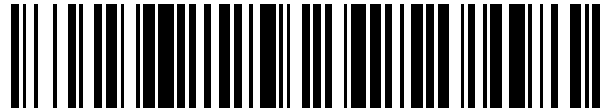


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 797 304**

51 Int. Cl.:

**G06Q 10/04** (2012.01)

**G06Q 10/06** (2012.01)

**G06Q 10/08** (2012.01)

**G01S 17/89** (2010.01)

**G06T 15/10** (2011.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **26.04.2017 PCT/IT2017/000085**

87 Fecha y número de publicación internacional: **16.11.2017 WO17195229**

96 Fecha de presentación y número de la solicitud europea: **26.04.2017 E 17732240 (1)**

97 Fecha y número de publicación de la concesión europea: **11.03.2020 EP 3455799**

54 Título: **Proceso y sistema para calcular el coste de materiales utilizables y consumibles para pintar vehículos de motor a partir del análisis de deformaciones en vehículos de motor**

30 Prioridad:

**09.05.2016 IT UA20163232**

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

**01.12.2020**

73 Titular/es:

**UESSE S.R.L. (100.0%)  
Via Pieragostini 73  
16151 Genova, IT**

72 Inventor/es:

**SIRI, FAUSTO**

74 Agente/Representante:

**SÁEZ MAESO, Ana**

ES 2 797 304 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

Proceso y sistema para calcular el coste de materiales utilizables y consumibles para pintar vehículos de motor a partir del análisis de deformaciones en vehículos de motor

5 La presente invención se refiere a un proceso para calcular el coste de materiales utilizables y consumibles para pintar vehículos de motor, a partir del análisis de deformaciones en los vehículos de motor.

Los procesos del tipo mencionado anteriormente son conocidos y ampliamente utilizados, pero tienen un problema importante. Ejemplo x de cálculo de coste son los documentos EP 0 867 233 que divulga una configuración de equipo de pintura y US 5,293,303 que divulga un sistema generador de estimación de costes.

10 De acuerdo con procesos anteriores, para calcular el coste de los materiales utilizables y consumibles, las casas de pintura se multiplican por un parámetro establecido convencional y arbitrariamente, que cambia de área a área (por ejemplo: en Milán el parámetro es de 20 euros por hora, en Turín 16 euros, en Nápoles 11 euros, etc.). Los fabricantes de productos de pintura tienen un software disponible que es capaz de calcular la cantidad de producto que se aplicará, pero calcularlo para toda la lámina de metal (por ejemplo, puerta o guardabarros): en realidad, esa cantidad ni siquiera se calcula, ya que las piezas del software tienen el llamado "coste por panel", que sin embargo no es correcto, ya que  
15 la parte dañada a pintar no es todo el panel, sino solo una parte del mismo.

La presente invención tiene como objetivo proporcionar un proceso y un sistema, de acuerdo con las respectivas reivindicaciones independientes, para calcular el coste de los materiales utilizables y consumibles del análisis de deformaciones en vehículos de motor, calculando el coste por centímetro cuadrado (o decímetro cuadrado) de los materiales que se utilizarán: de esta manera, teniendo el coste unitario por kg de pintura disponible, se logra también  
20 cuantificar con mayor precisión el coste del material que se utilizará, por ejemplo, en dos daños similares, pero en gatos con diferentes colores (mientras que ahora, dado que solo se calculan las horas de pintura multiplicadas por el parámetro, lo que se cuantifica es el mismo, independientemente del color).

Un objeto adicional de la presente invención es proporcionar un proceso y un sistema como se describió anteriormente, que se puede conectar operativamente a un medidor de tinte, de modo que, tomado del medidor de tinte, la fórmula  
25 para pintar un centímetro cuadrado, se puede determinar la cantidad exacta de producto que se comunicará al medidor de tinte, para pintar la porción seleccionada y nada más: esto es importante, ya que reduce los desechos y los impactos ambientales (eliminación de residuos no utilizados), además de proporcionar un ahorro de costes.

Se pretende que las reivindicaciones adjuntas sean una parte integral de la presente descripción.

30 Las características de la invención y las ventajas derivadas de la misma serán más claras a partir de la siguiente descripción detallada de las figuras adjuntas, en las que:

La Figura 1 es un diagrama de bloques del elemento principal del sistema de acuerdo con la presente invención;

La Figura 2 es un diagrama de flujo que muestra el esquema operativo del algoritmo recursivo; y

La Figura 3 es un diagrama de flujo, que describe cómo se verifica la posición de un punto con respecto a un triángulo.

35 En el sistema de la Figura 1, un operador inserta y carga en la memoria 8 de imágenes del vehículo una imagen tridimensional del vehículo dañado a través de medios para insertar datos o comandos 7; luego, el operador recuerda en la memoria de imágenes del vehículo dañado una imagen tridimensional del vehículo de muestra de la base de datos de imágenes de vehículos 4 de muestra, dicha imagen del vehículo de muestra correspondiente a la imagen del vehículo dañado, es decir, los dos vehículos deben ser de la misma marca, modelo y tipo. A través del programa de trabajo, la CPU compara la imagen tridimensional del vehículo dañado con la imagen tridimensional correspondiente  
40 del vehículo de muestra que identifica en comparación dichas dos imágenes: la ubicación del daño o la deformación y la detección de las regiones distorsionadas. Luego, la CPU a través del programa de trabajo, almacena los resultados de dicha comparación en la memoria de los programas de trabajo y, a través de algoritmos implementados desde el programa de trabajo, calcula el área y/o el volumen, y/o identifica la posición en el espacio del daño en el vehículo utilizando el algoritmo informático implementado desde el programa de trabajo en las regiones distorsionadas y/o en  
45 los resultados de dicha comparación almacenados en la memoria de los programas de trabajo. Por lo tanto, está claro que, para calcular ventajosamente el volumen de la región distorsionada o dañada, se utilizan imágenes tridimensionales, lo que permite que la CPU calcule automáticamente la profundidad de deformación en comparación con la imagen tridimensional correspondiente del vehículo de muestra.

50 Se proporciona además una unidad 9 de escaneo. El funcionamiento de la unidad 9 de escaneo proporciona un escaneo de un vehículo, preferiblemente con un láser, dando como resultado una matriz de datos de imagen o una imagen tridimensional del vehículo, que identifica los puntos que componen el contorno del vehículo.

En la selección en modo automático, el programa de trabajo compara la imagen del vehículo dañado con todas las imágenes de la base de datos de vehículos de muestra y recuerda de la base de datos de imágenes de vehículos de

muestra la imagen tridimensional del vehículo de muestra, que en comparación corresponde mucho más en los puntos con la imagen del vehículo dañado.

5 En el modo de selección automática, una vez que las dos imágenes del vehículo dañado y del vehículo de muestra están presentes en la memoria de los vehículos, a través del programa de trabajo, la CPU compara las imágenes tridimensionales del vehículo dañado y del correspondiente vehículo de muestra que detecta los puntos donde se produce una deformación o diferencia.

10 Los resultados de dicha comparación entre las imágenes del vehículo dañado y las imágenes del vehículo de muestra se insertan luego en la unidad de memoria del programa de trabajo, que, a través de la CPU para cada punto y/o área pequeña, identifica la deformación ocurrida, y calcula el perímetro, el área de la región dañada y/o el volumen de deformación y/o la profundidad de deformación a través de algoritmos conocidos.

De acuerdo con una solución preferida, es posible proporcionar que, a solicitud del programa, el operador inserte un acuerdo o comando de rechazo para dicho vehículo de muestra elegido de la CPU y exhibido en el monitor por el operador, con el fin de tener una confirmación de la selección automática de la CPU a través del programa de trabajo.

15 Con respecto a la definición/selección/cálculo de las áreas dañadas o distorsionadas y/o de los volúmenes, una dimensión y/o un área y/o un volumen de regiones dañadas de un tipo particular de vehículo se almacenan en la memoria de trabajo.

El programa de trabajo es tal que la CPU identifica a través de diversas formas, las partes que componen un vehículo y las combina con códigos de identificación predeterminados de las partes del vehículo, por ejemplo, identificando si el daño está en las puertas o en otras partes diferentes del vehículo.

20 De acuerdo con la presente invención, la CPU se proporciona para interactuar con una base de datos de partes que componen el vehículo, cada parte identificada por un código.

25 El programa de trabajo a través de la base de datos y la selección de partes es capaz de localizar el daño con referencia a las partes del vehículo que están deformadas de alguna manera dañadas; Por ejemplo, en las Figuras 9 y 10, el programa de trabajo ha identificado que la región de daño seleccionada se extiende en ambas puertas del vehículo mostrado.

Por lo tanto, la CPU a través del programa de trabajo produce y combina en cada parte del vehículo afectada por la deformación o daño un grado de severidad de la deformación proporcional a la deformación misma, es decir, la dimensión y/o el área y/o el volumen de la parte dañada.

30 La base de datos de partes del vehículo proporciona una combinación de una base de datos adicional de tiempos y/o costes para reemplazar y reparar partes, y de este modo el sistema a través de la CPU y el programa de trabajo permite evaluar el coste y las operaciones necesarias para reconstruir la parte en comparación con una base de datos de tiempos y costes de partes de vehículos de motor.

En particular, la evaluación de tiempos y costes se divide preferiblemente en tres pasos: reparar la lámina metálica, o trabajar la lámina metálica, pintar y ensamblar/desensamblar las piezas.

35 La base de datos de tiempos y costes de las partes del vehículo de motor para cada parte del vehículo de motor tiene preferiblemente una evaluación mínima y una evaluación máxima del tiempo y/o coste necesario para reparar la lámina metálica, o trabajar la lámina metálica y/o pintar y/o ensamblar/desensamblar partes y para cada parte del vehículo de motor además tiene una lista de partes para ser desensambladas/reensambladas para desensamblar/reensamblar cada una de las partes.

40 Por lo tanto, el sistema combina en cada parte dañada una evaluación de tiempo y/o coste preferiblemente dividida en dichos tres pasos (reparación de la lámina metálica, o trabajo de la lámina metálica, pintura y ensamblaje/desensamblaje de partes), siendo dicha evaluación elegida automáticamente, a través de algoritmos de trabajo dedicados, de la CPU en el rango que va desde dicha evaluación mínima hasta dicha evaluación máxima del tiempo y/o coste necesarios proporcionalmente al grado de severidad de la deformación o daño asignado a cada parte.

