



VNiVERSiDAD
DSALAMANCA

CAMPUS DE EXCELENCIA INTERNACIONAL

TRABAJO FIN DE MÁSTER

PROCESAMIENTO AUTOMÁTICO DE NUBES DE PUNTOS MLS PARA LA EXTRACCIÓN DE CARACTERÍSTICAS DE CARRETERAS

MÁSTER UNIVERSIDARIO EN GEOTECNOLOGÍAS
CARTOGRÁFICAS EN INGENIERÍA Y ARQUITECTURA

Autor: Jorge Mauricio Salas Sáenz

Tutores: Diego González Aguilera

José Antonio Martín Jiménez

SEPTIEMBRE DE 2021



Máster en Geotecnologías Cartográficas en Ingeniería y Arquitectura
Procesamiento automático de nubes de puntos MLS para la extracción de
características de carreteras





Agradecimientos

A todo el equipo de Grupo Acre y en especial a mis compañeros de departamento Eduardo, Álvaro y Oscar, por todo lo aprendido y por darme la oportunidad de trabajar con estos datos.

A mis padres por todo su apoyo y esfuerzo.

A Clara, Natalia y Sergio, por su ayuda en la realización del máster.

A mis tutores José Antonio Martín Jiménez y Diego González Aguilera por su apoyo, esfuerzo y dedicación en dirigirme con este proyecto.



Máster en Geotecnologías Cartográficas en Ingeniería y Arquitectura
Procesamiento automático de nubes de puntos MLS para la extracción de
características de carreteras





Resumen

Este trabajo fin de máster tiene como finalidad conocer más a fondo las nubes de puntos, sobre todo las de carreteras.

Para esto se va a contar el procesamiento para obtener nubes de puntos con el equipo Leica Pegasus Two, explicando el procedimiento para calcular la trayectoria y extraer la nube de puntos.

A partir de esto, se va a desarrollar un algoritmo utilizando Point Cloud Library (PCL), basado en el lenguaje C++ para extraer elementos de formas automática y de esta manera simplificar el arduo proceso de digitalización sobre nubes de puntos.

Finalmente se compararán y complementarán estos resultados obtenidos con un caso real práctico de un trabajo realizado en Colombia.



Abstract

The purpose of this study is to learn more about the point cloud, especially those for roads.

For this, the processing to obtain the point of clouds with the Leica Pegasus Two equipment will be counted, explaining the procedure to calculate the trajectory and extract the point cloud.

From this, an algorithm will be developed using Point Cloud Library (PCL), based on the C++ language to extract elements automatically and in this way simplify the arduous process of digitization on point cloud.

Finally, these results obtained will be purchased and complemented with a real practical case of a work carried out in Colombia.



Máster en Geotecnologías Cartográficas en Ingeniería y Arquitectura
Procesamiento automático de nubes de puntos MLS para la extracción de
características de carreteras





Índice

Agradecimientos.....	2
Resumen.....	4
Abstract.....	5
índice de imágenes.....	8
1. Introducción.....	10
2. Estado del arte.....	11
3. Objetivos.....	13
3.1 General.....	13
3.2 Específicos.....	13
4. Captura de datos con MLS.....	13
4.1 Localización.....	13
4.2 Equipo utilizado y proceso de captura de datos.....	14
4.2.1 Partes del equipo.....	14
4.2.2 Procesado de datos.....	15
5. Metodología.....	22
5.1 Extracción de elementos.....	22
5.1.1 Filtros utilizados.....	22
5.1.2 Detección de elementos.....	26
6. Resultados.....	27
7. Comparación de resultados.....	38
8. Conclusiones.....	43
9. Desarrollo futuro y mejoras.....	44
10. Referencias.....	45
11. Bibliografía.....	48



Índice de imágenes

Figura 1. GPS-Van.....	11
Figura 2. Localización de la zona de toma de datos.	13
Figura 3. Leica Pegasus Two.	14
Figura 4. Distancia a antena fija.	15
Figura 5. Procesado de trayectoria MLS.	17
Figura 6. Parámetros GNSS del procesado de la trayectoria MLS.	18
Figura 7. Diferencias entre la ida y la vuelta en los ejes XYZ.	19
Figura 8. Desviaciones estándar en los ejes XYZ.	20
Figura 9. Procesado en Leica Pegasus Manager.	21
Figura 10. Filtro "Statistical Outlier Removal".....	23
Figura 11. Filtro "VoxelGrid".	23
Figura 12. "Progressive Morphological Filter".	24
Figura 13. Filtro "Pass Through".	25
Figura 14. Filtro "Concave Hull",.....	25
Figura 15. Traducción Cloud Compare.	27
Figura 16. Nube de puntos de prueba.....	28
Figura 17. Nube de puntos reducida.	29
Figura 18. Nubes de puntos "Terreno" y "No terreno".	29
Figura 19. Segmentación línea blanca.	30
Figura 20. Segmentación línea blanca sin ruido.	30
Figura 21. Línea blanca tras aplicarle el filtro "VoxelGrid".....	31
Figura 22. Traducción de la nube de puntos *.PCD.	32
Figura 23. Segmentación asfalto.	32
Figura 24. Segmentación asfalto sin ruido.	33
Figura 25. Contorno de asfalto.	33
Figura 26. Formato de fichero *.PCD.	34
Figura 27. Importar puntos MDT.....	34
Figura 28. Puntos importados en Autocad/MDT.....	35
Figura 29. Polilínea del contorno de asfalto.	35
Figura 30. Puntos reducidos para obtener MDT.....	36
Figura 31. Crear superficie en Autocad/MDT.	36
Figura 32. Superficie - MDT.	37
Figura 33. Curvado.	37
Figura 34. Comparación de nube de puntos con delineación.	38
Figura 35. Comparación de bordes de asfalto.	39
Figura 36. Delineación, trabajo 2020.	39
Figura 37. Curvado TFM.	40



