de la descripción descrita en las reivindicaciones adjuntas. Tales modificaciones no han de comprenderse individualmente a partir de la idea técnica o perspectiva de las realizaciones.

Diversos elementos de los aparatos de las realizaciones pueden implementarse mediante hardware, software, firmware o una combinación de los mismos. Diversos elementos en las realizaciones pueden implementarse mediante un único chip, por ejemplo, un único circuito de hardware. Según realizaciones, los componentes según las realizaciones pueden implementarse como chips separados, respectivamente. Según realizaciones, al menos uno o más de los componentes del aparato según las realizaciones pueden incluir uno o más procesadores capaces de ejecutar uno o más programas. El uno o más programas pueden realizar una cualquiera o más de las operaciones/métodos según las realizaciones o incluir instrucciones para realizar las mismas. Las instrucciones ejecutables para realizar el método/operaciones del aparato según las realizaciones pueden almacenarse en un CRM no transitorio u otros productos de programa informático configurados para ejecutarse por uno o más procesadores, o pueden almacenarse en un CRM transitorio u otros productos de programa informático configurados para ejecutarse por uno o más procesadores. Además, la memoria según las realizaciones puede utilizarse como un concepto que cubre no solo memorias volátiles (por ejemplo, RAM) sino también memorias no volátiles, memorias flash y PROM. Además, también puede implementarse en forma de una onda portadora, tal como transmisión a través de Internet. Además, el medio de grabación legible por procesador puede distribuirse a sistemas informáticos conectados a través de una red de tal manera que el código legible por procesador puede almacenarse y ejecutarse de manera distribuida.

En este documento, el término "/" y "," deberían interpretarse como indicativo de "y/o". Por ejemplo, la expresión "A/B" puede significar "A y/o B". Además, "A, B" puede significar "A y/o B". Además, "A/B/C" puede significar "al menos uno de A, B y/o C". "A, B, C" también puede significar "al menos uno de A, B y/o C".

Además, en el documento, el término "o" debería interpretarse como "y/o". Por ejemplo, la expresión "A o B" puede significar 1) solo A, 2) solo B y/o 3) tanto A como B. En otras palabras, el término "o" en este documento debería interpretarse como "adicional o alternativamente".

Diversos elementos de las realizaciones pueden implementarse mediante hardware, software, firmware o una combinación de los mismos. Diversos elementos en las realizaciones pueden ejecutarse mediante un único chip tal como un único circuito de hardware. Según realizaciones, el elemento puede ejecutarse selectivamente mediante chips separados, respectivamente. Según las realizaciones, al menos uno de los elementos de las realizaciones puede ejecutarse en uno o más procesadores que incluyen instrucciones para realizar operaciones según las realizaciones.

Términos tales como primero y segundo pueden utilizarse para describir diversos elementos de las realizaciones. Sin embargo, varios componentes según las realizaciones no deberían estar limitados por los términos anteriores. Estos términos se utilizan únicamente para distinguir un elemento de otro. Por ejemplo, una primera señal de entrada de usuario puede denominarse segunda señal de entrada de usuario. De manera similar, la segunda señal de entrada de usuario puede denominarse primera señal de entrada de usuario. El uso de estos términos debería interpretarse como que no se aparta del alcance de las diversas realizaciones. La primera señal de entrada de usuario y la segunda señal de entrada de usuario son ambas señales de entrada de usuario, pero no significan la misma señal de entrada de usuario a menos que el contexto indique claramente lo contrario.

La terminología utilizada para describir las realizaciones se utiliza con el propósito de describir realizaciones particulares solamente y no pretende ser limitante de las realizaciones. Como se utiliza en la descripción de las realizaciones y en las reivindicaciones, las formas singulares "un/una", "uno" y "el/la" incluyen referentes plurales a menos que el contexto indique claramente lo contrario. La expresión "y/o" se utiliza para incluir todas las combinaciones posibles de términos. Los términos tales como "incluye" o "tiene" pretenden indicar la existencia de figuras, números, etapas, elementos y/o componentes y debería entenderse que no excluye la posibilidad de existencia de existencia adicional de figuras, números, etapas, elementos y/o componentes. Como se utiliza en la presente memoria, las expresiones condicionales tales como "si" y "cuando" no se limitan a un caso opcional y se pretende que se interpreten, cuando se cumple una condición específica, para realizar la operación relacionada o interpretar la definición relacionada según la condición específica.

Además, las operaciones según las realizaciones que se describen en el presente documento pueden realizarse mediante dispositivos de transmisión/recepción que incluyen una memoria y/o un procesador. La memoria puede almacenar programas para procesar/controlar operaciones según las realizaciones y el procesador puede

controlar operaciones variables que se describen en el presente documento. El procesador puede denominarse controlador, etc. Las operaciones según las realizaciones pueden implementarse mediante firmware, software y/o una combinación de los mismos y el firmware, software y/o una combinación de los mismos pueden almacenarse en un procesador o una memoria.

5 Modo para la invención

Como se ha descrito anteriormente, los contenidos relacionados se han descrito en el mejor modo para llevar a cabo las realizaciones.

Aplicabilidad industrial

- 10 Como se ha descrito anteriormente, las realizaciones pueden aplicarse completa o parcialmente al dispositivo y sistema de transmisión/recepción de datos de la nube de puntos. Será evidente para los expertos en la técnica que se pueden hacer cambios o modificaciones de diversas maneras a las realizaciones dentro del alcance de las realizaciones. Por lo tanto, se pretende que las realizaciones cubran las modificaciones y variaciones de esta descripción siempre que estén dentro del alcance de las reivindicaciones adjuntas y sus equivalentes.

REIVINDICACIONES

1.- Un método de transmisión de datos de la nube de puntos, comprendiendo el método:

la adquisición de datos de la nube de puntos;

la codificación (71001) de información de geometría que incluye posiciones de puntos de los datos de la nube de puntos;

la generación de uno o más niveles de detalle, LOD, basándose en un árbol octal y la selección de uno o más puntos contiguos de cada punto que se ha de codificar por atributo basándose en el uno o más LOD, en donde cada nivel de profundidad del árbol octal coincide con cada LOD, en donde un LOD superior incluye puntos pertenecientes a un LOD inferior, en donde el uno o más puntos contiguos seleccionados de cada punto están ubicados dentro de una distancia máxima entre puntos contiguos, en donde la distancia máxima entre puntos contiguos se obtiene multiplicando una distancia base por un intervalo máximo de puntos contiguos, en donde la distancia base es una distancia diagonal de un nodo en cada LOD, en donde el intervalo máximo de puntos contiguos es un número de nodos alrededor de cada punto, y en donde el uno o más puntos contiguos seleccionados de cada punto se utiliza para generar información de atributo predicha de cada punto;

la codificación (71002) de información de atributo de cada punto basándose en la información de atributo predicha de cada punto; y

la transmisión (71003) de la información de geometría codificada, la información de atributo codificada y la información de señalización.

2.- El método de la reivindicación 1,

en donde el intervalo máximo de puntos contiguos se señala en la información de señalización.

3.- Un aparato de transmisión de datos de la nube de puntos, comprendiendo el aparato:

una unidad (10001) de adquisición configurada para adquirir datos de la nube de puntos;

un codificador (51006) de geometría configurado para codificar información de geometría que incluye posiciones de puntos de los datos de la nube de puntos;

un codificador (51007) de atributos configurado para:

generar uno o más niveles de detalle, LOD, basándose en un árbol octal,

seleccionar uno o más puntos contiguos de cada punto que se ha de codificar por atributo basándose en el uno o más LOD, en donde cada nivel de profundidad del árbol octal coincide con cada LOD, en donde un LOD superior incluye puntos que pertenecen a un LOD inferior, en donde el uno o más puntos contiguos seleccionados de cada punto se ubican dentro de una distancia máxima entre puntos contiguos, en donde la distancia máxima entre puntos contiguos se obtiene multiplicando una distancia base por un intervalo máximo de puntos contiguos, en donde la distancia base es una distancia diagonal de un nodo en cada LOD, en donde el intervalo máximo de puntos contiguos es un número de nodos alrededor de cada punto, y en donde el uno o más puntos contiguos seleccionados de cada punto se utiliza para generar información de atributo predicha de cada punto, y

codificar la información de atributo de cada punto basándose en la información de atributo predicha de cada punto; y

un transmisor (10003) configurado para transmitir la información de geometría codificada, la información de atributo codificada e información de señalización.

4.- El aparato de la reivindicación 3,

en donde el intervalo máximo de puntos contiguos se señala en la información de señalización.

5.- Un aparato de recepción de datos de la nube de puntos, comprendiendo el aparato:

un receptor (13000) configurado para recibir información de geometría, información de atributo e información de señalización;

un decodificador (61003) de geometría configurado para decodificar la información de geometría basándose en la información de señalización;

un decodificador (61004) de atributo configurado para:

generar uno o más niveles de detalle, LOD, basándose en un árbol octal,

seleccionar uno o más puntos contiguos de cada punto que se ha de decodificar por atributo basándose en el uno o más LOD, en donde cada nivel de profundidad del árbol octal coincide con cada LOD, en donde un LOD superior incluye puntos que pertenecen a un LOD inferior, en donde el uno o más puntos contiguos seleccionados de cada punto se ubican dentro de una distancia máxima entre puntos contiguos, en donde la distancia máxima entre puntos contiguos se obtiene multiplicando una distancia base por un intervalo máximo de puntos contiguos, en donde la distancia base es una distancia diagonal de un nodo en cada LOD, en donde el intervalo máximo de puntos contiguos es un número de nodos alrededor de cada punto, y en donde el uno o más puntos contiguos seleccionados de cada punto se utiliza para generar información de atributo predicha de cada punto, y

decodificar la información de atributo de cada punto basándose en la información de atributo predicha de cada punto y la información de señalización; y

un renderizador (10007) configurado para renderizar datos de la nube de puntos restaurados basándose en la información de geometría decodificada y la información de atributo decodificada.

6.- El aparato de la reivindicación 5,

en donde el intervalo máximo de puntos contiguos se señala en la información de señalización.

7.- Un método de recepción de datos de la nube de puntos, comprendiendo el método:

la recepción (81001) de información de geometría, información de atributo e información de señalización;

la decodificación (81002) de la información de geometría a partir de la información de señalización;

la generación de uno o más niveles de detalle, LOD, basándose en un árbol octal;

la selección de uno o más puntos contiguos de cada punto que se ha de decodificar por atributo basándose en el uno o más LOD, en donde cada nivel de profundidad del árbol octal coincide con cada LOD, en donde un LOD superior incluye puntos pertenecientes a un LOD inferior, en donde el uno o más puntos contiguos seleccionados de cada punto están ubicados dentro de una distancia máxima entre puntos contiguos, en donde la distancia máxima entre puntos contiguos se obtiene multiplicando una distancia base por un intervalo máximo de puntos contiguos, en donde la distancia base es una distancia diagonal de un nodo en cada LOD, en donde el intervalo máximo de puntos contiguos es un número de nodos alrededor de cada punto, y en donde el uno o más puntos contiguos seleccionados de cada punto se utiliza para generar información de atributo predicho de cada punto;

la decodificación (81003) de la información de atributo de cada punto basándose en la información de atributo predicha de cada punto y la información de señalización; y

la renderización (81004) de datos de la nube de puntos restaurados basándose en la información de geometría decodificada y la información de atributo decodificado.

8.- El método de la reivindicación 7,

en donde el intervalo máximo de puntos contiguos se señala en la información de señalización.