45 Por lo tanto, el sistema suma los costes/tiempos necesarios para los tres pasos de cada parte del vehículo dañado y señala los resultados obtenidos a través de dicha unidad para mostrar los resultados, preferiblemente a través de un monitor.

50 Para cada parte dañada, el sistema compara dicha suma de costes/tiempos necesarios para reparar con el coste/tiempo necesario para reemplazar completamente dicha parte dañada por una nueva parte, y preferiblemente elige automáticamente reemplazar dicha parte dañada o distorsionada o si reparar la misma parte eligiendo el coste/tiempo mínimo.

También en este caso, el sistema muestra los resultados obtenidos a través de unidades de visualización de resultados, posiblemente solicitando al operador una confirmación de la elección preferida.

También es posible proporcionar el sistema conectado a una base de datos de partes de vehículos de motor en un almacén para que el sistema compare las partes que se reemplazarán con las partes en el almacén, produciendo un reporte con partes disponibles/partes no disponibles, y posiblemente produciendo un pedido en papel o electrónicamente para suministrar al almacén partes tomadas para reemplazos y/o partes no disponibles.

- 5 La invención proporciona además un proceso para identificar, analizar y estimar automáticamente deformaciones en vehículos de motor, comprendiendo dicho proceso los siguientes pasos:
- cargar datos de imagen relevantes para al menos una imagen tridimensional de un vehículo dañado en la memoria de imágenes de un vehículo;
  - 10 - en la memoria de imágenes de los vehículos dañados, recordando los datos de imagen de al menos una imagen tridimensional de un vehículo de muestra de una base de datos de imágenes de vehículos de muestra, siendo dicha imagen del vehículo de muestra la imagen de un tipo de vehículo correspondiente al tipo de vehículo dañado;
  - mostrar datos de imagen relevantes para la imagen del vehículo dañado y el tipo correspondiente de vehículo no dañado;
  - 15 - comparar automáticamente la imagen tridimensional del vehículo dañado con la imagen tridimensional correspondiente del vehículo de muestra que identifica, mediante una comparación automática entre dichas dos imágenes: posición de daño o deformación y detección de las regiones distorsionadas;
  - seleccionar a través de herramientas gráficas para delimitar o señalar las regiones dañadas o distorsionadas identificadas por la comparación automática en al menos una de las dos imágenes;
  - 20 - almacenar píxeles de imagen y datos de imagen respectivos seleccionados como correspondientes a las regiones dañadas o distorsionadas como resultado de dicha comparación en una memoria de un programa de trabajo;
  - calcular el perímetro, área y/o volumen de la región o regiones dañadas o distorsionadas y/o parámetros morfológicos adicionales a través de algoritmos implementados por un programa de trabajo, y/o identificar la posición en el espacio del daño en el vehículo utilizando el algoritmo informático implementado desde el programa de trabajo sobre regiones distorsionadas y/o sobre los resultados de dicha comparación almacenados en la memoria de los programas de trabajo;
  - 25 - calcular un grado de severidad de deformación y asignar dicho grado de severidad de deformación a cada región dañada o distorsionada;
  - calcular los tiempos de trabajo y los costes para reparar el área dañada o distorsionada, a partir de dicho perímetro, área y/o volumen y/o parámetros morfológicos calculados, utilizando una base de datos de tiempos y costes de reparación disponibles para cada tipo de vehículo; y
  - 30 - producir una imagen virtual del vehículo de muestra, que se compone del conjunto de imágenes virtuales de las partes estructurales individuales del vehículo, que forman unidades estructurales independientes, es decir, partes individuales del vehículo, cuya imagen virtual se muestra como imagen del vehículo ensamblado o como una imagen del vehículo despiezado, cada parte estructural del vehículo se identifica unívocamente mediante un código de identificación, y corresponde a la misma parte estructural del vehículo dañado;
  - 35 en el paso de calcular el perímetro, el área y/o el volumen de la región o regiones dañadas o distorsionadas y/o parámetros morfológicos adicionales, y/o identificar la posición en el espacio del daño en el vehículo se realiza a través de los siguientes pasos:
  - Adquirir datos mediante escaneo láser, en el que la tecnología de escaneo láser adquiere digitalmente objetos tridimensionales de varios tamaños como nubes de puntos. La descripción digital geométrica del objeto es discreta:
  - 40 cuanto mayor sea la resolución establecida para la adquisición, más densa será la nube de puntos y, por lo tanto, el detalle que se mostrará. Cada punto está definido por una posición espacial con coordenadas x y z con respecto al punto de origen representado por la posición del escáner y, si el escáner es capaz de detectar también el mapeo fotográfico del objeto (a través de una cámara fotográfica digital incorporada), desde el valor RGB, por lo tanto, desde el valor cromático del punto adquirido. Con respecto a los tipos de escáner, los escáneres láser 3D pueden emplear
  - 45 fundamentalmente dos tipos de medición: el tiempo de vuelo, para objetos grandes espaciados por 2 m a más de 150 m (palacios, cuadrados, monumentos), o la triangulación óptica, para escaneos de resolución muy alta de objetos con tamaños reducidos colocados a una distancia de aproximadamente 0.6 a 25 m. La resolución de escaneo varía de acuerdo con el tipo de instrumentos y los objetivos de uso, desde valores submilimétricos (por debajo de 0,21 mm) del escáner, que utilizan la triangulación óptica, hasta valores milimétricos o centimétricos de escáneres de tiempo de
  - 50 vuelo.
  - datos de postprocesamiento: después de haber adquirido los datos, se realizan una serie de operaciones que permiten la visualización en 3D para la adquisición realizada.

A continuación, se incluyen los diversos pasos realizados, en orden de ejecución:

a) alineación de escaneos y grabación

5 Cada escaneo produce una nube de puntos, que describen parcialmente el objeto, tanto porque el rayo láser emitido graba el objeto desde un ángulo particular (punto de vista), y por lo tanto las superficies ocultas permanecerán ocultas también cuando se adquiere digitalmente, y porque el campo del escáner visual está limitado a un cierto ángulo (generalmente 40 ° -60 °). Por esta razón, es necesario realizar numerosos escaneos para obtener una cobertura total de adquisición.

10 Para poder alinear y volver a componer en una imagen global que muestre las nubes parciales de puntos obtenidos de cada escaneo, es necesario que los diversos escaneos tengan zonas superpuestas comunes (al menos aproximadamente 30 %) que comprendan puntos de objeto que sean fácilmente reconocibles. Al alinear (o grabar) las nubes parciales, será necesario señalar al software cuáles son los puntos de conexión comunes en los que los diferentes escaneos deben alinearse correctamente.

15 Para trabajar con una mejor precisión, se pueden usar "objetivos" u "blancos" con diversas formas y tamaños (esféricos, con disco plano, etc.), que generalmente se suministran con el escáner, y por lo tanto son parte de un sistema propietario, que los reconoce y clasifica automáticamente, a través del software, tan pronto como finaliza cada adquisición.

Un sistema de grabación potencialmente aún más preciso proporciona el uso de datos topográficos (detección de posición, de puntos bien reconocibles del objeto o blanco).

b) limpiar la nube de puntos y filtrar el ruido

Se eliminan los puntos aislados y se cancela el ruido producido por la desviación estándar del escáner.

20 c) triangulación y construcción de la malla

Esta es una operación muy delicada cuyo buen resultado depende de la calidad de los datos adquiridos.

d) cierre de agujeros de malla y corrección de caras anómalas

25 A menudo, debido al ruido o la ausencia de datos en las mallas, que deben integrarse, en el modelo final hay lugares vacíos o "agujeros". No son deseados desde un punto de vista estético y en lo que respecta al intento de reproducir el objeto real de la manera más fiel posible.

e) aplicar texturas al modelo 3D

El resultado final de aplicar una textura debe aparecer como una superposición perfecta y una correspondencia geométrica precisa, con respecto a la superficie real del objeto.

- mostrar el resultado obtenido

30 El propósito de la visualización es hacer que los datos adquiridos puedan interpretarse de la mejor manera posible.

En lo que respecta a la gestión y estructura de datos 3D, en los sistemas modernos de adquisición y gestión de datos, el escaneo 3D produce, de una manera más o menos directa, una serie de tripletes de coordenadas, cada una correspondiente a cada punto adquirido.

35 Los valores obtenidos se pueden organizar en una matriz rectangular, en realidad una imagen, llamada RI (imagen de rango, a saber, cuadrículas de distancias nxm que describen una superficie con coordenadas cilíndricas o cartesianas), en la que cada píxel asume un valor de profundidad con respecto a un punto de referencia o plano. La colocación del píxel en la cuadrícula determina la posición del punto en la superficie de referencia.

Los datos así organizados tienen la gran ventaja operativa de su compacidad. De hecho, para cada punto, es suficiente almacenar solo la información de profundidad para reconstruir el triplete cartesiano, que describe sus coordenadas.

40 Sin embargo, esta representación del objeto es intrínseca, ya que se refiere al punto de vista que lo generó: las oclusiones determinadas a partir de los objetos de primer plano con respecto a los objetos de fondo no pueden llenarse directamente. Para poder hacer que la escena sea extrínseca, es decir, hacerla utilizable desde cualquier punto de vista, generalmente se procede a través de algoritmos de triangulación del tipo Delaunay.

45 A través de este pasaje, cada píxel, después de haber sido declarado en sus tres coordenadas 3D, se conecta a otros dos píxeles a través de segmentos llamados bordes para formar una cara triangular, generalmente caracterizada por una normal, que determina su orientación en el espacio. Cada vértice es compartido por muchas caras.

En particular, la triangulación de Delaunay continúa con estos pasos: dados n puntos en un plano, se unen con segmentos para que cada región dentro del Casco Convexo sea un triángulo.

La triangulación de Delaunay es un gráfico plano. Para construir la triangulación de Delaunay, es suficiente trazar un segmento entre cada par de puntos, que comparten un arco. Este proceso divide la envoltura convexa en triángulos. Cada punto se conecta a través de un arco con sus puntos más cercanos.

- gestionar datos en el programa utilizado a través de mallas de polígonos

- 5 Los datos procedentes de diversos escaneos se recopilan en una estructura de datos creada y gestionada por el programa para gestionar adquisiciones 3D en las que se ha trabajado. Las mallas han sido calculadas a partir de nubes de puntos y luego han sido sometidas a triangulación. El triángulo es el polígono convexo más simple y debido a esto se elige como referencia.