<i>Figura 38. Comparación de curvados.....</i>	<i>40</i>
<i>Figura 39. Comparación curvados, detalle.</i>	<i>41</i>
<i>Figura 40. Comparación de superficies.</i>	<i>42</i>

Índice de tablas

<i>Tabla 1. Diferencias de puntos entre ambas superficies.....</i>	<i>42.</i>
--------------------------------------------------------------------	------------



1. Introducción

Durante gran parte de mi vida laboral he trabajado con nubes de puntos, actualmente me dedico a trabajos de captura masiva de datos con distintos sensores como son el láser escáner, Mobile LiDAR System (MLS) y vehículos aéreos no tripulados (UAV).

Una de las grandes dificultades de trabajar con nubes de puntos es la gran cantidad de información que se tiene, cuando muchas veces no es necesaria toda. Es por ello, que existen softwares que ayudan a simplificar muchas tareas, extrayendo algunos elementos o clasificando las nubes.

Uno de los casos más comunes a los que me enfrento es a la digitalización de carreteras a partir de datos obtenidos mediante MLS, en estos casos el entregable a los clientes suele un archivo de CAD con la línea blanca y límites de asfalto y un archivo con un modelo digital del terreno (MDT) del suelo.

Debido al gran trabajo que supone delinear sobre una nube de puntos, en el presente trabajo fin de máster, en adelante TFM, he desarrollado una herramienta que me ayude a simplificar este proceso. Esta herramienta se basa en algoritmos adaptados de Point Cloud Library (PCL), que es una biblioteca de código abierto basada en el lenguaje C++.

Finalmente se compararán los resultados obtenidos con esta herramienta con los obtenidos en un trabajo real en el que había que delinear distintos elementos de una carretera.

2. Estado del arte

Los sistemas Light Detection and Ranging (LiDAR) se han utilizado ampliamente en topografía desde principios de los años 70. Al ser una tecnología que permite la medición de distancia utilizando como emisor un láser escáner, midiendo por tiempo de vuelo o cambio de fase. (González-Jorge et al. (2013))

El primer sistema MLS para la recolección de datos cartográficos fue el GPS-Van, este se desarrolló por el FHWA (Federal Highway Administration, o la administración Federal de carreteras, de Estados Unidos). Este sistema usaba un receptor GPS, dos cámaras digitales, dos cámaras de video a color y una unidad inercial con dos giróscopos y un codificador. El sistema consigue una precisión en localización de aproximadamente 0.1 metros para objetos con una distancia inferior a 20 metros del vehículo (Novak & Feng (1993), citado en Porrás Díaz et al. (2018)).



Figura 1. GPS-Van

Fuente:

<http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=05796394CD86186067D1F192C306B23F?doi=10.1.1.471.1414&rep=rep1&type=pdf>

Posteriormente, en los noventa, se desarrolla el sistema VISAT, compuesto por un sistema GPS, una cámara de video S-VHS, una unidad de control de imagen, ocho cámaras CCD integradas y un sistema inercial. Este sistema capta imágenes en blanco y negro, georreferenciadas a 60km/h, obteniendo una precisión en el posicionamiento de 0.3 metros de posición absoluta y 0.1 metros de posición relativa para puntos que están a un radio de 35 metros. (Tao (2000), Tao et al. (2001), citado en Porrás Díaz et al. (2018)).



Durante las últimas décadas, los avances en Sistemas Globales de Navegación por Satélite (GNSS, Global Navigation Satellite System), LiDAR (Light Detection and Ranging), la adquisición de información radiométrica y la hibridación entre sensores, han revolucionado la adquisición masiva de datos, así como las técnicas de inspección y seguimiento. Un claro ejemplo de esto es el Mobile LiDAR System (MLS) (Bitenc et al. (2011), Gonzalez-Jorge et al. (2013), Mc Elhinney et al. (2010), Puente et al., (2013), citado en Martín-Jiménez et al., 2018).

Como dijo Varela-Gonzalez et al. (2014), los sistemas MLS permiten adquirir información geométrica y radiométrica del entorno 3D con alta resolución, precisión y productividad. Sin embargo, a diferencia de otros sistemas topográficos tradicionales, estos registran información de forma masiva. Esto exige ordenadores potentes y el desarrollo de herramientas de software específicas para el correcto procesamiento de los datos, ya que de lo contrario a productividad ganada en la fase de recopilación de datos puede perderse durante el procesamiento.

Con ello surge la necesidad de clasificar y segmentar nubes de puntos. Estos son algunos ejemplos que se han desarrollado:

- Jaakkola et al. (2008), trabajó en la detección de bordillos utilizando gradientes.
- Foucher et al. (2011), se centraron en el desarrollo de la detección de marcas de viales urbanos como son los pasos de cebra, flechas y líneas blancas.
- Varela-Gonzalez et al. (2014), desarrollaron un filtro para eliminar automáticamente vehículos de nubes de puntos de carreteras.
- Holgado-Barco et al. (2015), desarrollaron un método para extraer el eje de la carretera y los parámetros horizontales como son rectas, arcos circulares y clotoides para estudiar si estos cumplen con la norma de carreteras.