FIG. 1

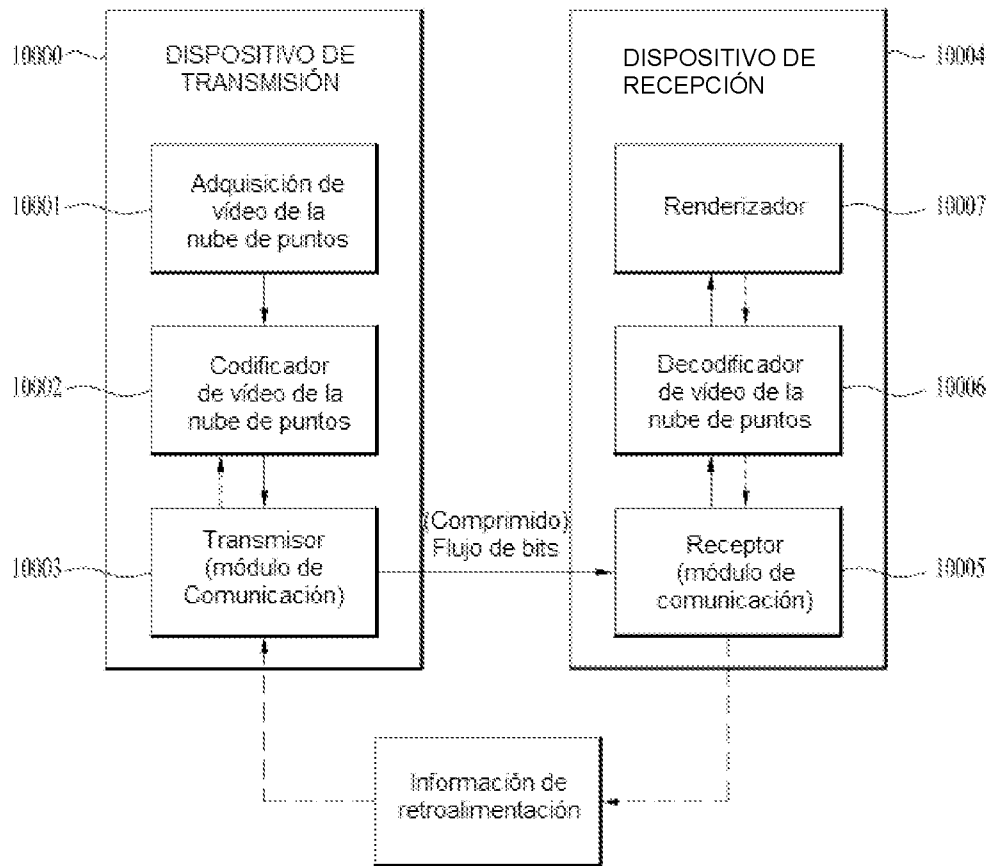


FIG. 2

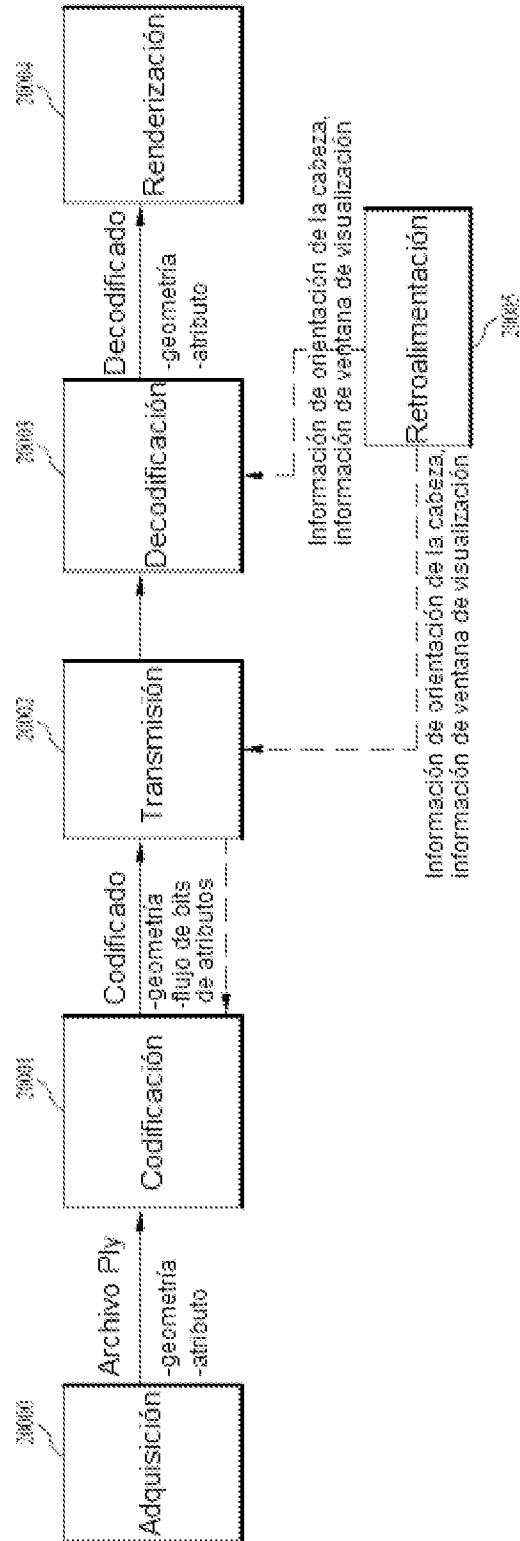


FIG. 3

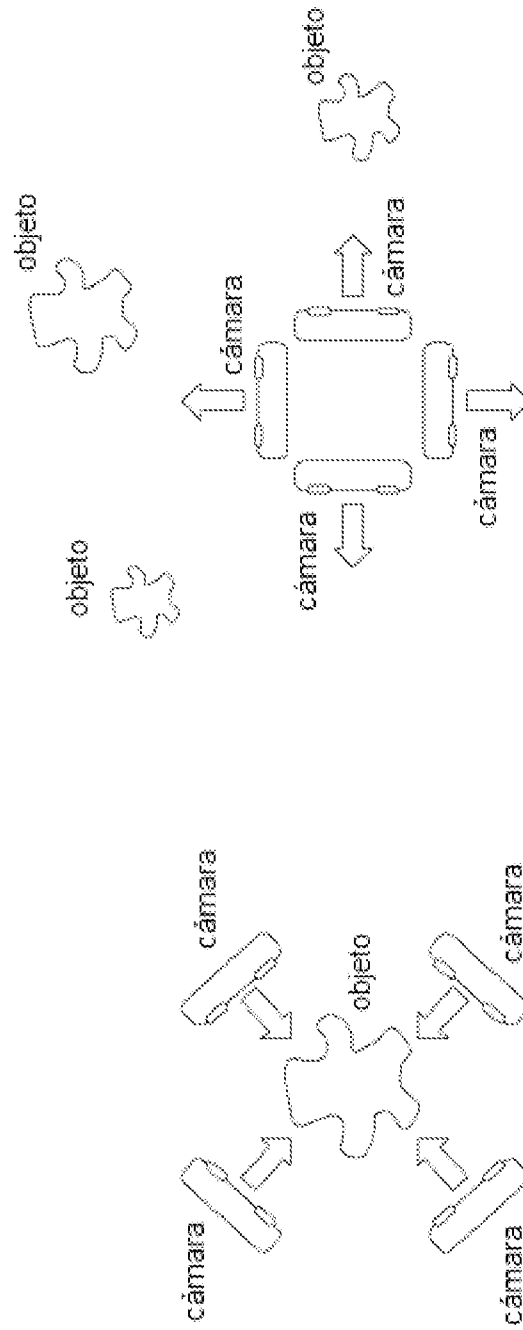


FIG. 4

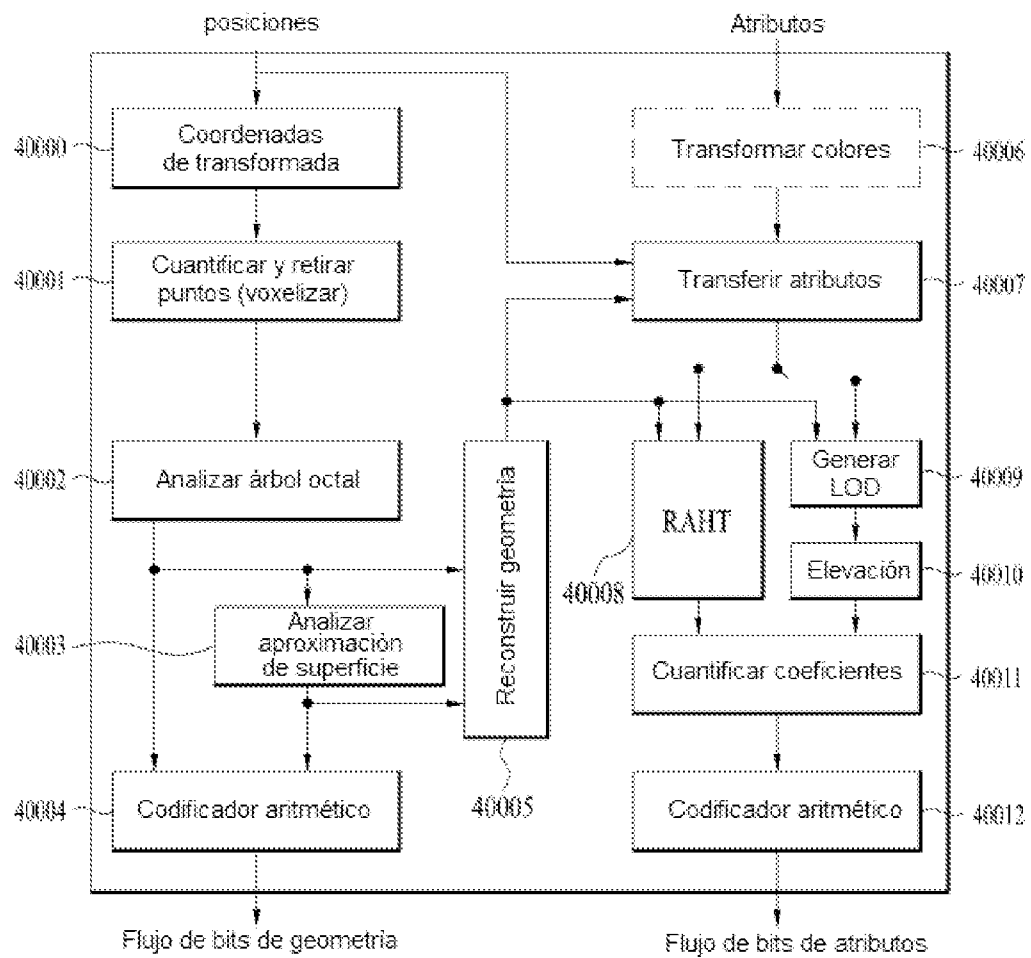


FIG. 5

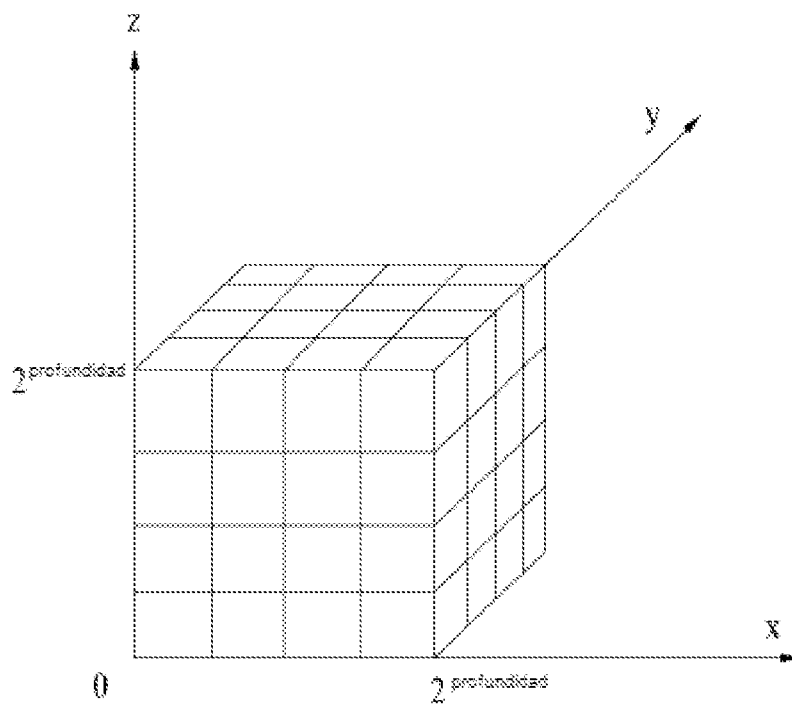


FIG. 6

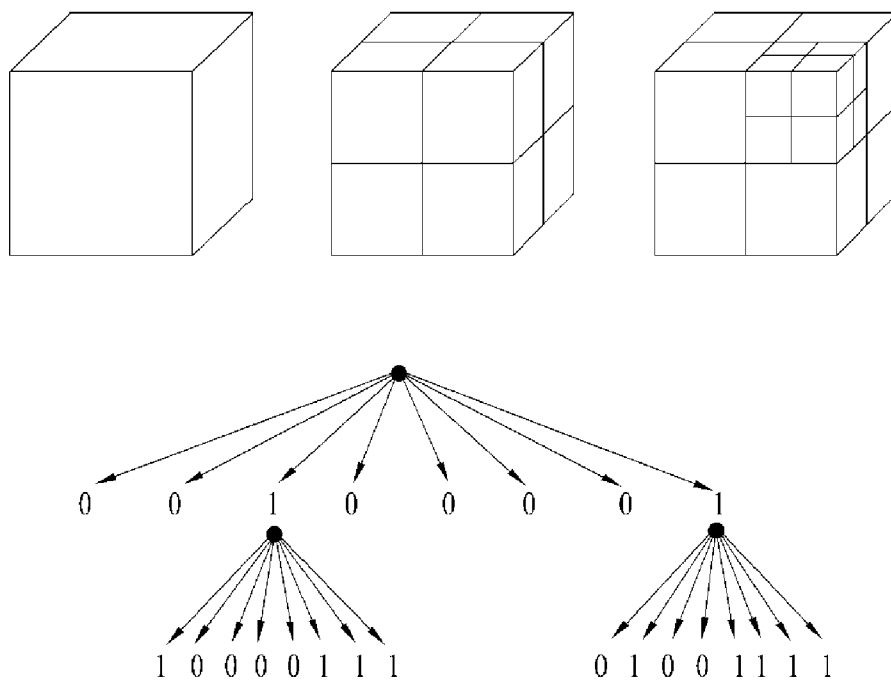


FIG. 7

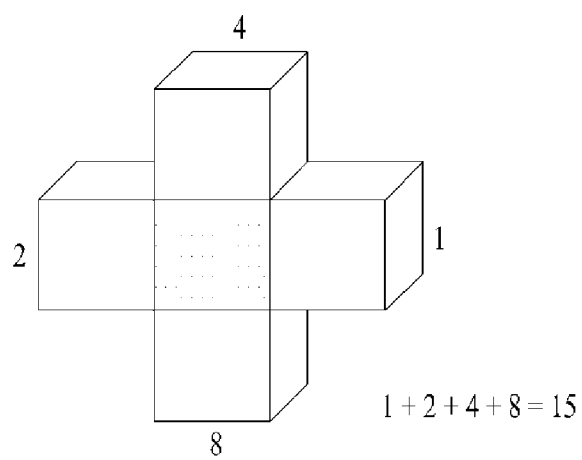
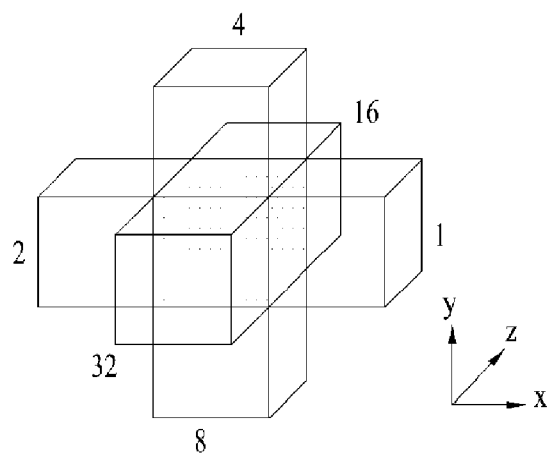