- 10 La triangulación permite realizar una serie de operaciones gráficas que no serían posibles solo con las nubes de puntos, como por ejemplo aplicar superficies, colores y texturas, área de cálculo, perímetro, rellenar agujeros, eliminar partes irregulares y eliminar imprecisiones de datos detectados por el escáner.

Las superficies más complejas necesitan más triángulos para ser representados. Cuanto mayor sea el número de triángulos, mejor será la capacidad del modelo para representar los detalles.

- 15 Una malla de polígonos se compone de una colección de lados, vértices y polígonos conectados entre sí de manera que cada lado sea compartido por dos polígonos como máximo. Un lado conecta dos vértices, y un polígono es una secuencia cerrada de lados. Un lado puede ser compartido por dos polígonos adyacentes, un vértice es compartido por al menos dos lados, y cada lado finalmente debe ser parte de algún polígono.

- 20 Una malla de polígonos se puede representar de diferentes maneras; además, se pueden usar diferentes representaciones también dentro de una sola aplicación. De hecho, es posible elegir una representación para almacenamiento externo, una para uso interno y otra para crear interactivamente la malla.

Los criterios básicos para evaluar las diferentes representaciones son el coste en el tiempo y en el espacio de las representaciones mismas. Las operaciones típicas en una malla de polígonos son:

encontrar todos los lados incidentes en un cierto vértice;

ubicar los vértices conectados por un lado;

- 25 encontrar los lados de un polígono;

mostrar la malla;

identificar posibles errores en la representación (por ejemplo, vértices, o lados, o polígonos faltantes).

En lo que respecta a la representación de mallas de polígonos, existen tres tipos posibles de representaciones de mallas de polígonos.

- 30 En la representación explícita, cada polígono está representado por una lista de coordenadas de vértices:  $P = ((x_1, y_1, z_1), (x_2, y_2, z_2), \dots, (x_n, y_n, z_n))$

- 35 Los vértices se almacenan en el orden de acuerdo con el cual uno quisiera encontrarse con ellos al moverse a lo largo del polígono. En otras palabras, los pares de vértices siguientes están conectados por lados; Además, hay un lado que conecta el último vértice de la lista con el primero. Para un solo polígono, esta representación es eficiente en el espacio; para una malla de polígonos, en cambio, se emplea un poco de espacio inútilmente, porque las coordenadas de los vértices compartidos se almacenan dos veces. Además, no hay una representación explícita de los lados y vértices compartidos. Por ejemplo, para arrastrar un vértice junto con todos los lados incidentes sobre él, en una aplicación interactiva típica, uno tiene que encontrar todos los polígonos, que comparten el vértice seleccionado. Esta búsqueda requiere diferentes comparaciones entre las coordenadas de los vértices de un polígono y las de todos los demás polígonos. El método más eficiente para llevar a cabo esta tarea sería ordenar todas las  $N$  coordenadas, operación que requiere, en el mejor de los casos, un tiempo del orden de  $N \log_2 N$ .

- 45 Además, también existe la posibilidad de que, debido al redondeo debido al cálculo, el mismo vértice pueda tener coordenadas diferentes en cada polígono. En cuanto a la visualización, esta representación requiere la transformación de cada vértice y el recorte de cada lado de cada polígono. Además, los lados compartidos se dibujan dos veces, y esto puede insertar inconvenientes debido a la sobreescritura. En los polígonos definidos usando los punteros a las listas de vértices, cada vértice de la malla de polígonos se almacena en la lista de vértices  $V = ((x_1, y_1, z_1), (x_2, y_2, z_2), \dots, (x_n, y_n, z_n))$  solo una vez.

Por lo tanto, un polígono se define mediante una lista de punteros a la lista de vértices.

- 50 Esta representación tiene varias ventajas. En primer lugar, garantiza un fuerte ahorro de espacio en la memoria. Además, las coordenadas de un vértice pueden modificarse fácilmente. En el otro extremo, sin embargo, todavía es

difícil, desde el punto de vista de los cálculos, encontrar los polígonos que comparten un lado, y los lados compartidos se vuelven a dibujar dos veces.

5 Estos dos problemas pueden eliminarse representando explícitamente los lados, como ocurre en la representación a través de punteros a listas de lados. En este caso, todavía hay una lista V de vértices, pero el polígono está representado por una lista de punteros a una lista de lados, en la que cada lado aparece solo una vez. Además, al lado de cada lado de la lista, hay dos punteros al par de vértices, almacenados en la lista de vértices, que definen sus extremos, y uno o dos punteros a los polígonos a los que pertenece el lado.

Por lo tanto, se describe un polígono de la lista  $P = (L_1, L_2, \dots, L_n)$  y un lado de la lista  $E = (V_1, V_2, P_1, P_2)$ . Cuando un lado pertenece a un solo polígono  $P_1$  o  $P_2$  será nulo.

10 Los polígonos se representan en la pantalla mostrando sus lados, para evitar operaciones de conversión de escaneo y recorte redundantes.

En ninguna de las tres representaciones, la determinación de los lados incidentes a un cierto vértice se puede hacer de manera eficiente: todos los lados deben ser inspeccionados.

15 Obviamente, es posible agregar información explícita para permitir determinar estas relaciones de una manera más eficiente.

- representación de datos en el programa de adquisición 3D

La elección que se ha hecho en el software de gestión de datos 3D es mantener una lista de lados, una lista de vértices, una lista de triángulos del objeto y una lista opcional de triángulos que apuntan a cada vértice, creados e ideados para las necesidades de procesamiento y gestión de datos.

20 Para eliminar la redundancia de los vértices, lo primero que se ha hecho es construir una lista de los vértices (sin repeticiones) usando una matriz "Vt".

De esta manera, se elimina la duplicación de los vértices, pero no en los lados: un lado que pertenece a dos triángulos se almacena dos veces.

Las búsquedas de incidencia continúan siendo complejas.

25 Un segundo paso ha sido eliminar la redundancia de los lados. Se ha creado una lista de los lados (matriz "Et"), ordenada en v1 y luego en v2, cada una compuesta de dos enteros que indican la posición, en la matriz de vértices, de los dos vértices incidentes en el lado.

Por ejemplo, el lado  $l_1$  apunta a los dos vértices a y b, el lado  $l_2$  apunta a los vértices b y c y así sucesivamente.

30 En lo que respecta a la lista de triángulos ("Tt"), contiene todos los triángulos en una matriz de polígonos "Tpol". Cada triángulo tiene una información sobre su normalidad (componentes  $N_x, N_y, N_z$ ), los índices de textura (si están presentes) y una matriz que contiene tres enteros que representan las posiciones en la matriz de vértices "Vt" de los vértices que pertenecen al triángulo.

En lo que respecta a la lista opcional adicional de triángulos que apuntan a cada vértice, está compuesta por un vector de enteros que contiene los índices de los vértices utilizados por cada triángulo.

35 - corregir los defectos de adquisición con la reconstrucción de áreas faltantes

Los datos obtenidos mediante escaneos de modelos 3D típicamente contienen partes faltantes y lugares vacíos o "agujeros".

Esto puede ser causado por escaneos con artefactos o por cavidades demasiado profundas en el modelo que no pueden observarse con un ángulo de escaneo particular.

40 Además, los lugares vacíos pueden ser causados también por una baja reflexión (por ejemplo, partes que reflejan el rayo láser debido al material del que están compuestos), por límites que se ocupan de posicionar el láser del escáner o simplemente por la falta de vistas.

En lo que respecta a las técnicas para reconstruir superficies en modelos escaneados, la mayoría de los trabajos realizados para reconstruir las áreas faltantes se basa en una representación volumétrica de los objetos.

45 El llenado de agujeros puede realizarse como una operación de postprocesamiento, aplicado después de la reconstrucción de la superficie o puede integrarse en el algoritmo de reconstrucción de la superficie. En este segundo caso, debe hacerse una distinción entre los algoritmos de reconstrucción que operan en mallas conectadas de un conjunto de muestras y algoritmos que operan en nubes de puntos en 3D.

- llenar como postprocesamiento

Para rellenar áreas de una superficie recién reconstruida, un enfoque ampliamente utilizado es triangular cada componente conectado de la superficie, con el fin de obtener una malla poligonal y solo después llenar cada lugar vacío con un cierto algoritmo elegido.

- reconstrucción basada en mallas con relleno de lugares vacíos

- 5 En tal caso, cada escaneo se trata como una imagen de rango. Un algoritmo de este tipo, ampliamente difundido en la literatura, es el creado por Curless y Levoy (VRIP). Con este método, el espacio se representa con un conjunto de cubos elementales (vóxeles) y no con elementos de superficie (píxeles). La idea que subraya cada método volumétrico es asignar una función  $D(x)$  a puntos en el espacio, muestreándola junto a los vóxeles en los que se divide el espacio y extrayendo una superficie definida implícitamente por  $D(x) = s$ , donde  $s$  es un umbral dado. La función  $D(x)$  es la distancia con signo entre el punto  $x$  en el espacio y la superficie del objeto.

10 El signo señala la posición del punto con respecto al objeto: los puntos internos y externos tendrán signos opuestos. Cuando se calcula la función  $D(x)$ , la isosuperficie se puede extraer fácilmente, es decir, la superficie con umbral  $s = 0$ , por lo tanto, a una distancia nula del objeto. Corresponde a la mejor aproximación del modelo buscado.

$$D(\mathbf{x}) = \frac{\sum w_i(\mathbf{x})d_i(\mathbf{x})}{\sum w_i(\mathbf{x})}$$

- 15 
$$W(\mathbf{x}) = \sum w_i(\mathbf{x})$$
 donde  $D(x)$  es la función implícita que se busca y  $W(x)$  es la función acumulativa de los pesos asignados a cada vértice de la superficie.  $x$  es el punto cuya distancia desde la superficie debe calcularse;  $d(x)$  es la distancia entre el punto  $x$  y la malla  $M$ ,  $w_i(x)$  es el peso asociado con el punto  $p$  íésimo, la intersección entre la malla y la línea recta que comienza desde el sensor del sistema de medición y pasa por el punto  $x$ .

- 20 A los vóxeles del volumen que contiene el objeto, se le asigna uno de los tres estados: no visible, vacío o siguiente a la superficie. Los pasos del algoritmo son los siguientes:

1. Todos los vóxeles se inicializan al estado no visible.
2. Se asigna una función de distancia a los vóxeles junto a la superficie, como se describió anteriormente.
3. Se siguen las líneas, que unen el sensor de medición a los puntos de la malla y el estado de vacío se asigna a los vóxeles fuera de la superficie. Este proceso se llama talla del espacio.
4. Además de extraer una isosuperficie a una distancia  $d(x) = 0$ , se extrae una superficie entre vóxeles con estado vacío y vóxeles con estado no visible.