Si bien existen muchos métodos para la extracción y segmentación de elementos de nubes de puntos, los algoritmos utilizados en este TFM están más enfocados a un sistema en concreto, Leica Pegasus Two, y a optimizar un tipo de trabajo en concreto, como es la delineación de la geometría de carreteras.

3. Objetivos

3.1 General

Desarrollar un algoritmo basado en PCL que permita extraer y segmentar elementos de forma automática para facilitar el manejo de nubes de puntos.

Esta aplicación se centrará en la extracción de elementos de la carretera como son línea blanca, borde de asfalto y un modelo digital del terreno con un curvado.

3.2 Específicos

- Aprender más acerca de PCL y de filtros de nubes de puntos.
- Mayor manejo de softwares de nubes de puntos.

4. Captura de datos con MLS

A continuación, se va a contar como fue la captura de datos con MLS y el procesamiento de los datos hasta obtener la nube de puntos.

4.1 Localización

El área de estudio de este proyecto se localizó en El Boquerón, al suroeste de Bogotá, Colombia. Estos datos pertenecen a un trabajo real que fue realizado en el año 2020, para este trabajo se midieron 3440 metros de carretera convencional con MLS.

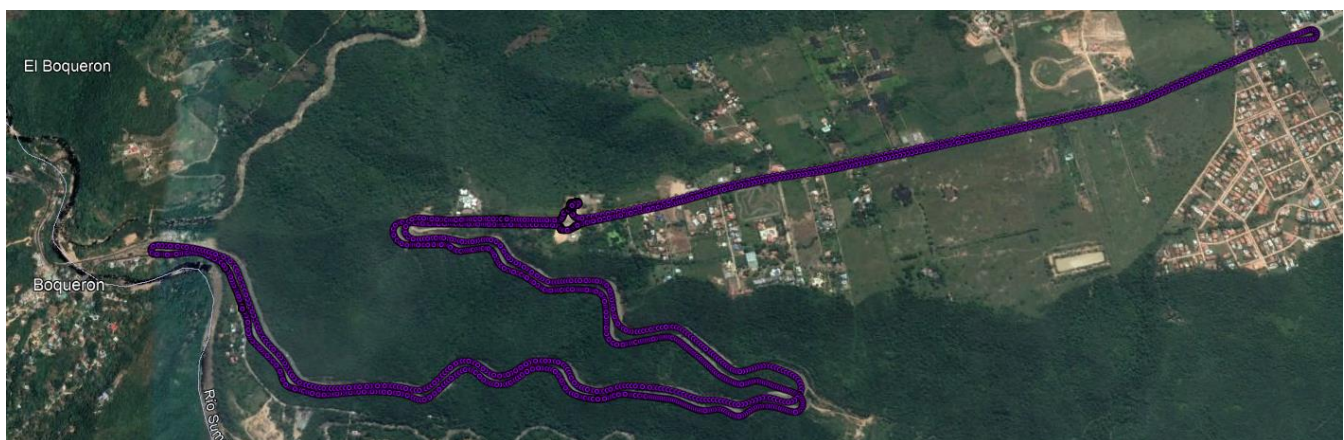


Figura 2. Localización de la zona de toma de datos.

4.2 Equipo utilizado y proceso de captura de datos

El equipo utilizado fue el Leica Pegasus Two, este equipo va montado sobre un vehículo siendo capaz de registrar 1.000.000 de puntos por segundo, con un alcance máximo de 120 metros a cada lado de vehículo y circulando a una velocidad máxima de 90 km/h.



Figura 3. Leica Pegasus Two.

4.2.1 Partes del equipo

Sistema GNSS: Se trabaja con GPS diferencial, suelen ir una o dos antenas GNSS sobre el vehículo, en caso de contar con dos antenas se podría saber la attitude, que es la orientación del vehículo, ya que entre las dos antenas se forma un vector que indicaría la dirección.

Sistema Inercial: IMU (Unidad de Medida Inercial), compuesta por 3 giroscopios y 3 acelerómetros, en algunos casos lleva un magnetómetro.

Los acelerómetros sirven para conocer los desplazamientos, tres acelerómetros uno para eje, XYZ, y tres giróscopos para conocer las rotaciones en los tres ejes.

Los sistemas inerciales son un complemento de los sistemas GNSS, se debe limitar su uso ya que introducen error de navegación por estima. Sin embargo, son necesarios para el cálculo preciso de la trayectoria en zonas donde el receptor GNSS no tiene cobertura.

Odómetro: es un sistema opcional. Este sensor mide el número de vueltas de una rueda, conociendo el diámetro de la rueda en el que va montado se puede calcular el desplazamiento total.

Láser escáner: el láser escáner que va montado sobre en estas plataformas es, generalmente, 2D, como es el caso de Leica Pegasus Two, que cuenta con un láser escáner ZF9012.

Cámaras: Este equipo cuenta con 7 cámaras con la opción, mientras se procesan los datos, de obtener una imagen esférica 360°.