FIG. 8

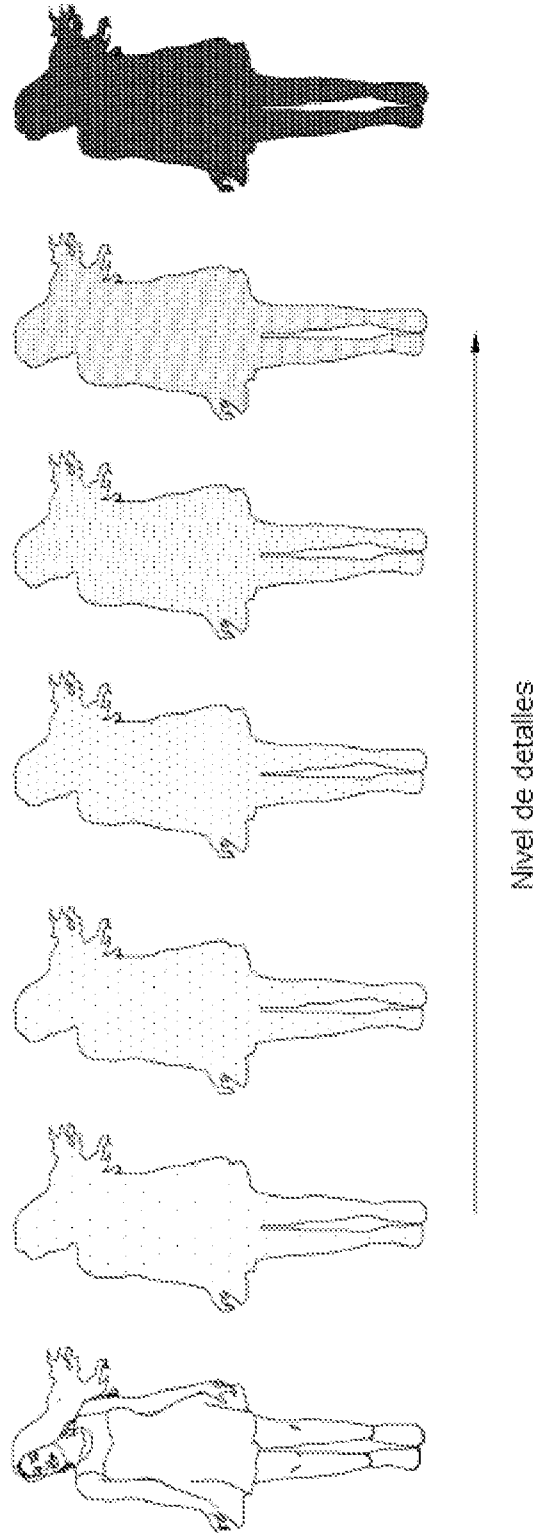


FIG. 9

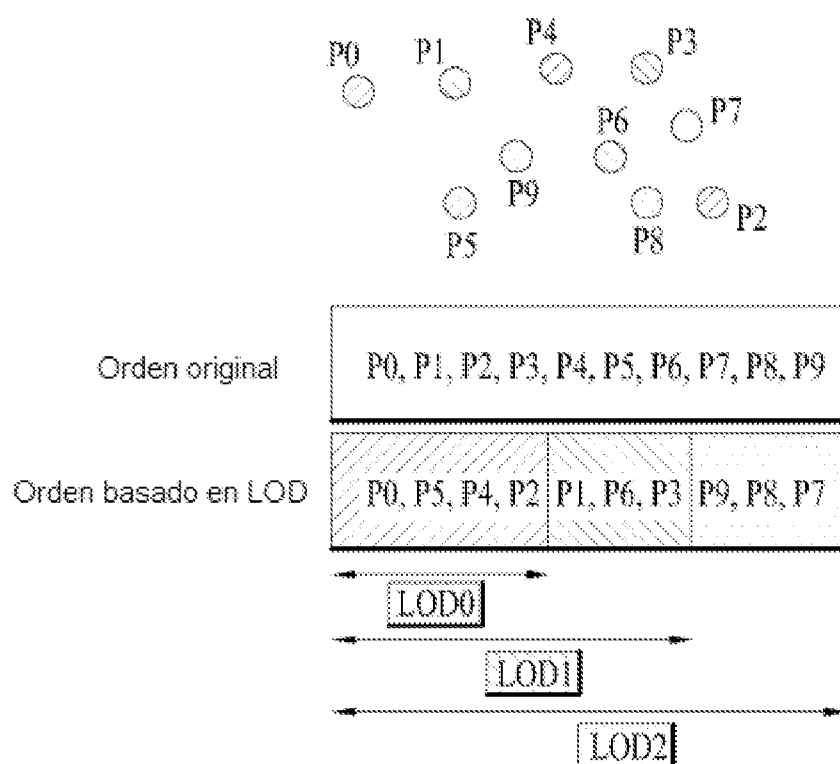


FIG. 10

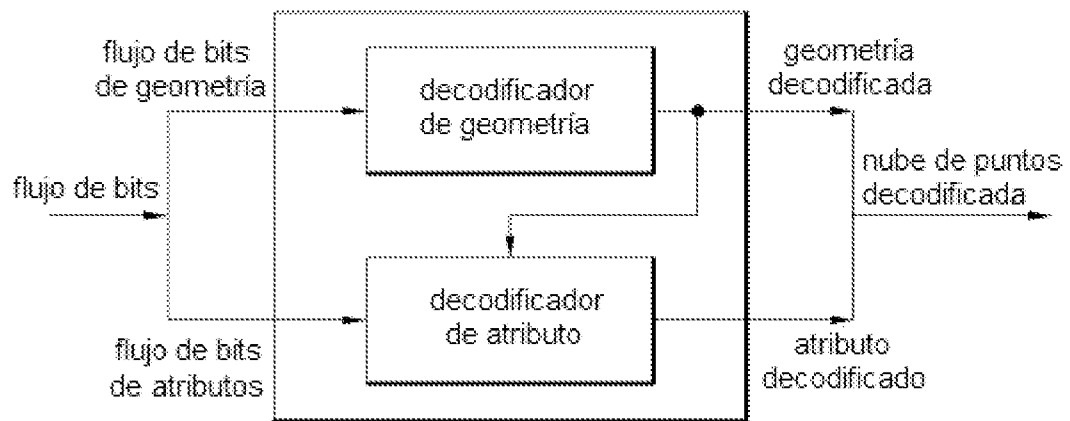


FIG. 11

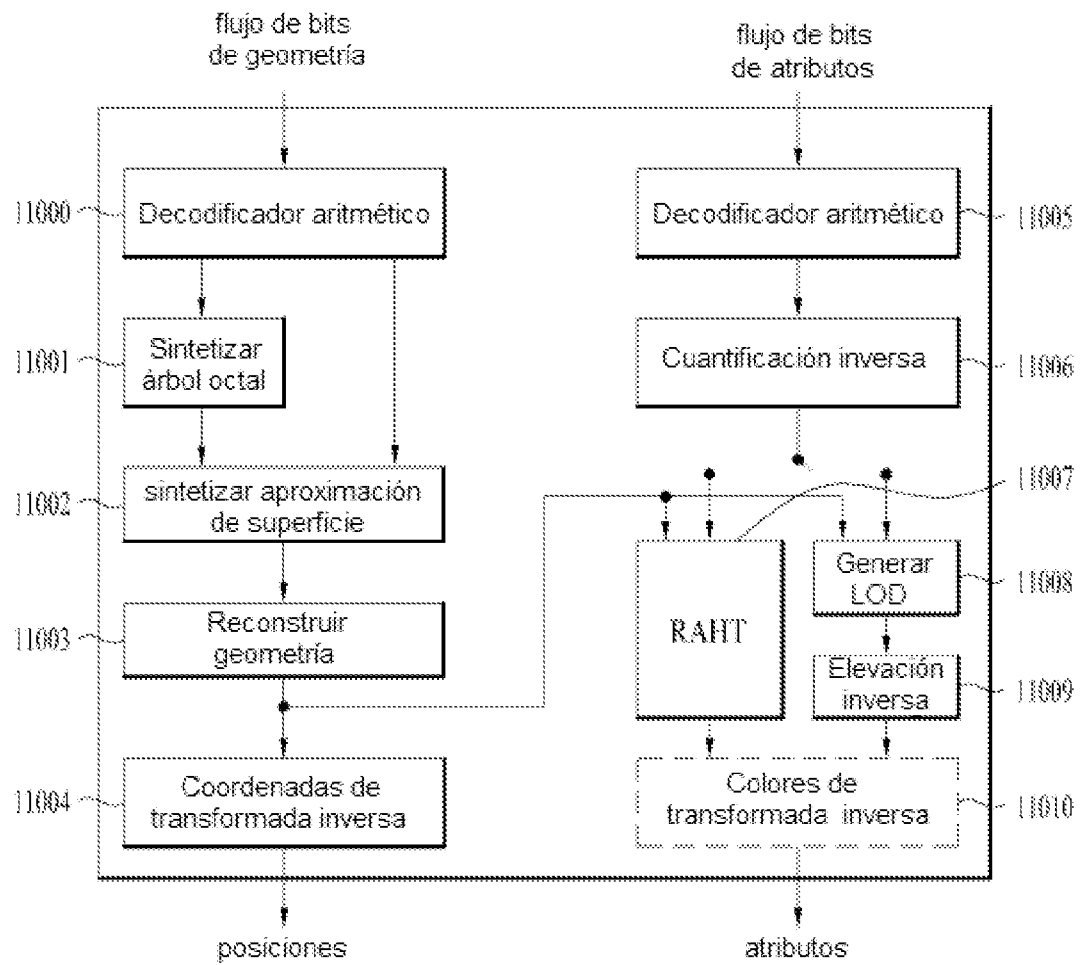


FIG. 12

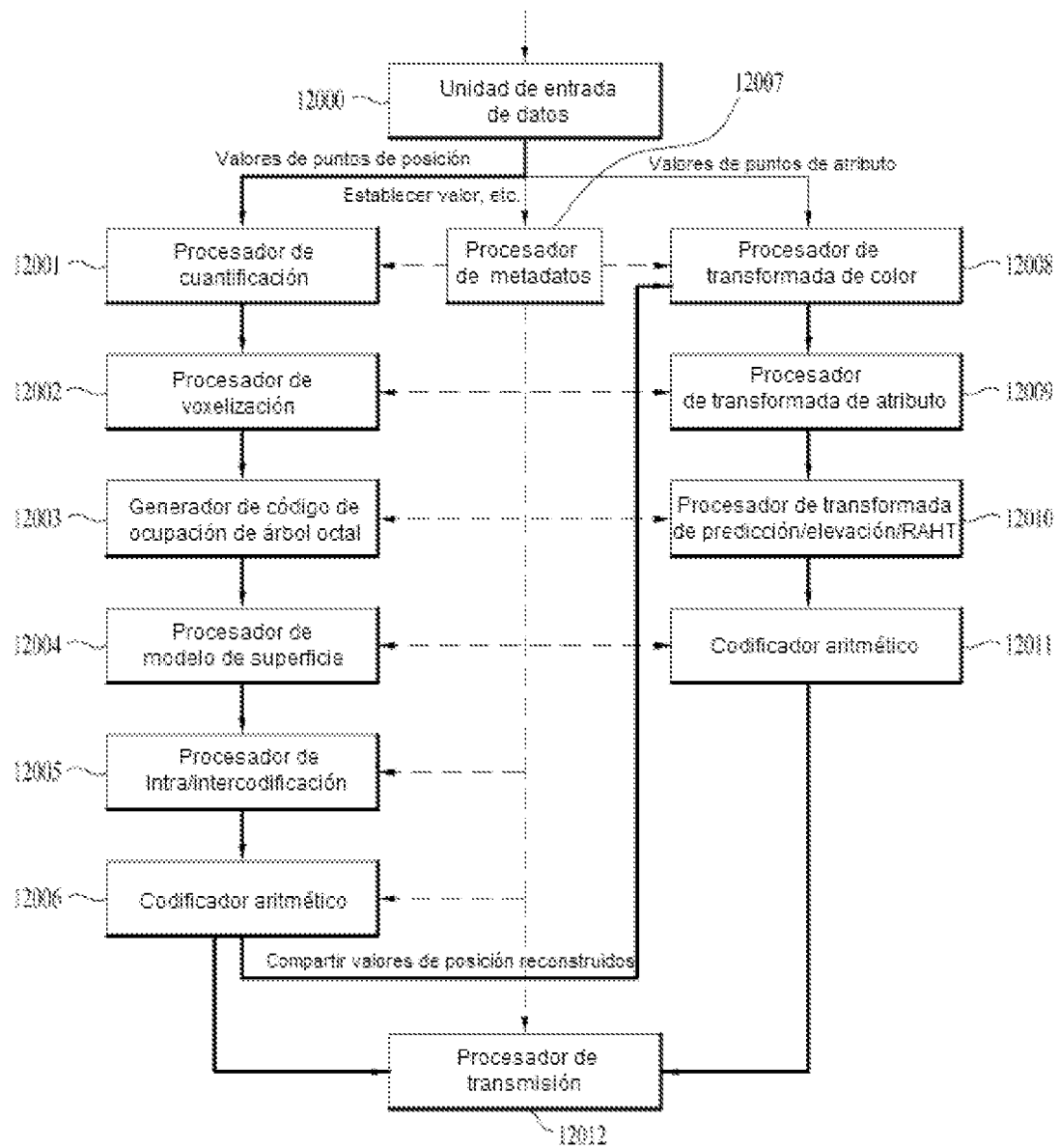


FIG. 13

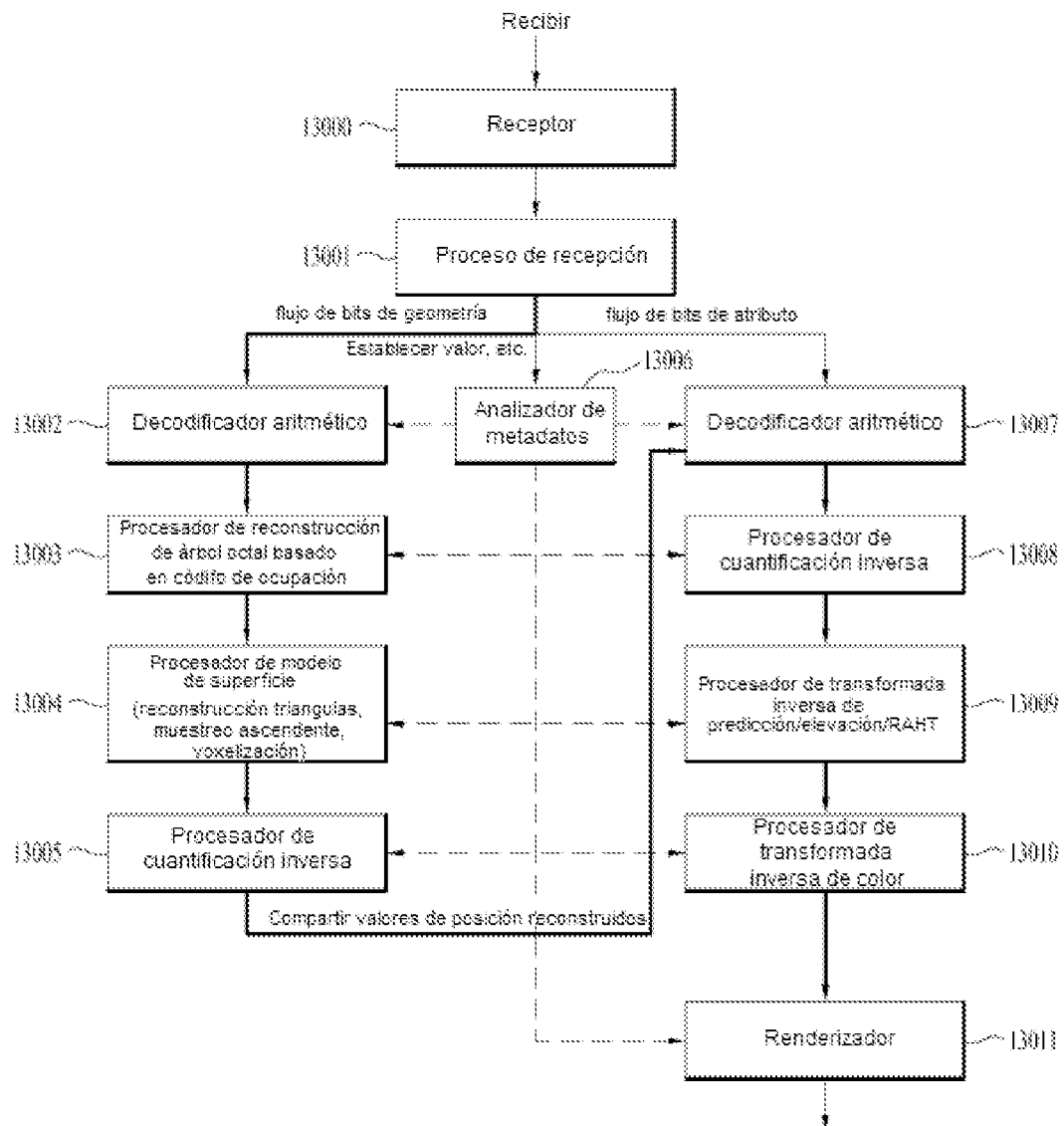


FIG. 14

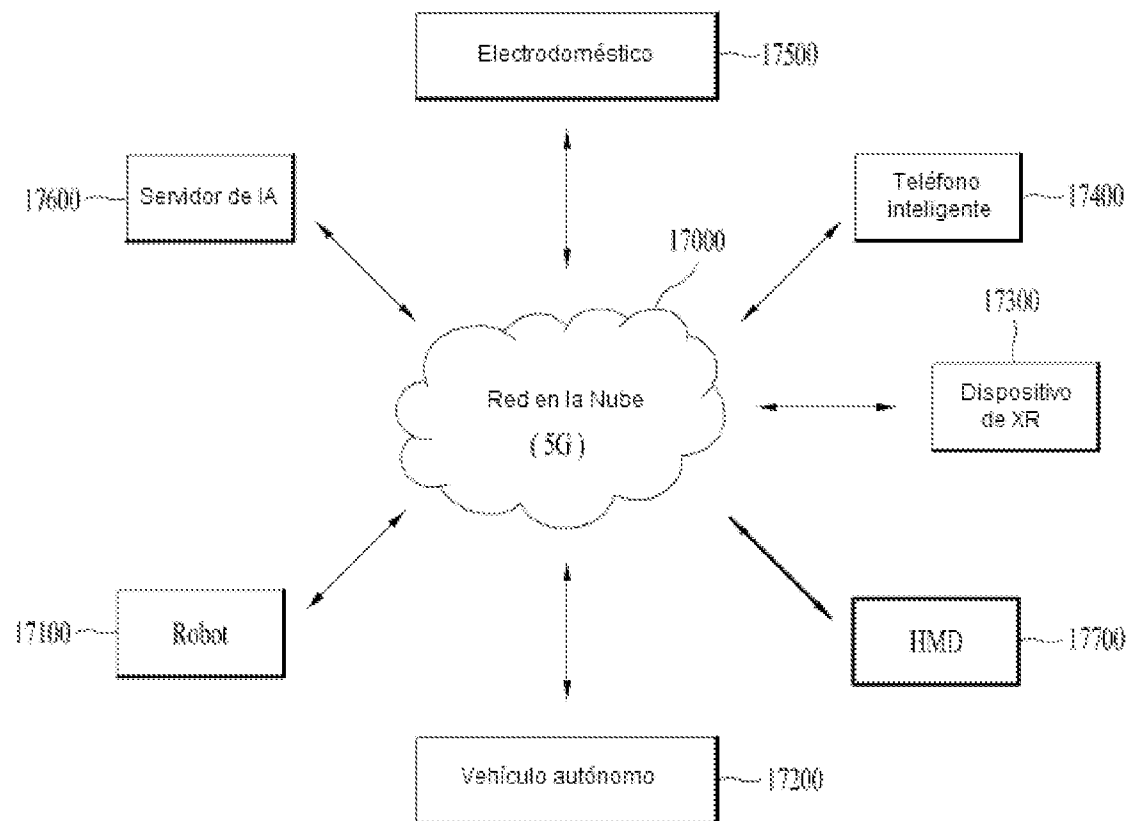


FIG. 15

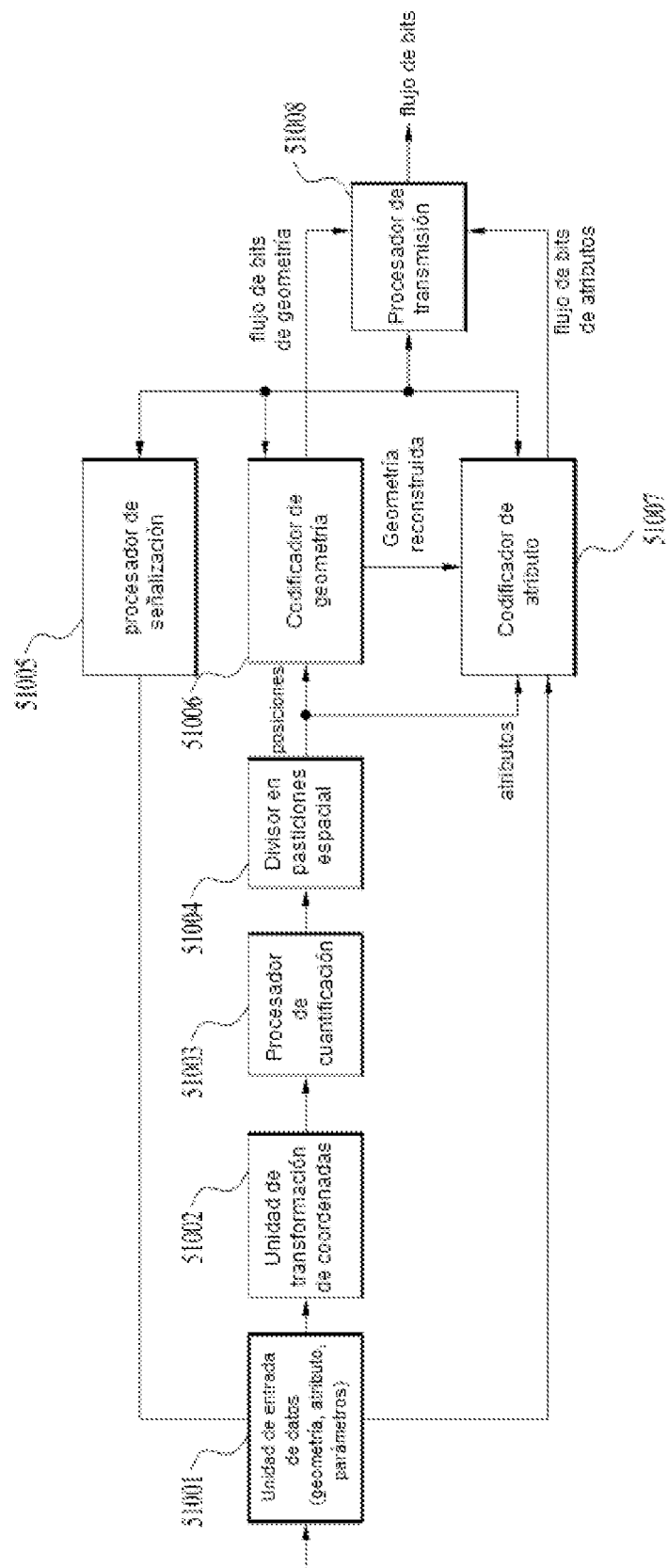


FIG. 16