- 30 El resultado es obtener una intersección de todos los puntos reconocidos como internos a la superficie, a los que se asigna el estado no visible. La superficie final se extrae no solo donde  $d(x) = 0$ , sino también entre los puntos evaluados como externos, o con estado vacío, y aquellos internos o con un estado no visible.

- Reconstruir a partir de nubes de puntos con relleno de lugares vacíos.

El tercer método trata con la unión de todos los escaneos como un único conjunto desorganizado de puntos 3D, que deben adaptarse a una superficie continua. De esta manera, se crea una sola superficie y los lugares vacíos se eliminan simultáneamente, integrando la información proveniente del escaneo láser.

- 35 En el caso de una pantalla volumétrica de tipo vóxel, en la literatura a menudo se usa una representación poligonal para la superficie obtenida a través de los algoritmos de los cubos de marcha y de los cubos divisores.

- corregir los defectos de adquisición en el programa de gestión de datos 3D

Para este propósito, se usa un algoritmo para llenar lugares vacíos creados para el programa de gestión de datos 3D.

- 40 No se ha utilizado un método volumétrico, pero se ha tenido en cuenta la información derivada de la triangulación de mallas y de los lados del borde calculados como líneas a las que apunta un solo triángulo.

Se han realizado varios algoritmos antes de llegar a una reconstrucción satisfactoria de las superficies.

Todos los algoritmos son capaces de llenar muchos agujeros presentes en la selección si pertenecen a un solo objeto. Si se seleccionan superficies que pertenecen a muchos objetos, se muestra una pantalla de error.

- 45 Para el funcionamiento de los diversos algoritmos, es necesaria una selección activa en la parte de la superficie, que comprende el error. En primer lugar, se calculan los lados del borde que se encuentran dentro de la selección realizada



manualmente, y se colocan en una matriz de lados que se pasa a una función externa para su procesamiento. Este último funciona para controlar si los lados pertenecen a un contorno cerrado y si existe un solo contorno cerrado o si hay muchos contornos en la selección.

5 Los lados están ordenados de modo que el vértice v1 del primer lado colocado en la matriz sea igual al vértice v2 del lado contiguo en la matriz.

Si el vértice v1 del lado contiguo al lado iésimo es v1, se invierte con v2 y, en consecuencia, v2 con v1, para poder realizar comprobaciones en lados consecutivos siempre en el vértice v1.

Lo que se pasa como referencia al algoritmo principal es una nueva matriz de lados ordenados que pertenecen a un solo contorno (si está presente).

10 El algoritmo recursivo utilizado no agrega puntos al objeto, pero cierra el error uniendo los vértices del contorno con una secuencia determinada, de tal manera que triangula el lugar vacío. El objetivo principal ha sido llenar las áreas de los agujeros, sin tener en cuenta la posible superposición de los lados del contorno del error.

Con este algoritmo, como con los otros hechos, es posible detectar la presencia de ramas en un error, pero a diferencia de las siguientes, no se verifica la presencia de intersecciones entre los lados del contorno.

15 Los vértices que pertenecen a más de un contorno cerrado se consideran "ramas".

Los pasos principales de este algoritmo son:

la búsqueda del primer lado en la matriz de lados y del vértice v1 del lado consecutivo;

la formación de un nuevo triángulo usando v1 y v2 del lado iésimo y v1 del lado (iésimo + 1);

la adición iterativa de más triángulos para llenar toda el área del lugar vacío;

20 el cálculo de líneas normales a triángulos;

la copia de los índices de las texturas asociadas con los triángulos;

una verificación adicional para la búsqueda de otros lados del contorno y el posible relleno de las otras áreas cerradas.

25 Los problemas que ha dado este tipo de proceso están relacionados con la posible presencia de intersecciones entre los lados del borde como ya se mencionó y la unión externa vértice-vértice para formar un nuevo triángulo, sin tener en cuenta los ángulos obtusos internos.

Un segundo algoritmo que se está haciendo, llamado baricéntrico, se ocupa de llenar los lugares vacíos presentes en la superficie seleccionada calculando para cada uno de ellos el baricentro y luego creando nuevos triángulos dentro de los diversos agujeros.

30 El algoritmo es capaz de llenar todos los lugares vacíos presentes en la selección porque busca cíclicamente los contornos cerrados dentro de la malla, hasta que no se encuentre ninguno.

Cada lugar vacío se tiene en cuenta y se llena de forma diferente e independiente de los otros que están presentes.

1. El baricentro, en consecuencia, se calcula aplicando adecuadamente la operación de promedio a las coordenadas x, y y z de los diversos vértices del error. De este modo, se obtiene un nuevo vértice, que se agrega al vector de los vértices totales del objeto (Vt).

35 2. Luego, se calcula el baricentro de todos los nuevos triángulos creados virtualmente; solo en ese momento, se actualiza la lista total de triángulos totales del objeto (Tt).

3. Los únicos triángulos agregados son aquellos que tienen como vértice el baricentro calculado en el paso 2.

Estos cubren totalmente la superficie del lugar vacío y hacen que la malla del objeto sea más gruesa en esa área, con respecto a los triángulos obtenidos en el paso 1.

40 Como ya se indicó, el algoritmo, después de haber cerrado un error, busca otro posible error dentro de la selección realizada. De acuerdo con el mismo proceso del algoritmo anterior, si no se encuentra una rama, la superficie cerrada aguas arriba de la rama y la corriente abajo (si existen) se consideran dos agujeros diferentes.

Este algoritmo para reconstruir las superficies se ha insertado en la versión final del programa y su uso es aconsejable para los lugares vacíos de la forma circular, elíptica y similares.

En cambio, funciona incorrectamente si los agujeros tienen una forma de "U", ya que en este caso el baricentro del agujero cae en la parte de la superficie ya sometida a triangulación o de todos modos crea intersecciones entre los lados.

El algoritmo recursivo optimizado se inserta en la versión final del programa de gestión para datos 3D.

- 5 La principal diferencia con respecto al algoritmo recursivo anterior es la introducción de un dispositivo de creación de vóxel, a menudo mencionado en la literatura, para someter los polígonos a una triangulación.

El dispositivo de creación de vóxel permite llenar polígonos cóncavos simples, cubriendo así la mayoría de las situaciones posibles en una superficie agujereada.

- 10 La única condición particular que no se trata y resuelve es la presencia de un contorno cerrado dentro de otro, es decir, la presencia de una "isla" dentro de un agujero.

En cambio, se resuelven los siguientes casos:

- uniones externas causadas por la presencia de ángulos obtusos dentro de la superficie del agujero
- unión con posible intersección de los lados
- contornos auto intersectantes (señala el contorno incorrecto y se detiene).

- 15 Una complicación adicional, con respecto al algoritmo recursivo anterior, ha sido tener que tomar los puntos (vértices) del contorno cerrado de 3D a 2D para verificar posibles intersecciones entre los lados.

Los N vértices 3D que pertenecen a los lados verdes que se están considerando ya han sido ordenados, ya que son los vértices v1 de cada lado del contorno dispuestos en la matriz de lados ordenados (Ec).

Los vértices se proyectan en un plano buscado como el "mejor plano descrito por los N puntos 3D".

- 20 La ecuación de un plano genérico en el espacio es:

$$Ax + By + Cz + D = 0$$

donde A, B, C son los coeficientes que definen la línea normal al plano.

Los coeficientes se calculan de acuerdo con las siguientes expresiones:

25

$$A = \frac{1}{2} \sum_{i=0}^{N-1} \{(z_i + z_{i+1})(y_{i+1} - y_i)\}$$

$$B = \frac{1}{2} \sum_{i=0}^{N-1} \{(x_i + x_{i+1})(z_{i+1} - z_i)\}$$

$$C = \frac{1}{2} \sum_{i=0}^{N-1} \{(y_i + y_{i+1})(x_{i+1} - x_i)\}$$

Luego se estandarizan para obtener D en la forma correcta:

$$A = \frac{A}{\sqrt{(A^2 + B^2 + C^2)}} \quad B = \frac{B}{\sqrt{(A^2 + B^2 + C^2)}}$$

$$C = \frac{C}{\sqrt{(A^2 + B^2 + C^2)}}$$

$$D = \frac{-(Ax + By + Cz)}{\sqrt{(A^2 + B^2 + C^2)}}$$

30

Cuando se conocen las coordenadas 2D, se guardan en un vector que pertenece a una estructura de datos que tiene tres campos: dos flotantes x, y para las coordenadas y una id entera para almacenar el valor de v1 del lado, es decir, la posición del vértice en el vector de vértices totales del objeto (Vt).

- 5 La primera verificación realizada se ocupa de la autointersección de los lados del contorno. Si esta búsqueda proporciona un resultado negativo, se verifica si los vértices 2D se almacenan con la dirección del movimiento en sentido horario o antihorario en el contorno.

Si la dirección es en sentido antihorario, el vector se invierte.

Una función muy importante, creada para el funcionamiento correcto del dispositivo de creación de vóxel, se ocupa de calcular todos los N ángulos internos formados por cada par de lados del contorno.

- 10 Para almacenar cada valor de ángulo (medido en radianes) se usa una estructura similar a la de los vértices 2D.

En tal caso, el valor del ángulo se guarda en x y el valor del vértice relacionado con ese ángulo se guarda en id. El campo y se usa luego para el procesamiento.

Una vez que se han obtenido todos los ángulos, el primer ángulo de la matriz se compara con todos los demás ángulos, para buscar el que tenga el valor máximo.

- 15 Esta búsqueda se ha realizado para disminuir el número de ángulos obtusos desde los primeros pasos del algoritmo.

El primer triángulo que se creará será el que tenga ángulos de  $200^\circ$ ,  $83^\circ$  y  $77^\circ$ .

Cuando se conoce el ángulo máximo, se realizan más comprobaciones.

Se verifica que el nuevo segmento, uniendo el vértice relacionado con el ángulo máximo y el siguiente vértice de dos posiciones, no se ha construido fuera del agujero.

- 20 El segmento que se debe crear se analiza para agregar un triángulo a la estructura del objeto, de modo que no se interseque con ninguno de los lados verdes ya existentes del contorno.