4.2.2 Procesado de datos

Después de la toma de datos en campo, se procede al procesamiento de datos. Lo primero de todo es calcular la trayectoria, la cual se calcula en función del tiempo GNSS.

La distancia máxima entre el máster (Antena Fija) y el rover (antena del MMS en movimiento), puede variar en función del fabricante, en el caso del Leica Pegasus Two, se recomienda que no sea superior a 30km.

En este caso, como se puede observar en la siguiente imagen es de un caso de MMS, al Sur de Bogotá, en este caso sólo hay un máster (PUNTOPEGA), ya que la zona medida no supera los 5km y se encuentra dentro de la zona medida.

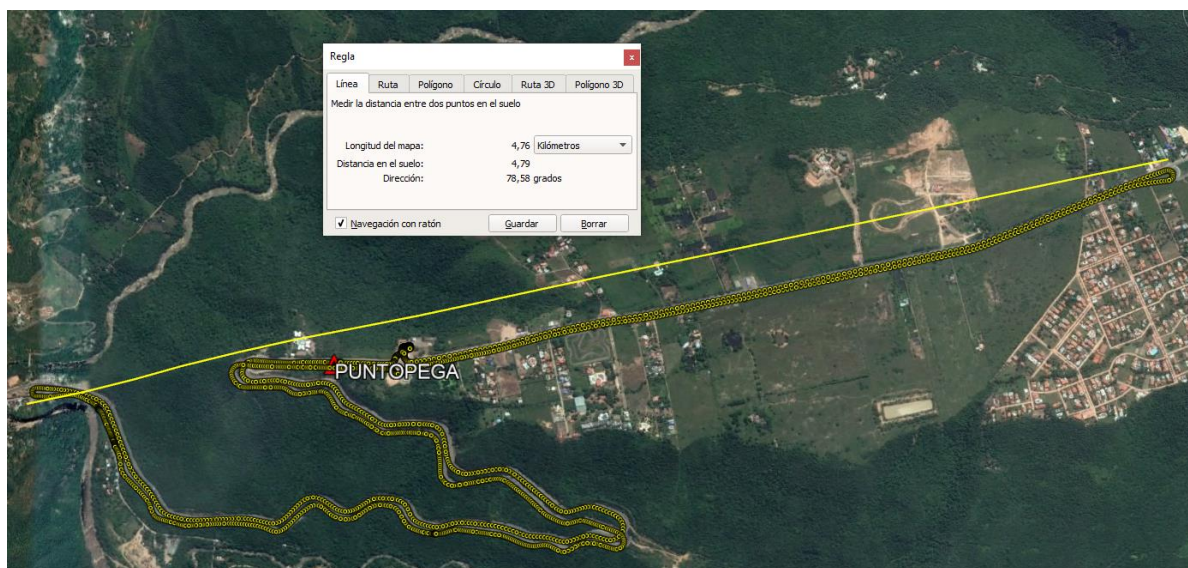


Figura 4. Distancia a antena fija.



Lo primero para calcular la trayectoria es tener los datos de observación del máster, con estos datos y los datos de la antena GNSS del Mobile Mapping (rover), los datos del inercial y los del odómetro, en caso de que se utilizara, se calculará la trayectoria con el software NovAtel Inertial Explorer.

La trayectoria calculada con este software servirá para posteriormente extraer las imágenes y nubes de puntos en función del tiempo GNSS de la trayectoria ajustada.

El método de procesado es "Diferencial GNSS", este método se utiliza para realizar mediciones entre dos o más receptores GNSS y así calcular la corrección diferencial entre las medidas proporcionando una mayor precisión a las coordenadas finales, ya que estos datos pasan por un Filtro de Kalman, que estima de todas las medidas cuales son las más y menos precisas, obteniendo como resultado una SBET (Smooth Best Estimate Trajectory), que se traduciría como mejor estimación de la trayectoria.

En este software también permite elegir la dirección en la que se quieren procesar los datos, al haber hecho un dinámico y estático al principio y final se pueden calcular los datos de la trayectoria en ambos sentidos; lo ideal es calcularlo en ambos sentidos y con "Multi-Pass", esto significa que la calculará en ambos sentidos y ajustará la trayectoria entre sí, de esta manera se obtiene una comprobación sobre la calidad de la trayectoria.



Process Tightly Coupled

Processing Method
 Differential GNSS Precise Point Positioning (PPP)

Processing Direction
 Both Forward Reverse Multi-pass

Processing Settings
Profile: Use Current Project Settings Filter Profiles
Datum: WGS84

IMU Installation
 Read rotations and lever arms from IMR file

Lever Arm Offset (IMU to GNSS antenna)
X: -0.088 m Y: -0.068 m Z: 0.473 m Z to ARP Z to Phase Centre

Body to IMU Rotation (order: Z, X, Y)
X: 0.000 deg Y: 0.000 deg Z: 180.000 deg

GNSS Heading Offset
0.000 deg

Processing Information
Description: TC (2) User: Unknown

Figura 5. Procesado de trayectoria MLS.

Los siguientes parámetros que se pueden editar son el intervalo de segundos que se quiere utilizar para procesar la trayectoria y la máscara de elevación.