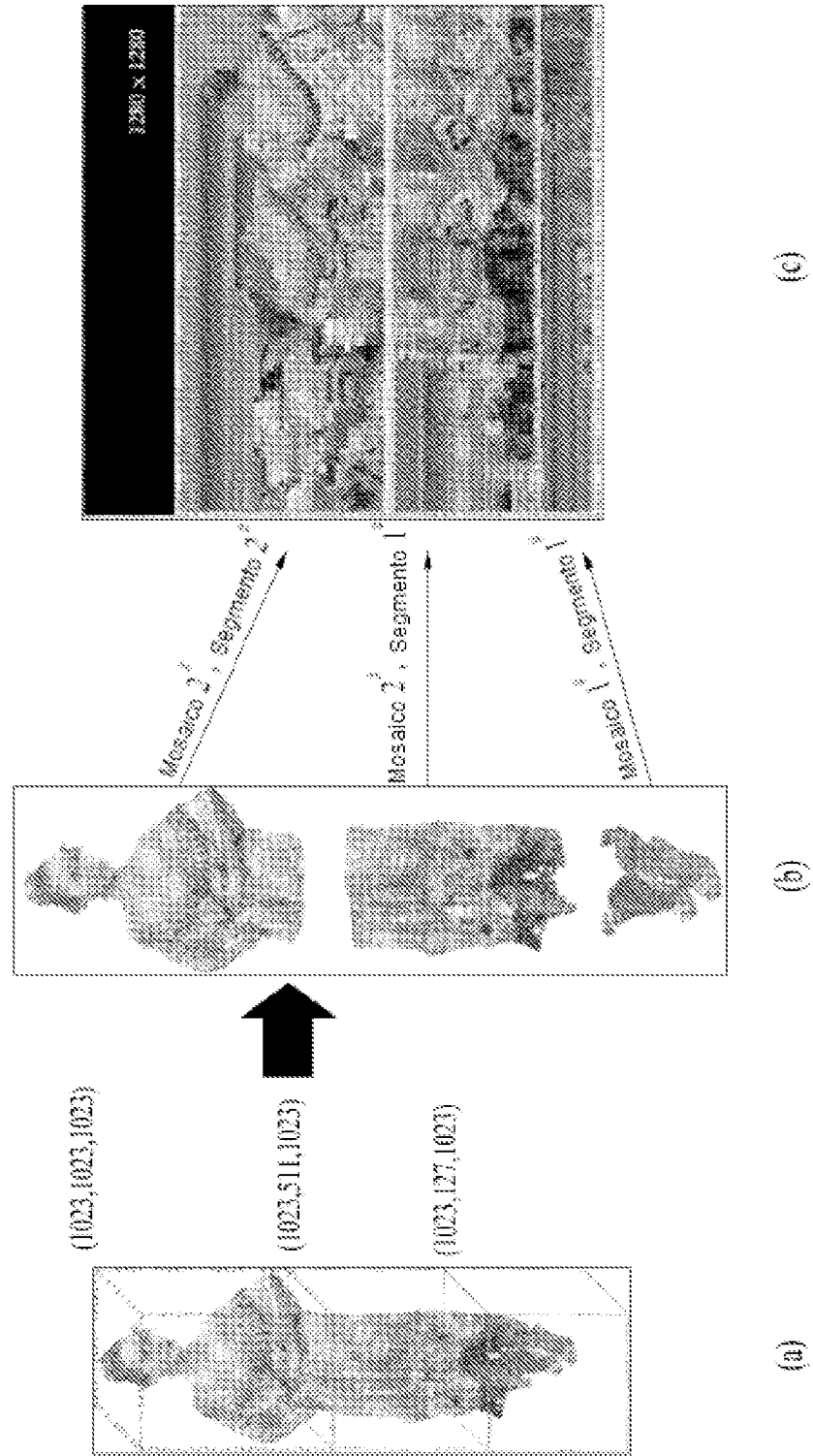


FIG. 17

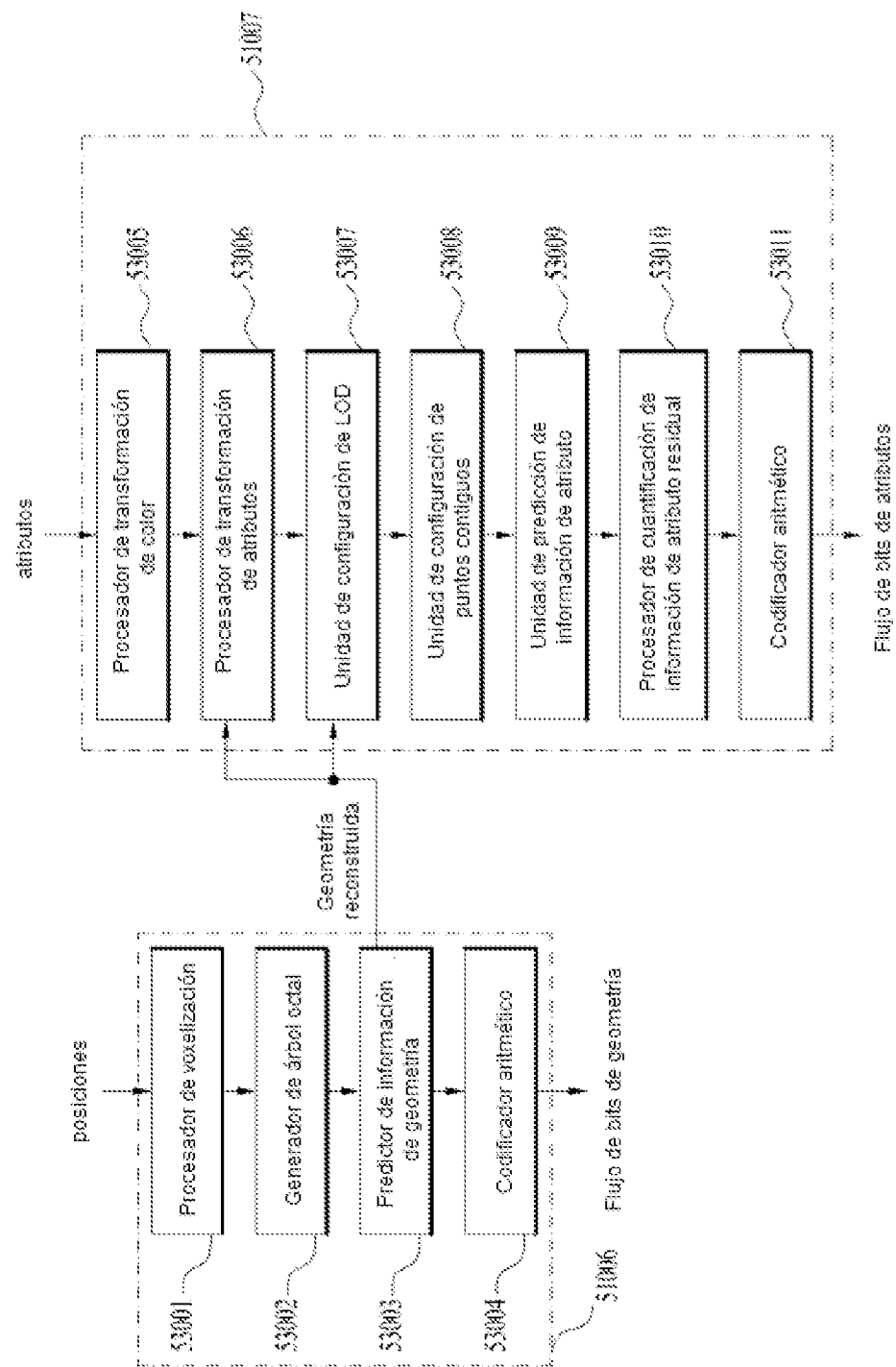


FIG. 18

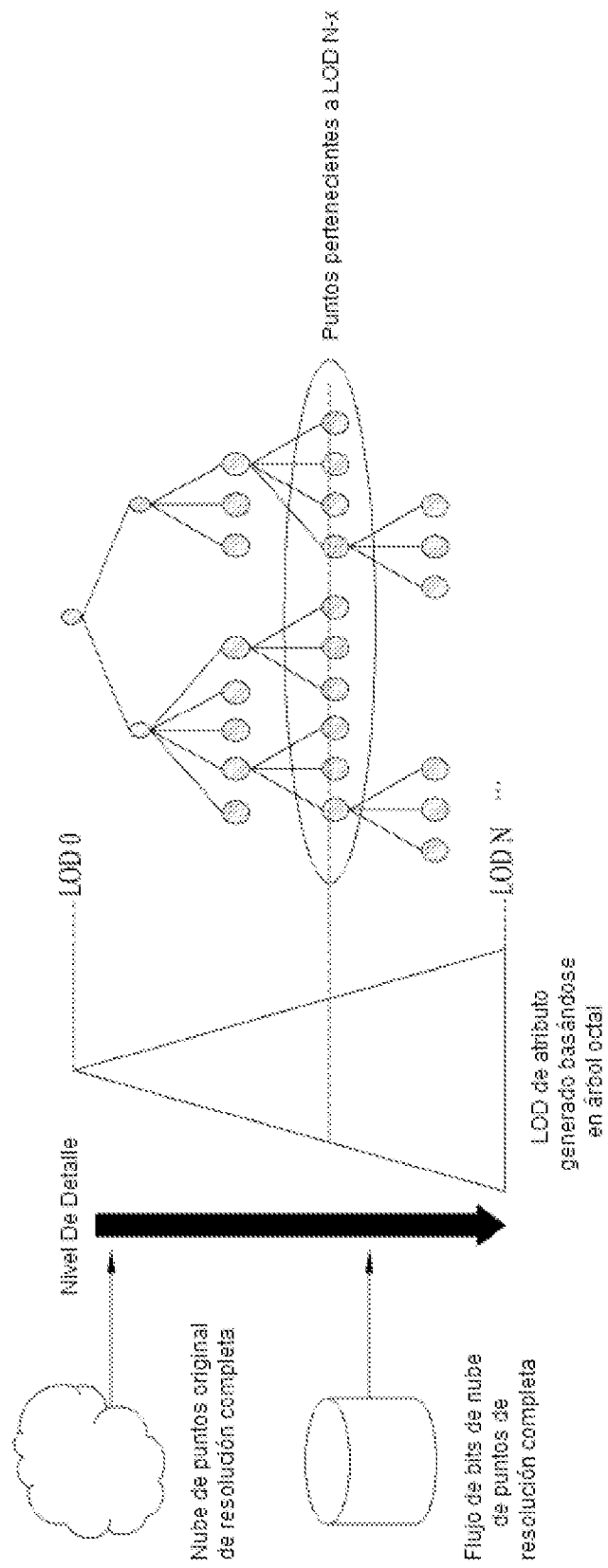


FIG. 19

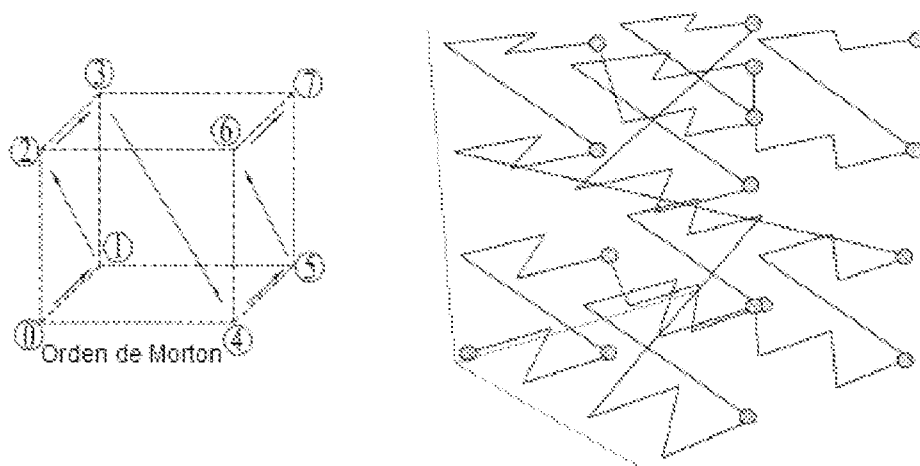


FIG. 20

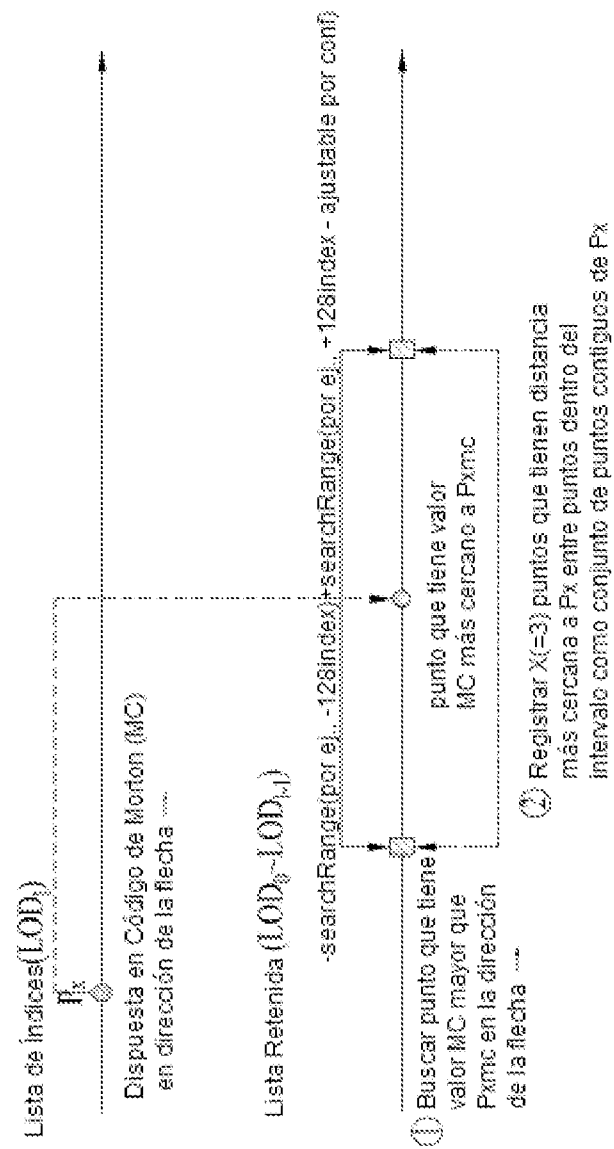
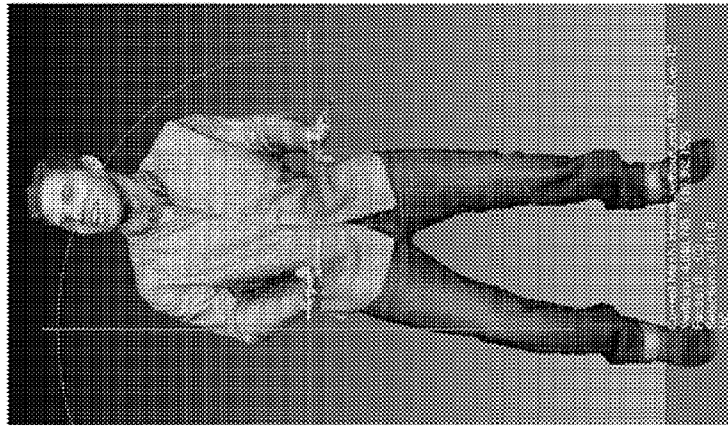


FIG. 21



(a)



(b)

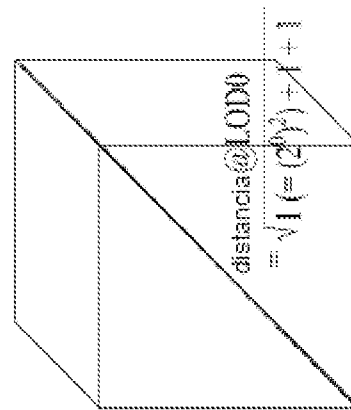
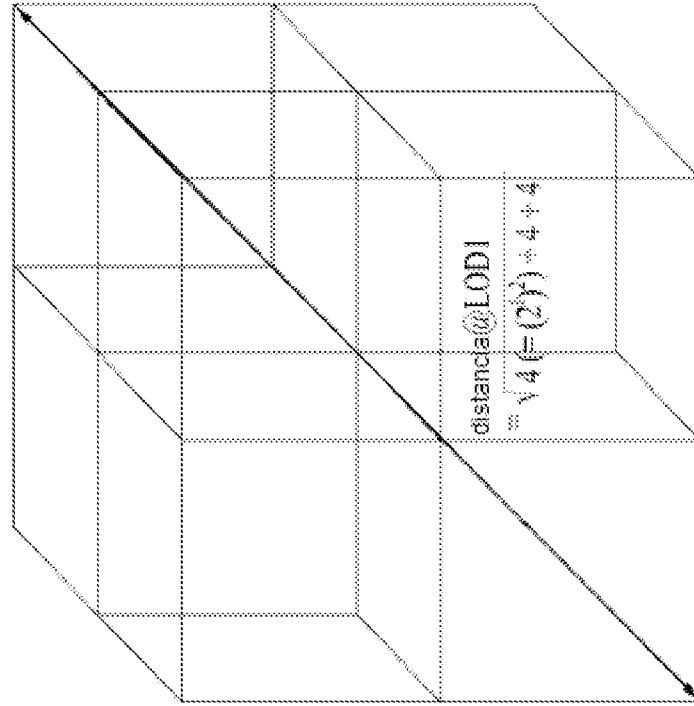
FIG. 22

FORD(#0100)				QNX(#0001)			
Número de puntos para NN=1	4			Número de puntos para NN=1	3496		
Número de puntos para NN=2	60			Número de puntos para NN=2	219		
Número de puntos para NN=3	80201			Número de puntos para NN=3	27564		
Número de puntos total	80265			Número de puntos total	31279		
Distancia Norm2				Distancia Norm2			
	NN1	NN2	NN3		NN1	NN2	NN3
promedio	128	748	1765	promedio	129	25131	42839
min	1	1	1	min	1	2	8
max	256	1407616	1892992	max	256	116007040	310887696

(a)

(b)