Si una de estas comprobaciones proporciona un resultado positivo, se debe marcar el ángulo máximo actual (usando el campo y de la estructura de datos utilizada para los ángulos); se busca un nuevo ángulo máximo entre los restantes y el ciclo comienza nuevamente.

- 25 Si en cambio no ocurrieron casos particulares, se agregan los tres id, señalando los índices de los vértices necesarios para un nuevo triángulo, a una estructura de datos que contiene un vector de tres posiciones.

El punto 2D en la siguiente posición al que tiene el ángulo máximo debe eliminarse, porque a partir de ahora ya no será un vértice que pertenezca a los lados del contorno.

- 30 El algoritmo opera cíclicamente siempre que haya más de tres vértices de borde. Cuando solo se encuentran tres vértices, se insertan en la estructura que contiene los vértices en tripletes para los nuevos triángulos.

Estos últimos se agregarán a la lista de triángulos totales del objeto ( $T_t$ ). En la parte principal del algoritmo, se calculan las líneas normales de los nuevos triángulos agregados (en base al conocimiento de las líneas normales de los triángulos adyacentes al contorno) y los índices de textura (si están presentes).

El resultado final es una formación de vóxel aplicado a un polígono.

- 35 La mejora proporcionada por la triangulación con el dispositivo de creación de vóxel es clara.

Finalmente, la Figura 2 muestra un diagrama de flujo con el esquema operativo del algoritmo recursivo.

- 40 La integración de varias imágenes 3D adquiridas con un escáner láser es el paso central de muchas aplicaciones de software. Ensamblar las diversas vistas en una sola superficie poligonal es una operación que requiere un cálculo muy pesado, particularmente si las superficies 3D están modeladas con los vóxeles, como se puede encontrar a menudo en la literatura.

Localizar exactamente dónde los puntos de un objeto A se intersecan con los puntos de un objeto B en el espacio tridimensional es mucho más complejo con respecto al caso bidimensional.

En 3D, los lados de A pueden pasar por debajo o por encima de los lados de B sin que sus superficies se toquen mutuamente, dando de todos modos la idea visual de una intersección.

- 45 A continuación, se describen los métodos implementados para la reconstrucción de superficies a partir de imágenes de rango da.

Los trabajos publicados en la literatura que tratan sobre la integración de muchas mallas se pueden clasificar en dos grandes clases: método de estructura y métodos no estructurados.

La integración no estructurada se realiza uniendo todos los puntos derivados de los escaneos láser, luego suministrándolos a un procedimiento de reconstrucción poligonal.

- 5 La triangulación de Delaunay de un conjunto de puntos en el espacio 3D se ha propuesto como método básico para la reconstrucción.

Los métodos de integración estructurados usan información que trata de cómo se ha obtenido cada punto, usando los límites de error en la posición de un punto o información de proximidad entre los puntos dentro de una determinada imagen de rango. El algoritmo de Soucy y Laurendeau, ampliamente conocido en la literatura, utiliza una técnica de integración de estructuras para combinar múltiples imágenes.

10

Este es un método paramétrico porque realiza una reparametrización de las imágenes de rango.

Se ha visto que una imagen de rango es una colección de puntos  $(x; y; z)$  organizados como una matriz. Se asocia un plano con cada uno de ellos, que es un plan de imagen en el que se han proyectado los puntos del objeto, y una línea normal al plano, cuya dirección depende de la posición de un observador. A menudo es conveniente describir la superficie como una matriz de puntos distribuidos en una cuadrícula regular en un espacio parametrizado bidimensional. El paso del espacio tridimensional a un espacio bidimensional debe mantener las características de inyectividad: como máximo, un punto en el espacio 3D debe corresponder a un punto en el espacio 2D.

15

Con respecto a la parametrización en un espacio plano bidimensional, el plano en el que se produce la parametrización (también llamada interpolación) puede ser el plano de imagen de la imagen de rango o cualquier plano.

20

El algoritmo de Soucy y Laurendau se basa en subconjuntos canónicos de las imágenes de rango. El término subconjunto canónico de imágenes de rango  $K$  significa los puntos presentes en todas las imágenes  $K$ , obtenidos por la intersección de estas imágenes y ausentes en las otras imágenes.

Al modelar de manera distinta todos los subconjuntos, se obtienen porciones de superficie que describen completamente el objeto y que no se superponen mutuamente. Sin embargo, este enfoque tiene un problema: de esta manera, no es fácil modelar las áreas de superficie correspondientes a los bordes entre un subconjunto y otro, que resultan separados.

25

El algoritmo de Soucy y Laurendau es uno de los primeros algoritmos que han resuelto el problema de calcular una superficie a partir de diferentes imágenes de rango.

El algoritmo de Turk y Levoy es otro enfoque para el problema de integración.

30

La idea básica consiste en la eliminación de las partes superpuestas para obtener porciones de superficie que se utilizan para formar el modelo final.

El algoritmo se puede dividir en cuatro pasos siguientes: dado un par de rango de imágenes  $A$  y  $B$ :

1. Las áreas de las mallas que se superponen se eliminan.

35

2. Se realiza el cierre, es decir, las dos partes se unen mediante la operación de recorte "corte" de una de las mallas.

3. Se eliminan los triángulos más pequeños que se pueden obtener en el segundo paso.

4. Los vértices obtenidos de este modo se mueven hacia una "posición de consentimiento".

El proceso para eliminar áreas redundantes funciona de esta manera: se examinan los triángulos en los bordes de las mallas y se eliminan si son redundantes.

40

El proceso continúa hasta que no haya más triángulos redundantes. Dado un triángulo de  $A$ , es redundante si uno de sus vértices está dentro de una distancia límite de  $B$ . Se obtienen dos mallas que se superponen parcialmente solo al lado de los bordes. Ahora esta operación de cierre se une a las áreas de superficie obtenidas.

Los principales pasos operativos del algoritmo son:

45

- crear un nuevo objeto con las modificaciones debidas a la aplicación de los algoritmos para llenar los lugares vacíos o para unir dos objetos. Si no se ha elegido la creación de nuevos objetos, las modificaciones se realizan en el objeto/objetos iniciales;

integrando y uniendo dos mallas;

- elegir un umbral de distancia del plano único en el que cada triángulo del objeto A se establece con respecto a los puntos del objeto B. Se ha establecido a nivel de software que el umbral de distancia se refiere a un plano virtual arriba y un plano virtual debajo del plano del triángulo simple;

5 - considerar como extraíbles también los puntos de B que se encuentran fuera de los triángulos de A, pero dentro de una cierta distancia, fijada por dicho umbral. Los dos umbrales se han establecido en valores predeterminados que son los resultados adecuados para los escaneos utilizados como prueba;

- eliminar las partes de las superficies que se superponen.

Es necesario que solo dos objetos sean visibles para que el algoritmo funcione.

Todo esto se explica en detalle a continuación e incluye el uso de los umbrales previamente establecidos.

10 Es posible crear un solo objeto a partir de dos objetos diferentes previamente guardados, o al que se ha aplicado el algoritmo para eliminar áreas superpuestas. En este caso, es necesario que haya dos objetos visibles.

Si un solo objeto es visible, es posible unir las dos tiras del objeto que se muestra, de tal manera que cubra completamente esa parte de la superficie que, debido a la eliminación de la superposición, actualmente no tiene triangulación.

15 En cuanto al método de integración, para poder eliminar la superposición real y la superposición profunda, ha sido necesario trabajar en objetos sujetos a triangulación, porque es necesario tener información de la lista de triángulos totales de un objeto y de la lista de lados.

El algoritmo realizado consiste en una serie de pasos principales, cada uno de los cuales se explicará en detalle a continuación:

20 1. Retirar las partes superpuestas de dos mallas

2. Unir las dos mallas en una sola malla nueva

3. Rellenar el área interna de los dos contornos obtenidos con la adición de más triángulos

1. Retirar las partes superpuestas de dos mallas

25 Antes de poder unir dos adquisiciones o dos objetos previamente guardados en un solo archivo, es necesario eliminar los triángulos de una malla que se encuentran, dentro de una cierta distancia tolerada, dentro de la otra malla.

Como punto de partida para estudiar superficies superpuestas, se han creado e insertado dos superficies rectangulares perpendiculares al eje z en el programa de gestión de datos 3D.

Las dos superficies se han utilizado para tratar de eliminar las áreas superpuestas en el espacio 3D.

30 Si las dos superficies se ven frontalmente, parecen superpuestas mientras que, si están ligeramente giradas, aparecen como dos objetos diferentes que se encuentran en dos planos diferentes del espacio 3D.

Al igual que estas dos superficies que se superponen solo visualmente, también las adquisiciones obtenidas a través del escáner láser pueden intersectarse mutuamente solo visualmente.

35 Para comprender mejor lo que se entiende como intersección visual o superposición de profundidad, siempre se pueden tomar como referencia dos objetos A y B y uno puede pensar en construir una superficie arriba y una superficie debajo de A y luego verificar la presencia de puntos del contorno de B dentro del objeto que se está creando.

De esta manera se aclara la presencia y el uso de un umbral de distancia desde el plano.

Para eliminar más vértices de borde de B presentes fuera del objeto A, se introduce un segundo umbral.

Estos puntos están en estricta cercanía al contorno de A y debido a esto, si permanecen en el objeto final, pueden resultar un obstáculo para unir los contornos de los dos objetos que se realizarán después.

40 De acuerdo con un ejemplo de situación similar a la descrita, el segundo umbral sirve para permitir que un usuario también elimine algunos vértices externos. Con un cierto valor umbral, los vértices señalados con un círculo rojo se pueden eliminar. Esta operación es posible también después de haber realizado el algoritmo ya una vez; de hecho, al realizar el algoritmo nuevamente, se eliminarían los vértices externos, ya que se consideran un obstáculo para un buen resultado de unión.

45 Una diferencia del algoritmo de cierre de Turk y Levoy en el que solo se eliminan los vértices de uno de los dos objetos en el área superpuesta, en este algoritmo se eliminan los vértices de ambos objetos.

## ES 2 797 304 T3

Para poder realizar esta operación, los dos objetos se invierten cíclicamente, se pasan como referencia a la función que elimina la superposición. Los objetos visibles deben ser necesariamente solo dos para que sea posible usar el algoritmo descrito.

5 La primera operación que se realiza dentro del algoritmo es la copia local de los vértices de borde de un objeto y la copia de la lista de triángulos y de la lista de vértices del segundo objeto.