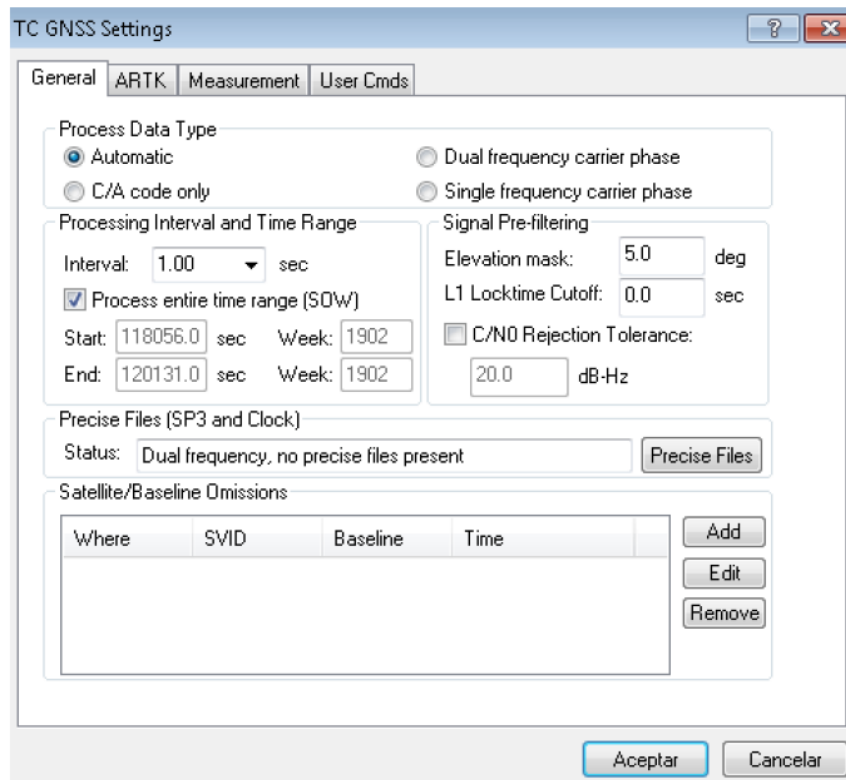


Figura 6. Parámetros GNSS del procesado de la trayectoria MLS.

Una vez terminada de procesar la trayectoria se analizan los resultados con las gráficas obtenidas del Inertial Explorer.

La primera gráfica es la siguiente:

Figure 3: Tfg [Smoothed TC Combined] - Forward/Reverse or Combined Separation Plot

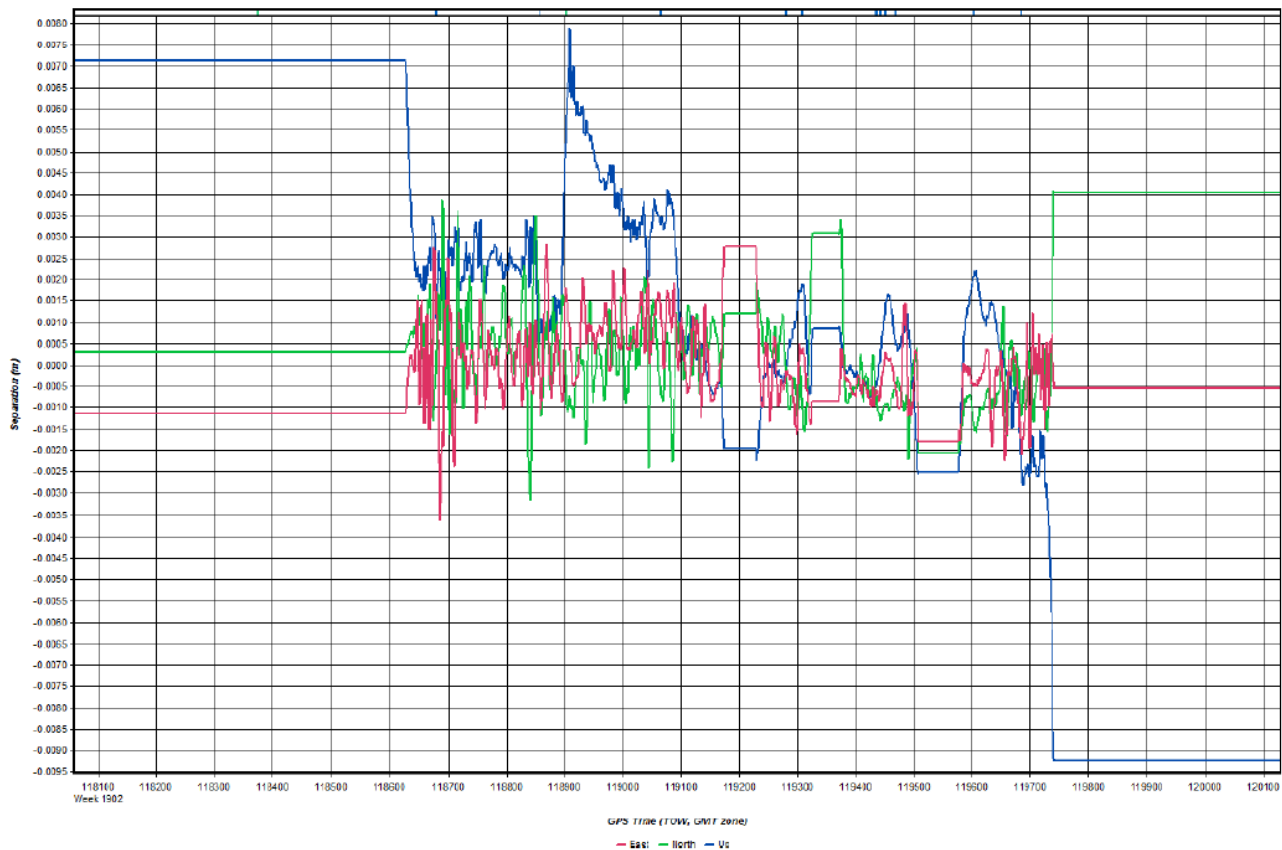


Figura 7. Diferencias entre la ida y la vuelta en los ejes XYZ.