FIG. 23



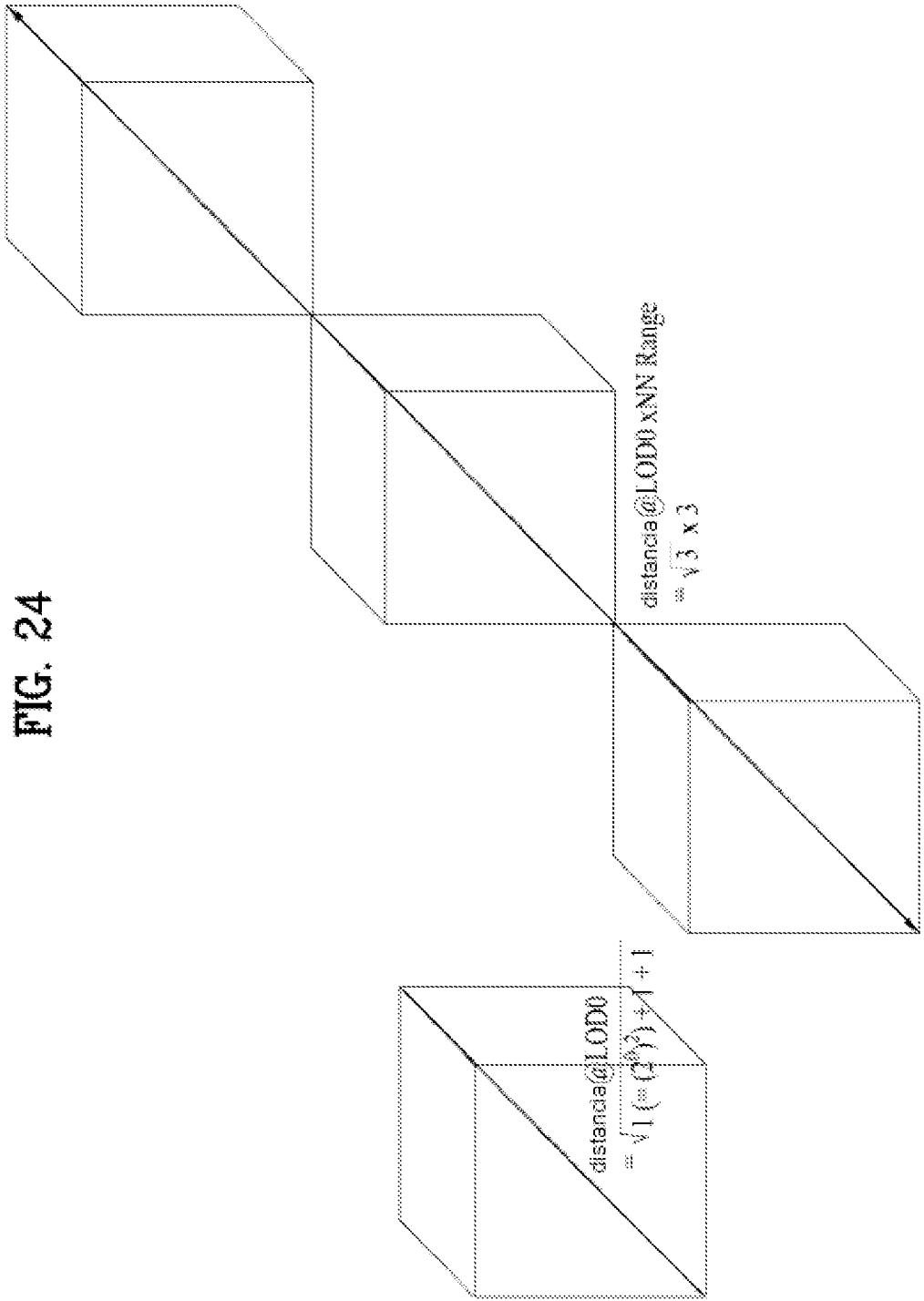


FIG. 25

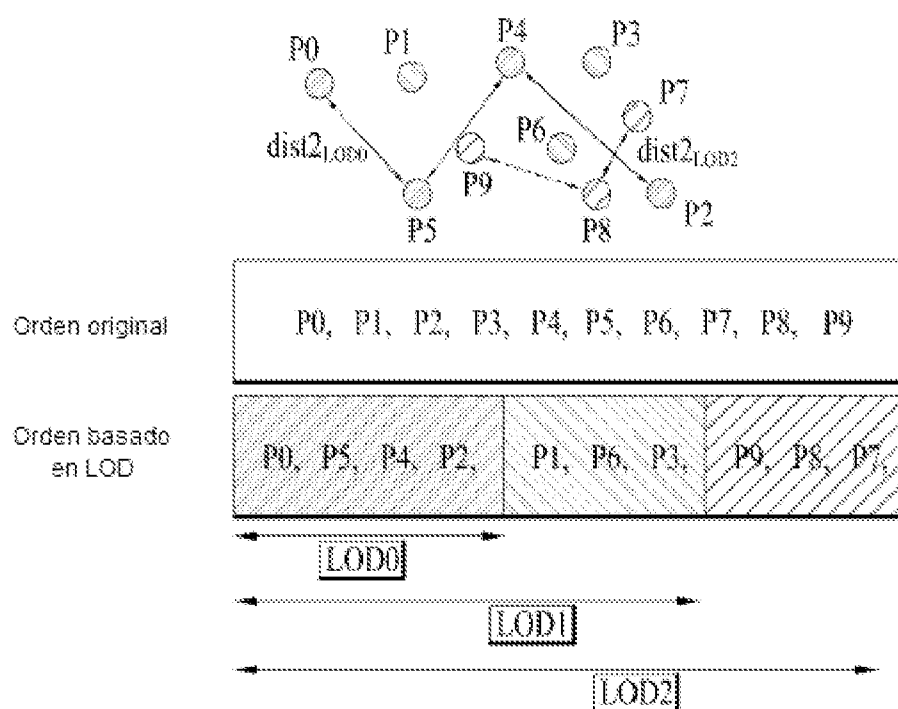


FIG. 26

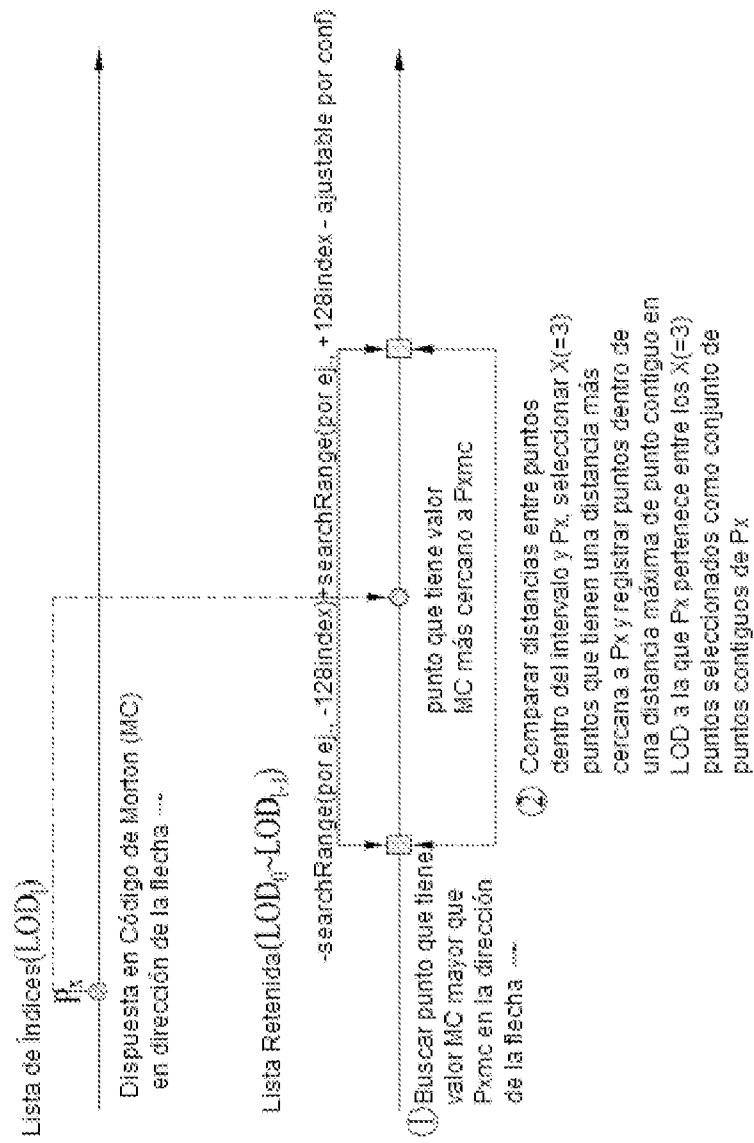


FIG. 27

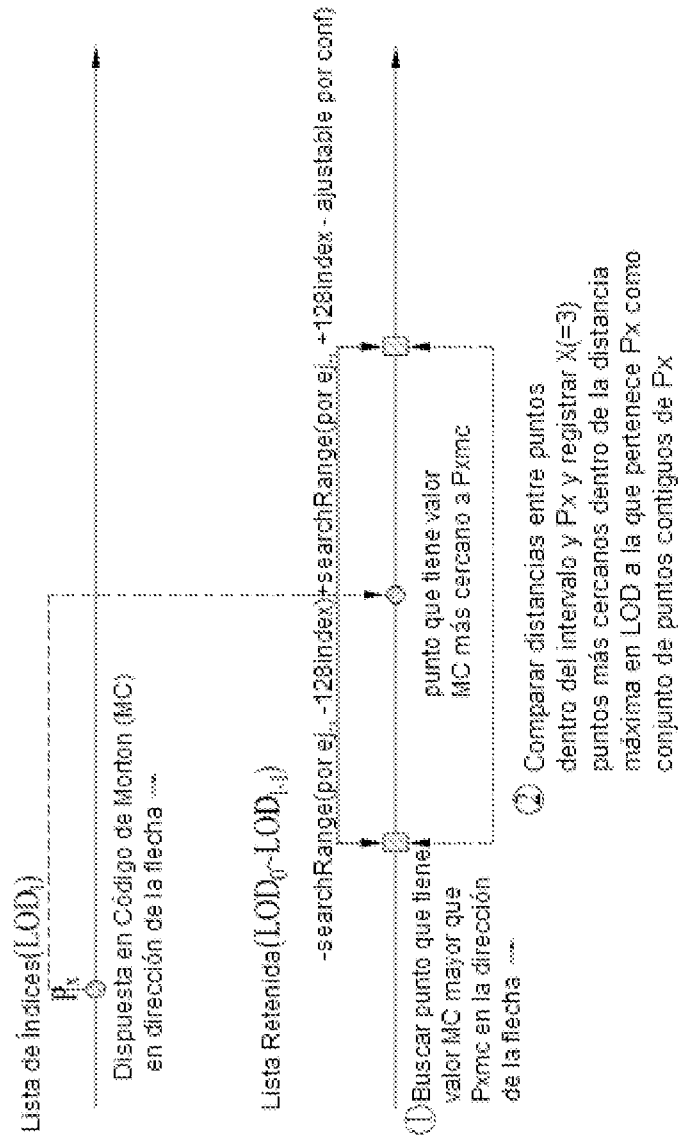


FIG. 28

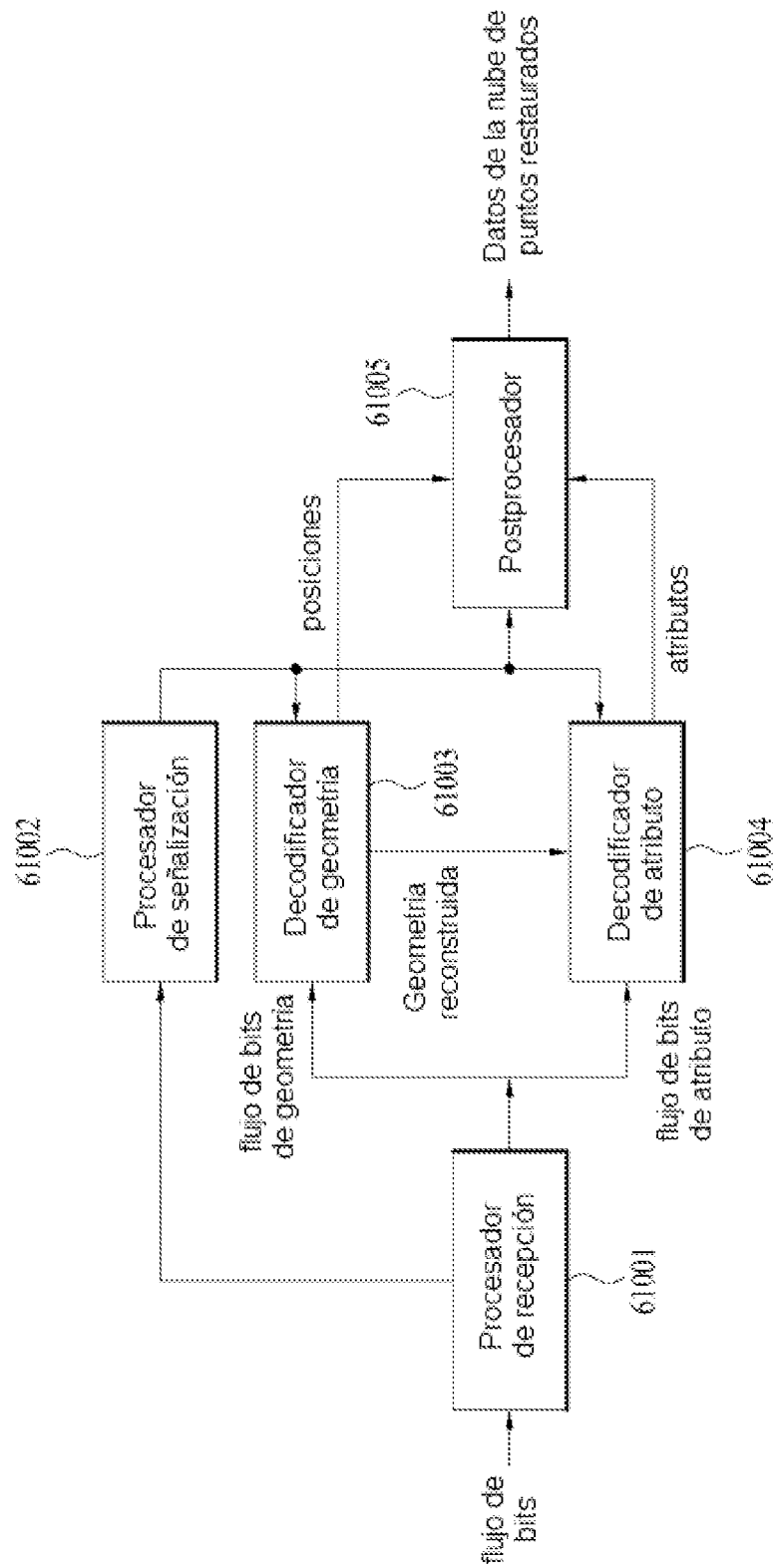


FIG. 29

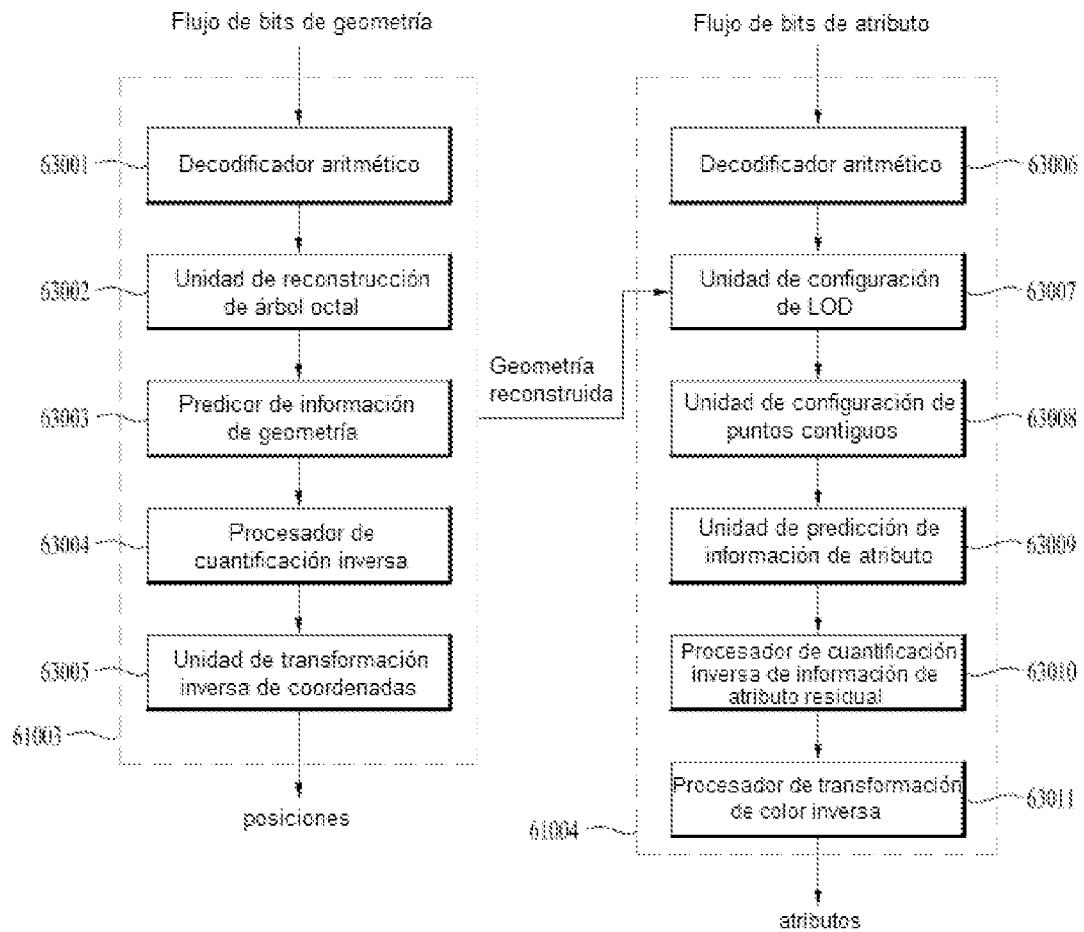


FIG. 31

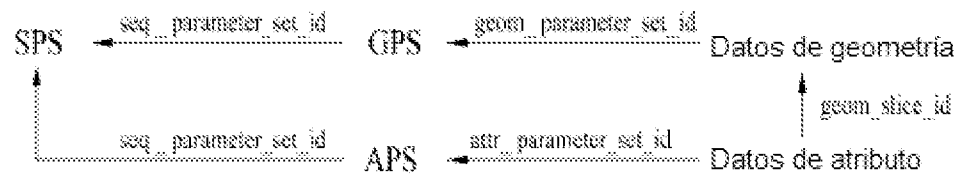


FIG. 32

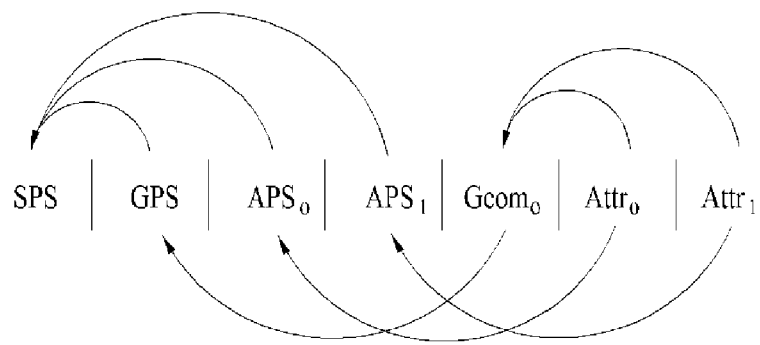


FIG. 33

scq_parameter_set_rbsp() {	Descriptor
profile_idc	u(8)
profile_compatibility_flags	u(24)
level_idc	u(8)
sps_bounding_box_present_flag	u(1)
if(sps_bounding_box_present_flag){	
sps_bounding_box_offset_x	se(v)
sps_bounding_box_offset_y	se(v)
sps_bounding_box_offset_z	se(v)
sps_bounding_box_offset_factor	ue(v)
sps_bounding_box_offset_width	ue(v)
sps_bounding_box_offset_height	ue(v)
sps_bounding_box_offset_depth	ue(v)
}	
sps_source_scale_factor[Ed. TMC13 v6 uses float, but integer is preferred.]	u(32)
sps_seq_parameter_set_id	ue(v)
sps_num_attribute_sets	ue(v)
for(i=0; i<sps_num_attribute_sets; i++) {	
attribute_dimension[i]	ue(v)
attribute_instance_id[i]	ue(v)
attribute_bitdepth[i]	ue(v)
attribute_cicp_colour_primaries[i]	ue(v)
attribute_cicp_transfer_characteristics[i]	ue(v)
attribute_cicp_matrix_coeffs[i]	ue(v)
attribute_cicp_video_full_range_flag[i]	u(1)
known_attribute_label_flag[i]	u(1)
if(known_attribute_label_flag[i]){	
known_attribute_label[i]	ue(v)
else	
attribute_label_four_byte[i]	u(32)
nearest_neighbour_max_range[i]	ue(v)
nearest_neighbour_min_range[i]	ue(v)
}	
sps_extension_present_flag	u(1)
if(sps_extension_present_flag)	
while(more_data_in_byte_stream())	
sps_extension_data_flag	u(1)
byte_alignment()	
}	

FIG. 34

geometry_parameter_set() {	Descriptor
gps_geom_parameter_set_id	ue(v)
gps_seq_parameter_set_id	ue(v)
gps_box_present_flag	u(1)
if(gps_box_present_flag){	
gps_gsh_box_log2_scale_present_flag	u(1)
if(gps_gsh_box_log2_scale_present_flag == 0)	
gps_gsh_box_log2_scale	ue(v)
}	
unique_geometry_points_flag	u(1)
neighbour_context_restriction_flag	u(1)
inferred_direct_coding_mode_enabled_flag	u(1)
bitwise_occupancy_coding_flag	u(1)
adjacent_child_contextualization_enabled_flag	u(1)
log2_neighbour_avail_boundary	ue(v)
log2_intra_pred_max_node_size	ue(v)
log2_trisoup_node_size	ue(v)
gps_extension_present_flag	u(1)
if(gps_extension_present_flag)	
while(more_data_in_byte_stream())	
gps_extension_data_flag	u(1)
byte_alignment()	
}	

FIG. 35

attribute_parameter_set() {	Descriptor
aps_attr_parameter_set_id	ue(v)
aps_seq_parameter_set_id	ue(v)
attr_coding_type	ue(v)
aps_attr_initial_qp	ue(v)
aps_attr_chroma_qp_offset	se(v)
aps_slice_qp_delta_present_flag	u(1)
isLifting = (attr_coding_type == 0 attr_coding_type == 2) ? 1 : 0	
if(isLifting) {	
lifting_num_pred_nearest_neighbours	ue(v)
lifting_max_num_direct_predictors	ue(v)