Tomando como referencia dos objetos A y B, el segundo objeto mencionado anteriormente no siempre será el objeto B, sino que será una vez B y luego A.

10 A diferencia de los algoritmos para reconstruir las áreas faltantes, en este caso no es necesario operar una determinada selección en la porción superpuesta, porque no se utiliza la información derivada de la lista de lados en la selección.

Por esta razón, los cálculos no son muy altos, ya que la verificación de la superposición debe realizarse para todos los triángulos de un objeto y para todos los lados del borde del otro objeto.

Después de haber copiado todos los vértices de borde de un objeto en un vector local y todos los vértices y los triángulos del otro objeto en otros vectores locales, se realiza una verificación preliminar.

15 De hecho, se verifica si la distancia entre el centro de gravedad del triángulo *j*-ésimo del segundo objeto y el vértice del borde *i*-ésimo del primer objeto es la mayor de una longitud *f* obtenida como:

$$f = c + d$$

donde *c* = distancia entre el centro de gravedad y el vértice del triángulo que está más alejado del centro de gravedad y *d* = umbral de distancia desde el plano.

20 Si este cálculo proporciona un resultado positivo, entonces el vértice *i*-ésimo se descarta para el triángulo en cuestión, de lo contrario se realiza el algoritmo, porque el vértice está dentro de una porción del espacio 3D restringido a una región cercana al triángulo.

Solo entonces el triángulo actual se proyecta de 3D a 2D (más precisamente se proyecta en el plano al que pertenecen sus tres vértices).

25 La proyección del triángulo en el plano se realiza utilizando la ecuación de un plano en el espacio 3D ( $Ax + By + Cz + D = 0$ ) y transportándolo, con la ayuda de matrices de transformación, las coordenadas 3D de vértices en coordenadas 2D.

30 Además, el vértice del contorno *i*-ésimo se proyecta en el plano del triángulo del objeto opuesto (el marcado por los coeficientes *A*, *B*, *C* y *D* calculados en el paso anterior) y se verifica si el vértice cae dentro de la región del espacio delimitada por el umbral de distancia desde el plano.

Si el vértice está dentro de esta región, la variable entera *a* perteneciente a ese vértice se coloca en -1 (se usará para la siguiente creación de los triángulos) y, en un vector adicional de enteros, se almacena -1 en la posición relacionada con este vértice.

35 De lo contrario, uno continúa con una verificación adicional en el vértice, probando si este último, además de pertenecer a la región del espacio delimitada por el "umbral de distancia del plano", también está dentro del triángulo.

Para saber si cierto punto está fuera o dentro del triángulo tomado como objeto, se realiza una función que verifica, comenzando por las coordenadas de los tres vértices del triángulo, cuando un punto es interno, externo o en el borde del mismo.

40 Desde el punto de vista algebraico, una línea recta divide el plano en tres conjuntos de puntos: dos semiplanos y la línea recta misma.

Los puntos del plano cartesiano se dividen en los tres conjuntos de acuerdo con el valor del polinomio  $P(x, y) = ax + by + c$ . Es decir, si  $P(x, y)$  produce un valor positivo, los puntos pertenecen a un semiplano, si son negativos pertenecen al semiplano opuesto, y si son nulos, se colocan en la línea recta con la ecuación  $P(x, y) = 0$ .

45 Dados dos puntos  $P_0(x_0, y_0)$  y  $P_1(x_1, y_1)$ , la ecuación canónica (o implícita) de la línea recta que pasa por los dos puntos es:

$$x(y_0 - y_1) - y(x_0 - x_1) + x_0 y_1 - x_1 y_0 = 0$$

El polinomio, primer miembro de la ecuación, es nulo en la línea recta, mientras que proporcionará valores positivos en un semiplano y valores negativos en el otro semiplano.

La función PuntoInternoTrg(std :: vector <punti2DF>trng) verifica si un punto p está afuera, en el borde o dentro del triángulo de vértices a, b y c.

La función PuntoInternoTrg recibe como argumentos las coordenadas de los vértices a, b, c del triángulo y las del punto p a verificar.

- 5 Con la función Segno se verifica si el punto p pertenece en cualquier caso al mismo semiplano que contiene el tercer vértice con respecto a la línea recta que pasa por los otros dos: el producto es positivo cuando p y el tercer vértice pertenece al mismo semiplano.

Los productos de las funciones Segno, registradas en las variables de control s1, s2 y s3, se utilizan para establecer la posición del punto p con respecto al triángulo de vértices a, b y c.

- 10 s1, s2, s3 de hecho contienen la posición de p con respecto a los tres lados.

Si el valor de las variables de control es negativo en al menos un caso, el punto p está fuera del triángulo.

Si los valores de verificación son todos positivos, el punto está fuera del triángulo. Finalmente, cuando ningún valor es negativo, pero en algún caso es nulo, el punto está en el borde del triángulo.

- 15 La figura 3 muestra un diagrama de flujo que describe cómo se verifica la posición de un punto con respecto a un triángulo.

Si el vértice es interno, la variable a del vértice se establece en -1.

Si el vértice no es interno, de todos modos, se verifica si el vértice está dentro de la distancia "soglia esterno trng". Para esto, se calcula una ecuación paramétrica para un segmento en el plano. Se calcula para cada lado del triángulo.

- 20 Esta serie de comprobaciones en el triángulo joteésimo del primer objeto se realiza para cada vértice del segundo objeto. Después de haber terminado los N vértices, el triángulo (joteésimo + 1) se analiza con todos los vértices que no aparecen con el valor -1 en el vector "verticieliminare".

Después de haber terminado las comprobaciones en los triángulos de un objeto, se crea su nueva lista de vértices y la lista de triángulos que quedó después de eliminar los diversos vértices.

- 25 Luego, los dos objetos A y B se invierten (intercambiando los punteros) y esta vez se verifica si los vértices de A se superponen a los triángulos B.

Para comprender cuándo debe finalizar la búsqueda de vértices superpuestos, en cada ciclo se compara si el tamaño del vector de triángulos totales del objeto modificado es el mismo del nuevo vector con los triángulos restantes. Si los dos tamaños son iguales, los vértices no se han eliminado del objeto que se va a "modificar" y, por lo tanto, la búsqueda ya no se realizará en el siguiente ciclo para ese objeto.

- 30 En cuanto a la visualización de objetos finales, pueden ocurrir dos situaciones:

- éste ha sido elegido para crear dos nuevos objetos y en tal caso los nuevos objetos se hacen visibles mientras que los primeros no se hacen visibles;
- las modificaciones se realizarán a los objetos originales.

- 35 A continuación, se incluyen algunos ejemplos de aplicación de diferentes valores umbral de la interfaz principal Riempi buchi Unioni.

De todos modos, el resultado es el de obtener dos nuevos objetos de acuerdo con los valores de umbral establecidos al principio que ahora deben unirse en un solo objeto.

2. Unir las dos mallas en una sola malla nueva

- 40 Para crear la superficie que falta en el área entre los contornos de los dos objetos, se ha decidido unir estos dos en un solo objeto nuevo que se agregará a la lista de objetos. Los dos objetos se vuelven no visibles, mientras que el nuevo objeto creado será el único visible.

3. Rellenar el área interna de los dos contornos obtenidos con la adición de más triángulos

Para integrar la porción de espacio entre los contornos de los dos objetos, permanecer vacío después de haber eliminado los vértices superpuestos, es posible agregar triángulos que cubran la diferencia creada entre las dos tiras.

- 45 Para hacer esto, es necesaria una selección de la región de espacio elegida.

La selección debe realizarse para comprender los lados verdes de dos contornos solamente o de un solo contorno en forma de "U".

Si se seleccionan más de dos ramas de borde, no hay área de espacio sujeta a triangulación y la irregularidad se señala en la consola generada por el programa principal cuando se abre.

- 5 La triangulación de la porción dentro de los contornos de los dos objetos se realiza usando el algoritmo recursivo descrito anteriormente.

La aplicación del algoritmo recursivo solo es posible en contornos cerrados.

Por esta razón, para llenar la porción entre las dos ramas, se añaden dos lados adicionales al vector de lados de borde:

- el creado al unir el primer vértice de la primera rama con el vértice final de la segunda rama más cercana al mismo;
- 10 - el obtenido al unir el último vértice de la primera rama con el vértice final restante de la segunda rama.

Si la selección se realiza en un contorno "U", el primer vértice se une con el último para crear el lado necesario para cerrar la región del espacio. Para no arriesgarse a unir dos vértices de un contorno rectilíneo, se realiza una comprobación de la longitud de la rama: si la distancia entre los dos vértices finales es inferior a la mitad de la longitud de la rama, se procede al cierre y a la triangulación.

- 15 Con respecto al relleno de lugares vacíos con una isla de puntos en el centro, ninguno de los algoritmos que se describen es capaz de eliminar lugares vacíos con partes de la superficie dentro de ellos, es decir, islas de puntos sujetos a triangulación en el centro de un agujero.

Al hacer el algoritmo de unión de los contornos, de hecho, ha sido posible hacerlo sin la falta de métodos previamente descritos.

- 20 Usando el algoritmo de unión de dos ramas verdes, se puede eliminar un lugar vacío que contiene una isla de puntos. Inicialmente, se puede proceder uniendo parte de los contornos del lugar vacío para obtener un agujero.

Luego, aplicando el algoritmo de relleno recursivo en la superficie ahora compuesta de un solo contorno cerrado, el lugar vacío se elimina totalmente.

El proceso inventivo comprende además los siguientes pasos:

- 25 - seleccionar el color del vehículo, o hacer un color personalizado e insertar sus valores relacionados;
- insertar datos relacionados con la capa de pintura de imprimación y la capa de pintura transparente;
  - insertar datos relacionados con otros materiales consumibles normalmente utilizados durante el proceso de pintura;
  - y
- 30 - calcular el coste de los materiales utilizables y consumibles para pintar vehículos de motor, a partir del análisis de las deformaciones en los vehículos de motor.

Con el proceso descrito anteriormente, se realiza una comparación entre los datos de imágenes relevantes con la imagen del vehículo dañado y los datos de imagen relevantes con la imagen del vehículo de muestra, identificando las regiones dañadas y/o distorsionadas que se identifican de la diferencia de los datos de imagen entre los datos de imagen relevantes para el vehículo dañado y los datos de imagen relevantes para el vehículo de muestra.