Esta gráfica muestra las diferencias entre el cálculo de la trayectoria de la ida y la vuelta, siendo el eje "X" tiempo GNSS y el Eje "Y" las diferencias en metros.

Las líneas de color Rojo se corresponden con las diferencias en "Este"; las de color verde en "Norte" y las azules en "Altura".

Las zonas donde la gráfica muestra una línea recta, al principio y al final, se corresponde con los dos estáticos que se realizaron, uno antes de la toma de datos y otro al finalizar la toma de datos.

La diferencia más grande que se encuentra durante la toma de datos es de 0.0078 m aproximadamente y se produce en cota, en planimetría los errores más grandes son de -0.0036 m en "Este" y 0.0031 m en "Norte".

La siguiente gráfica muestra las desviaciones en los distintos ejes, como se puede observar estas están en torno a 0.005 m en cota, y son de entre 0.002 y 0.004 metros en planimetría.

Figure 2: Tfg [Smoothed TC Combined] - Estimated Position Accuracy Plot

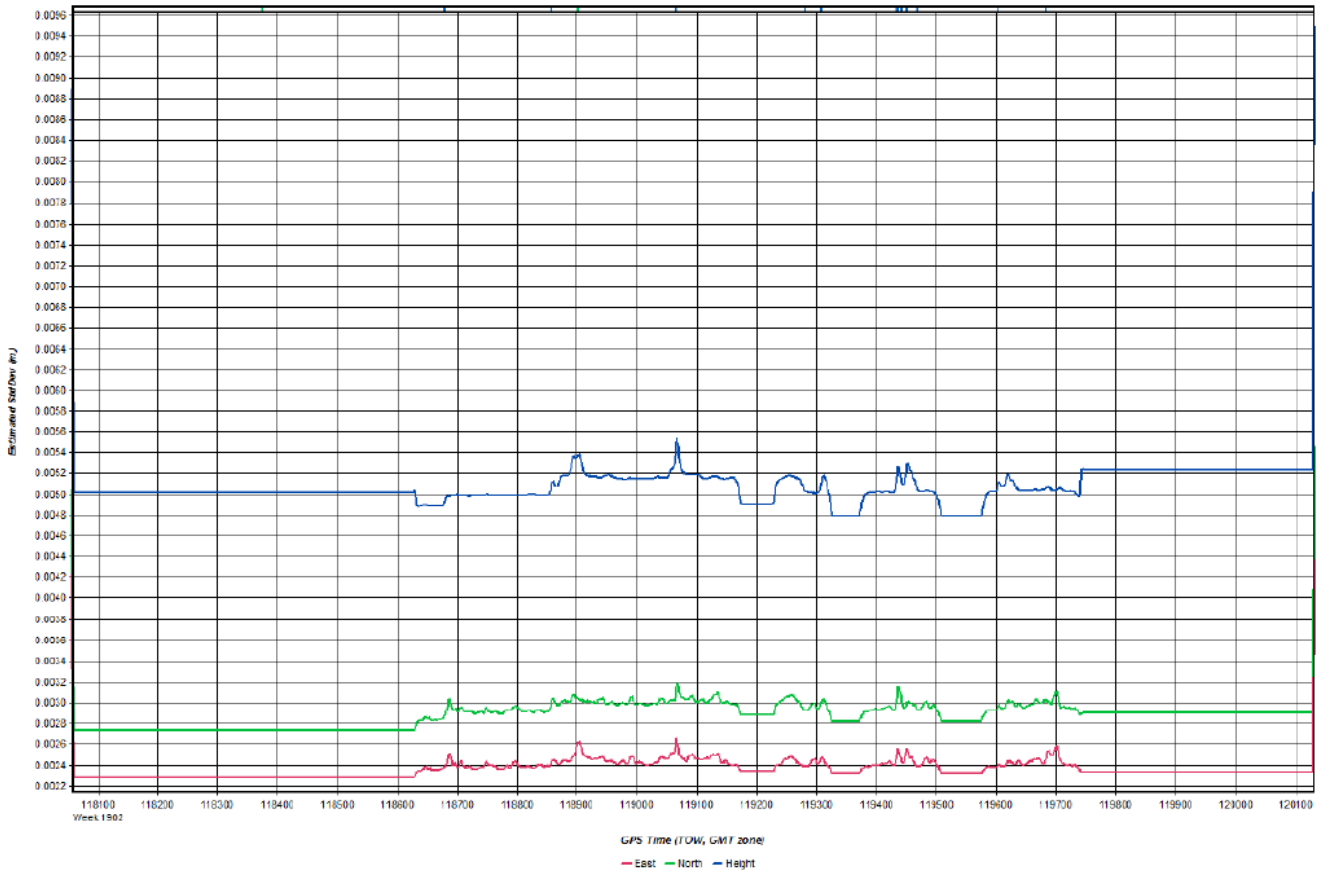


Figura 8. Desviaciones estándar en los ejes XYZ.