lifting_search_range	ue(v)
lifting_lod_regular_sampling_enabled_flag	u(1)
lifting_num_detail_levels_minus1	ue(v)
for(idx = 0; idx <= num_detail_levels_minus1; idx++) {	
if(lifting_lod_decimation_enabled_flag)	
lifting_sampling_period[idx]	ue(v)
else	
lifting_sampling_distance_squared[idx]	ue(v)
}	
nearest_neighbour_max_range	ue(v)
nearest_neighbour_min_range	ue(v)
different_nn_range_in_tile_flag	u(1)
}	
if(attr_coding_type == 0)	
lifting_adaptive_prediction_threshold	ue(v)
lifting_intra_lod_prediction_num_layers	ue(v)
}	
aps_extension_present_flag	u(1)
if(sps_extension_present_flag)	
while(more_data_in_byte_stream())	
aps_extension_data_flag	u(1)
byte_alignment()	
}	

FIG. 36

attribute_parameter_set() {	Descriptor
aps_attr_parameter_set_id	ue(v)
aps_seq_parameter_set_id	ue(v)
attr_coding_type	ue(v)
aps_attr_initial_qp	ue(v)
aps_attr_chroma_qp_offset	se(v)
aps_slice_qp_delta_present_flag	u(1)
isLifting = (attr_coding_type == 0 attr_coding_type == 2) ? 1 : 0	
if(isLifting) {	
lifting_num_pred_nearest_neighbours	ue(v)
lifting_max_num_direct_predictors	ue(v)
lifting_search_range	ue(v)
lifting_lod_regular_sampling_enabled_flag	u(1)
lifting_num_detail_levels_minus1	ue(v)
for(idx = 0; idx <= num_detail_levels_minus1; idx++) {	
if(lifting_lod_decimation_enabled_flag)	
lifting_sampling_period[idx]	ue(v)
else	
lifting_sampling_distance_squared[idx]	ue(v)
}	
different_nn_range_in_tile_flag	u(1)
different_nn_range_per_lod_flag	u(1)
if(!different_nn_range_per_lod_flag) {	
nearest_neighbour_max_range	ue(v)
nearest_neighbour_min_range	ue(v)
} else {	
for(idx = 0; idx <= num_detail_levels_minus1; idx++) {	
nearest_neighbour_max_range[idx]	ue(v)
nearest_neighbour_min_range[idx]	ue(v)
}	
}	
}	
if(attr_coding_type == 0)	
lifting_adaptive_prediction_threshold	ue(v)
lifting_intra_lod_prediction_num_layers	ue(v)
}	
aps_extension_present_flag	u(1)
if(sps_extension_present_flag)	
while(more_data_in_byte_stream())	
aps_extension_data_flag	u(1)
byte_alignment()	
}	

FIG. 37

tile_parameter_set() {	Descriptor
num_tiles	ue(v)
for(i = 0; i < num_tiles; i++) {	
tile_bounding_box_offset_x[i]	se(v)
tile_bounding_box_offset_y[i]	se(v)
tile_bounding_box_offset_z[i]	se(v)
tile_bounding_box_size_width[i]	ue(v)
tile_bounding_box_size_height[i]	ue(v)
tile_bounding_box_size_depth[i]	ue(v)
if (different_nn_range_in_tile_flag == true) {	
nearest_neighbour_max_range[i]	ue(v)
nearest_neighbour_min_range[i]	ue(v)
different_nn_range_in_slice_flag[i]	u(1)
if (different_nn_range_in_slice_flag[i] == true) {	
nearest_neighbour_offset_range_in_slice_flag[i]	u(1)
}	
}	
}	
byte_alignment()	
}	

FIG. 38

geometry_slice_bistream() {	Descriptor
geometry_slice_header()	
geometry_slice_data()	
}	

FIG. 39

geometry_slice_header() {	Descriptor
gsh_geom_parameter_set_id	ue(v)