- 35 Además, la selección de los datos de imagen del vehículo de muestra de una base de datos de imágenes de vehículos de muestra se realiza automáticamente, la imagen cuyos datos de imagen son diferentes en una cantidad mínima de los datos de imagen del vehículo seleccionado que se selecciona como imagen del vehículo de muestra.

- 40 Los datos de imagen se componen de una matriz tridimensional de datos de imagen, cada elemento de dicha matriz está relacionado unívocamente con los píxeles de una imagen bidimensional de acuerdo con un plano que pasa a través de dicho píxel y la comparación de los datos de imagen del vehículo dañado y la comparación entre los datos de imagen del vehículo de muestra ancha se componen de la comparación de datos de imagen que en los dos conjuntos de datos de imagen tienen la misma posición.

- 45 En el proceso de la invención, se calcula el área y/o el volumen de la región de imagen que es relevante para los píxeles de imagen del vehículo que coinciden con la región dañada, región que se determina seleccionando medios y/o comparando automáticamente los datos de la imagen del vehículo dañado con los datos de la imagen del vehículo de muestra.

Además, se produce una imagen virtual de un vehículo de muestra, que se compone del conjunto de imágenes virtuales de partes estructurales individuales de un vehículo que forman unidades estructurales independientes, es decir, partes individuales de un vehículo, cuya imagen virtual se puede visualizar como imagen del vehículo



ensamblado o como imagen despiezada del vehículo y/o como imagen de transparencia del vehículo dañado en el que también se muestran las imágenes de partes ocultas, cada parte estructural del vehículo se identifica unívocamente mediante un código de identificación.

- 5 Y aún más, también se define al menos una vista externa del vehículo, preferiblemente diversas vistas externas del vehículo tomadas desde sus diferentes lados y/o en perspectiva y al menos una vista interna del vehículo, preferiblemente diversas vistas internas del vehículo de manera similar a las vistas externas, siendo posible seleccionar gráficamente una o más partes estructurales del vehículo para su identificación como partes dañadas o una o más partes estructurales del vehículo identificadas como partes dañadas a través de una comparación automática entre imágenes virtuales del vehículo de muestra e imágenes virtuales del vehículo dañado.
- 10 En particular, una o más imágenes del vehículo dañado se obtienen también desde su interior, es decir, una o más imágenes del vehículo dañado en su interior.

**REIVINDICACIONES**

1. Proceso para calcular el coste de materiales utilizables y consumibles para pintar vehículos de motor, a partir de un análisis de deformaciones en los vehículos de motor, dicho proceso comprende los siguientes pasos:

- 5 - cargar datos de imagen relevantes para al menos una imagen tridimensional de un vehículo dañado en la memoria de imágenes de un vehículo;
  - en la memoria de imágenes de los vehículos dañados, recordar los datos de imagen de al menos una imagen tridimensional de un vehículo de muestra de una base de datos de imágenes de vehículos de muestra, siendo dicha imagen del vehículo de muestra la imagen de un tipo de vehículo correspondiente al tipo de vehículo dañado
  - 10 - mostrar datos de imagen relevantes para la imagen del vehículo dañado y el tipo correspondiente de vehículo no dañado;
  - comparar automáticamente la imagen tridimensional del vehículo dañado con la imagen tridimensional correspondiente del vehículo de muestra identificando, mediante una comparación automática entre dichas dos imágenes: posición de daño o deformación y detectar las regiones distorsionadas;
  - 15 - seleccionar, a través de herramientas gráficas para delimitar o señalar, las regiones dañadas o distorsionadas identificadas a partir de la comparación automática en al menos una de las dos imágenes;
  - almacenar píxeles de imagen y datos de imagen respectivos seleccionados como correspondientes a las regiones dañadas o distorsionadas como resultado de dicha comparación en una memoria de un programa de trabajo;
  - 20 - calcular el perímetro, área y/o volumen de la región o regiones dañadas o distorsionadas y/o parámetros morfológicos adicionales a través de algoritmos implementados por un programa de trabajo, y/o identificar la posición en el espacio del daño en el vehículo usando el algoritmo de cálculo implementado por el programa de trabajo en regiones distorsionadas y/o en los resultados de dicha comparación almacenados en la memoria de los programas de trabajo;
  - calcular un grado de severidad de deformación y asignar dicho grado de severidad de deformación a cada región dañada o distorsionada;
  - 25 - calcular los tiempos de trabajo y los costes para reparar el área dañada o distorsionada, a partir de dicho perímetro, área y/o volumen y/o parámetros morfológicos calculados, utilizando una base de datos de tiempos y costes de reparación disponibles para cada tipo de vehículo; y
  - 30 - producir una imagen virtual del vehículo de muestra que se compone del conjunto de imágenes virtuales de las partes estructurales individuales del vehículo que forman unidades estructurales independientes, es decir, partes individuales del vehículo, cuya imagen virtual se muestra como imagen del vehículo ensamblado o como imagen del vehículo despiezado, cada parte estructural del vehículo se identifica unívocamente mediante un código de identificación, y corresponde a la misma parte estructural del vehículo dañado;
- caracterizado porque el paso de calcular el perímetro, área y/o volumen de la región o regiones dañadas o distorsionadas y/o parámetros morfológicos adicionales, y/o identificar la posición en el espacio del daño en el vehículo se realiza a través de los siguientes pasos:
- 35 - adquirir datos mediante escaneo láser;
  - postprocesar los datos, que comprende los subpasos de a) alinear los escaneos y la grabación, b) limpiar la nube de puntos y filtrar el ruido, c) triangular y construir la malla, d) cerrar los agujeros de la malla y corregir las caras anómalas, y e) aplicar las texturas al modelo 3D;
  - visualizar el resultado obtenido;
  - 40 - gestionar los datos en el programa utilizado a través de mallas de polígonos;
  - mostrar datos en el programa de adquisición 3D;
  - llenar como postprocesamiento;
  - reconstruir basado en mallas con relleno de lugares vacíos;
  - reconstruir a partir de nubes de puntos con relleno de lugares vacíos;
  - 45 y porque este comprende además los siguientes pasos:
  - seleccionar el color del vehículo, o hacer un color personalizado e insertar sus valores relacionados;
  - insertar datos relacionados con la capa de pintura de imprimación y la capa de pintura transparente;

- insertar datos relacionados con otros materiales consumibles normalmente utilizados durante el proceso de pintura; y
  - calcular el coste de los materiales utilizables y consumibles para pintar vehículos de motor, a partir del análisis de las deformaciones en los vehículos de motor.
- 5 2. Proceso de acuerdo con la reivindicación 1, caracterizado porque se realiza una comparación entre los datos de imágenes relevantes con la imagen del vehículo dañado y los datos de imagen relevante con la imagen del vehículo de muestra, identificando las regiones dañadas y/o distorsionadas que se identifican por la diferencia de los datos de imagen entre los datos de imagen relevantes para el vehículo dañado y los datos de imagen relevantes para el vehículo de muestra.
- 10 3. Proceso de acuerdo con la reivindicación 1 o 2, caracterizado porque la selección de los datos de imagen del vehículo de muestra de una base de datos de imágenes de vehículos de muestra se realiza automáticamente, la imagen cuyos datos de imagen son diferentes en una cantidad mínima de los datos de imagen del vehículo seleccionado siendo seleccionado como imagen del vehículo de muestra.
- 15 4. Proceso de acuerdo con una cualquiera de las reivindicaciones anteriores, caracterizado porque los datos de imagen están compuestos por una matriz tridimensional de datos de imagen, cada elemento de dicha matriz está relacionado de manera unívoca con los píxeles de una imagen bidimensional de acuerdo con una plano que pasa a través de dicho píxel y la comparación de datos de imagen del vehículo dañado y de datos de imagen del vehículo de muestra que se está construyendo a partir de la comparación de datos de imagen que en los dos conjuntos de datos de imagen tienen la misma posición.
- 20 5. Proceso de acuerdo con una cualquiera de las reivindicaciones anteriores, caracterizado porque se calcula el área y/o el volumen de la región de imagen relevante para los píxeles de imagen del vehículo que coinciden con la región dañada, cuya región se determina por medios de selección y/o comparando automáticamente los datos de imagen del vehículo dañado con los datos de imagen del vehículo de muestra.
- 25 6. Proceso de acuerdo con la reivindicación 1, caracterizado porque se define al menos una vista externa del vehículo, preferiblemente diversas vistas externas del vehículo tomadas desde sus diferentes lados y/o en perspectiva y al menos una vista interna del vehículo, preferiblemente diversas vistas internas del vehículo de manera similar a las vistas externas, siendo posible seleccionar gráficamente una o más partes estructurales del vehículo para su identificación como partes dañadas o una o más partes estructurales del vehículo identificadas como partes dañadas a través de una comparación automática entre imágenes virtuales del vehículo de muestra e imágenes virtuales del
- 30 vehículo dañado.
7. Sistema para calcular el coste de materiales utilizables y consumibles para pintar vehículos de motor, a partir de un análisis de deformaciones en los vehículos de motor de acuerdo con el proceso de acuerdo con cualquiera de las reivindicaciones anteriores, caracterizado porque comprende al menos una unidad central de procesamiento (CPU) o unidad lógica central con la que están asociados los siguientes:
- 35 - al menos un programa para realizar los pasos del proceso de acuerdo con cualquiera de las reivindicaciones anteriores,
- al menos una memoria para almacenar dicho programa,
  - al menos una memoria para almacenar datos de imagen de al menos un vehículo,
  - al menos una base de datos de imágenes tridimensionales de vehículos de muestra,
- 40 - medios para insertar datos o comandos alfanuméricos y/o gráficos,
- seleccionar medios para seleccionar regiones dañadas o distorsionadas en dicho vehículo, y
  - al menos una unidad para imprimir y/o mostrar resultados, como un monitor.
- 45 8. Sistema de acuerdo con la reivindicación 7, caracterizado porque los medios gráficos para seleccionar el píxel y los datos respectivos de las imágenes que coinciden con las regiones dañadas o distorsionadas de los vehículos en las imágenes mostradas se seleccionan dibujando una línea de borde cerrada que rodea la región de la imagen del vehículo que coincide con la región dañada e identifica los datos de imagen del subconjunto de píxeles que caen dentro de dicha línea de borde como píxeles de los datos de imagen correspondientes a la región dañada o distorsionada del vehículo, dicho sistema además comprende una unidad para detectar la imagen tridimensional de un vehículo, cuya unidad proporciona un conjunto bidimensional o tridimensional de los datos de imagen almacenados
- 50 como datos de imagen del vehículo dañado.