Cuando se han revisado las gráficas y se observa que el resultado es bueno, se procede a exportar la trayectoria para utilizarla en el siguiente paso, que consiste en la extracción de las imágenes y nubes de puntos en función de esta trayectoria calculada.

En el caso de los datos del Leica Pegasus Two, el software utilizado para la extracción de imágenes y nubes de puntos es el Leica Pegasus Manager.

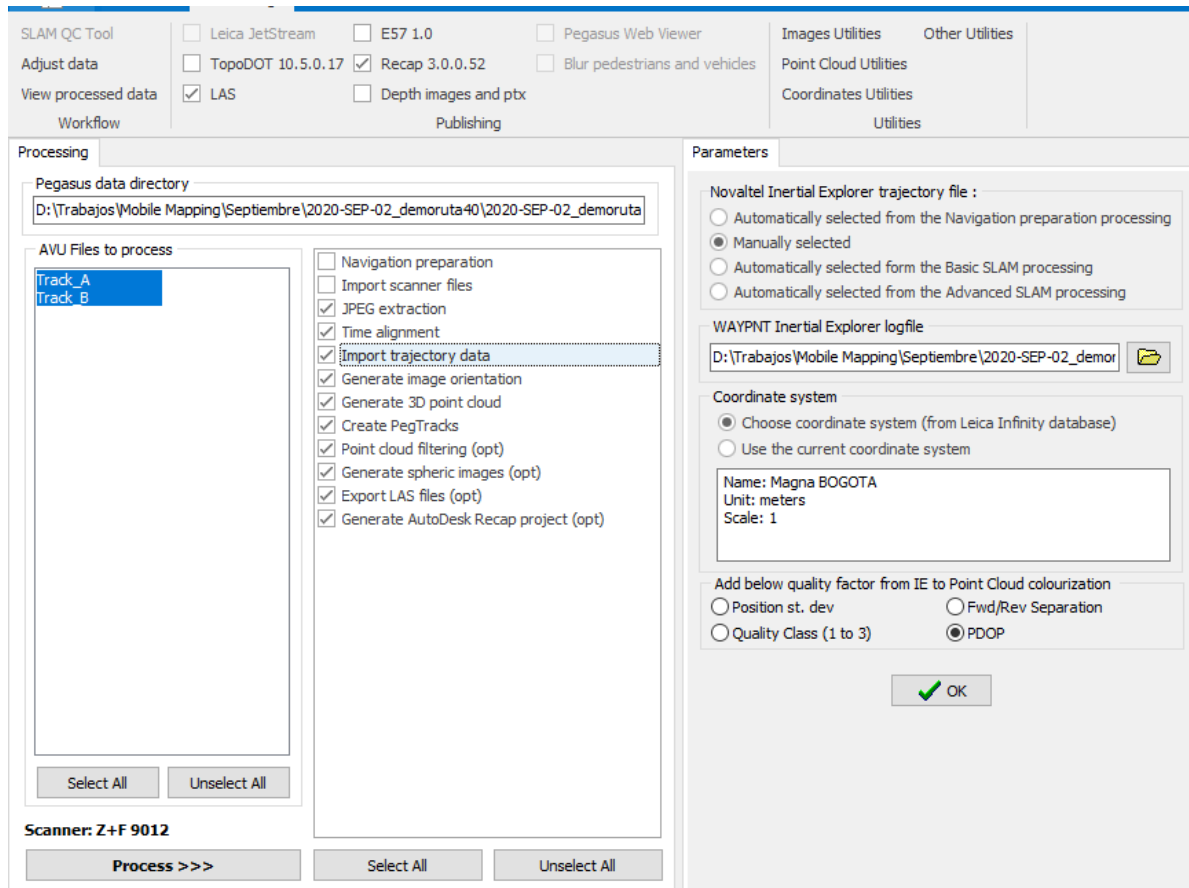


Figura 9. Procesado en Leica Pegasus Manager.

Seleccionada la trayectoria y el sistema de coordenadas, ya se procedería a extraer las imágenes y nubes de puntos, con el software Leica Pegasus Manager.

Finalmente se obtiene la nube de puntos, en este caso se obtuvieron dos tracks. Un track es el bloque de nube de puntos desde que se empieza a registrar datos, activando las cámaras y el láser escáner, hasta que se detiene.

Dentro de un mismo proyecto puede haber varios tracks. En este proyecto hay dos tracks, uno de ida y otro de vuelta, si bien por el ancho de la carretera era más que suficiente con uno, se grabaron dos para luego elegir el track en que hubiera menos coches haciendo sombra.



5. Metodología

En este apartado se va a contar las herramientas y filtros utilizados para la extracción de elementos.

Para realizar estas tareas se ha utilizado Point Cloud Library (PCL) sobre la que se adaptaron diferentes algoritmos que han permitido obtener los resultados esperados.

Todo el código utilizado en este TFM se subirá y compartirá para que se pueda utilizar como código abierto y libre.

5.1 Extracción de elementos

La primera parte del proceso de trabajo se consiste en preparar la nube de puntos, mediante una serie de filtros, para segmentarla en dos clases: 'terreno' y 'no terreno'.

5.1.1 Filtros utilizados

a) Filtro de reducción de ruido

Es un filtro llamado "*Statistical Outlier Removal*" que sirve para eliminar el ruido de las nubes de puntos.