gsh_tile_id	ue(v)
gsh_slice_id	ue(v)
if(gps_box_present_flag) {	
if(gps_gsh_box_log2_scale_present_flag)	
gsh_box_log2_scale	ue(v)
gsh_box_origin_x	ue(v)
gsh_box_origin_y	ue(v)
gsh_box_origin_z	ue(v)
}	
gsh_max_node_size_log2	ue(v)
gsh_num_points	ue(v)
byte_alignment()	
}	

FIG. 40

geometry_slice_data() {	Descriptor
for(depth = 0; depth < MaxGeometryOctreeDepth; depth++) {	
for(nodeldx = 0; nodeldx < NumNodesAtDepth[depth]; nodeldx++) {	
xN = NodeX[depth][nodeldx]	
yN = NodeY[depth][nodeldx]	
zN = NodeZ[depth][nodeldx]	
geometry_node(depth, nodeldx, xN, yN, zN)	
}	
}	
if(log2_trioupsize > 0)	
geometry_trisoup_data()	
}	

FIG. 41

attribute_slice_bitstream() {	Descriptor
attribute_slice_header()	
attribute_slice_data()	
}	

FIG. 42

attribute_slice_header() {	Descriptor
ash_attr_parameter_set_id	uc(v)
ash_attr_sps_attr_idx	ue(v)
ash_attr_geom_slice_id	ue(v)
if (aps_slice_qp_delta_present_flag) {	
ash_qp_delta_luma	se(v)
ash_qp_delta_chroma	se(v)
}	
if (different_nn_range_in_slice_flag == true) {	
if (nearest_neighbour_offset_range_in_slice_flag == false) {	
nearest_neighbour_absolute_max_range	ue(v)
nearest_neighbour_absolute_min_range	ue(v)
} else {	
nearest_neighbour_max_range_offset	ue(v)
nearest_neighbour_min_range_offset	ue(v)
}	
}	
byte_alignment()	
}	

FIG. 43

attribute_slice_header() {	Descriptor
ash_attr_parameter_set_id	ue(v)
ash_attr_sps_attr_idx	ue(v)
ash_attr_gcom_slice_id	ue(v)
if (aps_slice_qp_delta_present_flag) {	
ash_qp_delta_luma	se(v)
ash_qp_delta_chroma	sc(v)
}	
if (different_nn_range_in_slice_flag == true) {	
different_nn_range_per_lod_flag	u(1)
if (!different_nn_range_per_lod_flag) {	
if(nearest_neighbour_offset_range_in_slice_flag == false) {	
nearest_neighbour_absolute_max_range	ue(v)
nearest_neighbour_absolute_min_range	ue(v)
} else {	
nearest_neighbour_max_range_offset	ue(v)
nearest_neighbour_min_range_offset	ue(v)
}	
} else {	
if(nearest_neighbour_offset_range_in_slice_flag == false) {	
for(idx = 0; idx <= num_detail_levels_minus1; idx++) {	
nearest_neighbour_absolute_max_range[idx]	ue(v)
nearest_neighbour_absolute_min_range[idx]	ue(v)
}	
} else {	
for(idx = 0; idx <= num_detail_levels_minus1; idx++) {	
nearest_neighbour_max_range_offset [idx]	ue(v)
nearest_neighbour_max_range_offset [idx]	ue(v)
}	
}	
}	
}	
}	

FIG. 44

attribute_slice_data() {	Descriptor
dimension = attribute_dimension[ash_attr_sps_attr_idx]	
zerorun	ae(v)
for(i = 0; i < pointCount; i++) {	
if(attr_coding_type == 0 && maxPredDiff[i] > lifting_adaptive_prediction_threshold && MaxNumPredictors > 1) {	
predIndex[i]	ae(v)
}	
if(zerorun > 0) {	
for(k = 0; k < dimension ; k++)	
values[k][i] = 0	
zerorun --	
}	
else {	
attribute_coding(dimension, i)	ae(v)
zerorun	ae(v)
}	
}	
byte_alignment()	
}	

FIG. 45

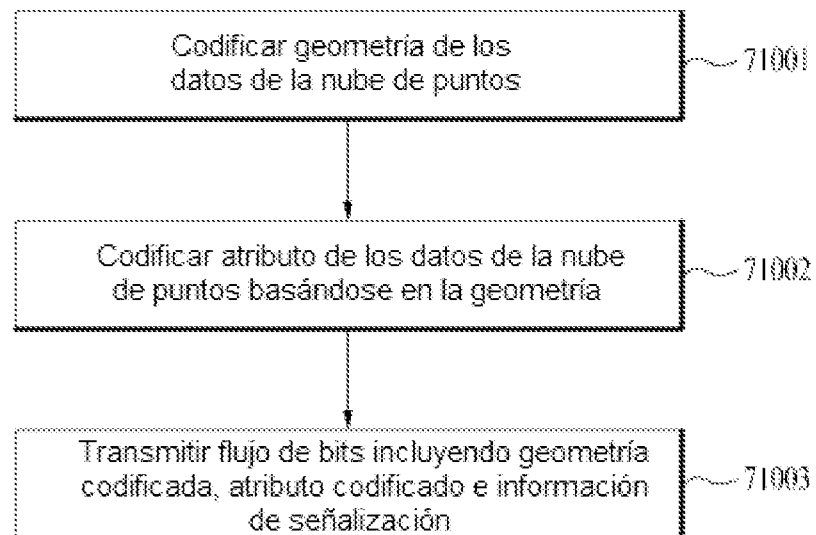


FIG. 46