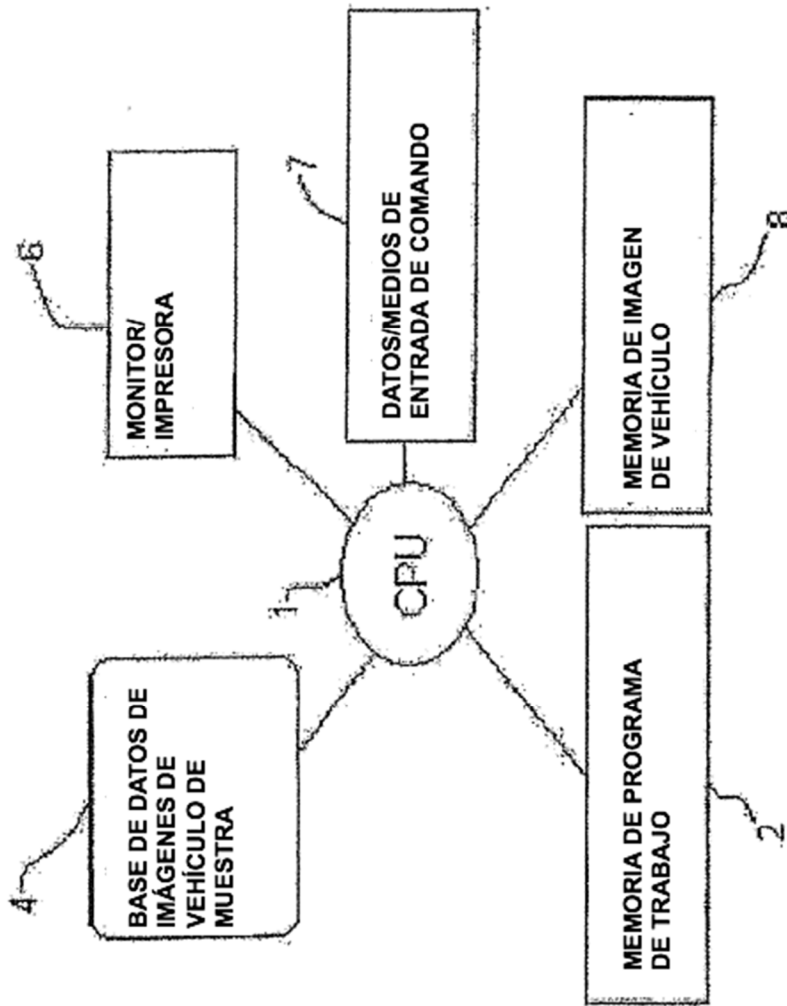


Fig.1

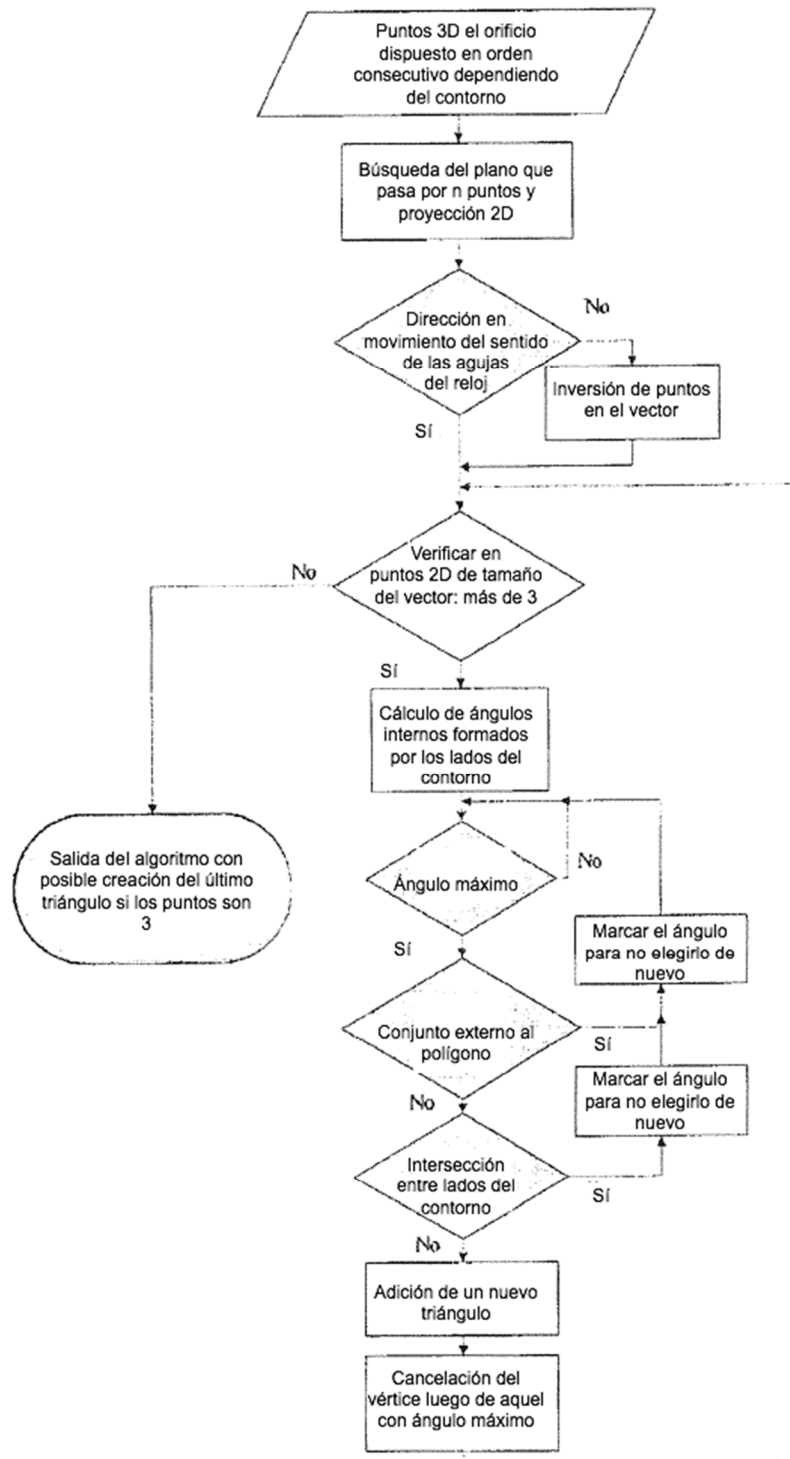


FIG. 2

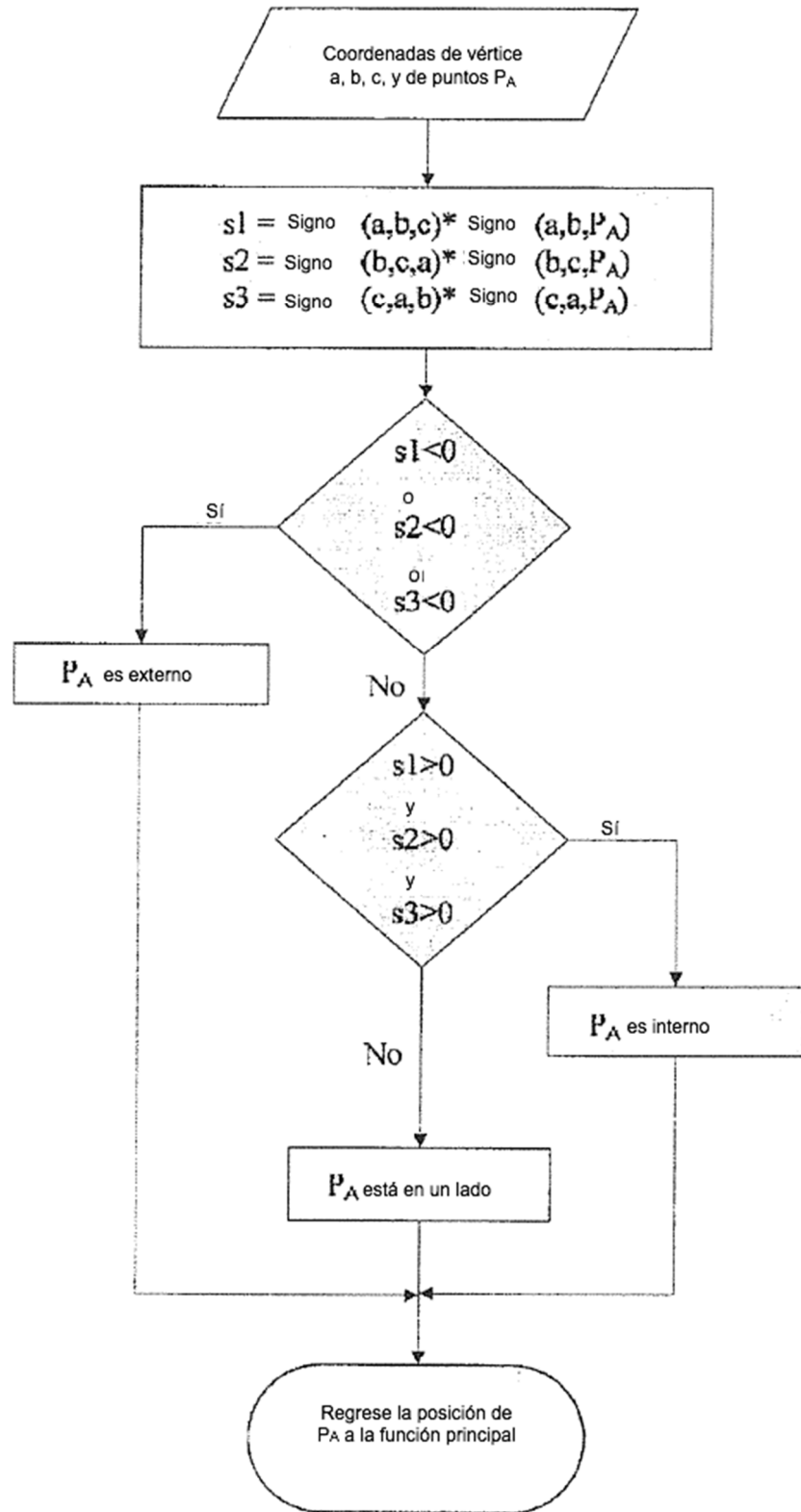


FIG. 3