Este filtro actúa detectando valores atípicos dispersos, mediante el cálculo de distancias de un punto al de sus vecinos. Se calcula una media y desviación estándar de los puntos vecinos y los que estén fuera estos umbrales, se consideran puntos de ruido.

En verde, la nube limpia tras aplicarle el filtro y en rojo el ruido detectado por el filtro.

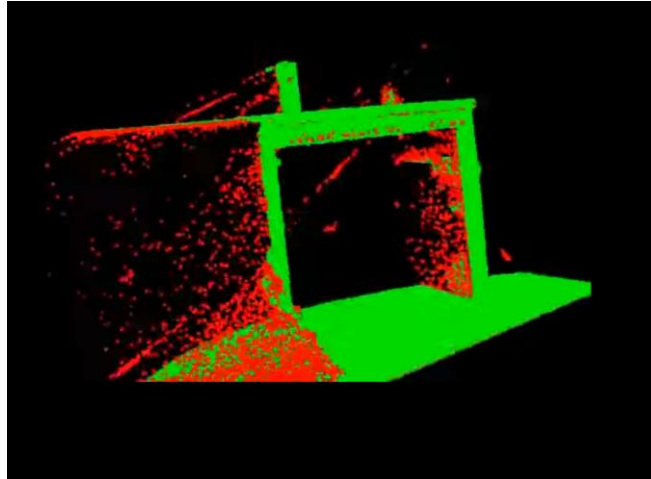


Figura 10. Filtro "Statistical Outlier Removal".

Fuente: https://pointclouds.org/documentation/tutorials/statistical_outlier.html

b) Filtro de reducción de número de puntos

Este filtro llamado "Voxel Grid", es una reducción de la nube de puntos siguiendo la distribución de una cuadrícula 3D.

A la izquierda la nube sin reducir y a la derecha la nube reducida.

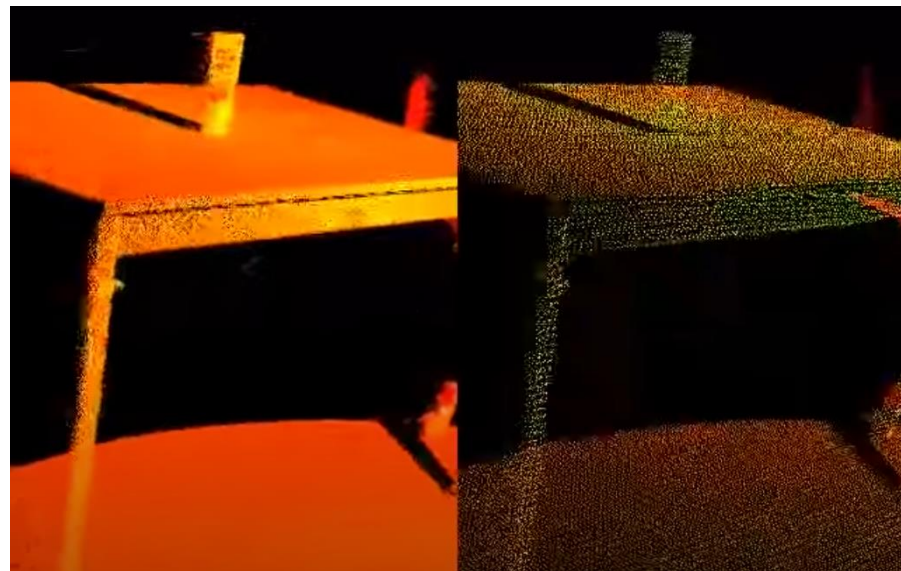


Figura 11. Filtro "VoxelGrid".

Fuente: https://pointclouds.org/documentation/tutorials/voxel_grid.html

c) Filtro de extracción de suelo

Este filtro llamado "*Progressive Morphological Filter*", sirve para segmentar la nube de puntos en dos llamadas "terreno" y "no terreno".

En escala de grises se encuentra la nube "terreno" y en intensidad de escala de colores la nube "no terreno".

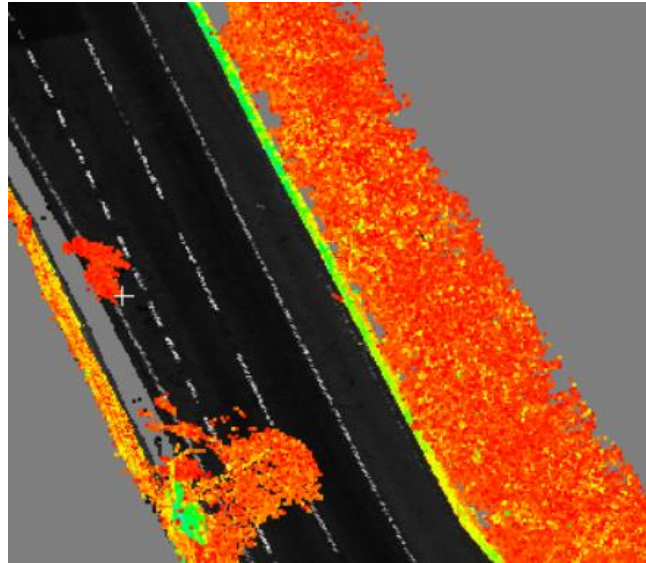


Figura 12. "*Progressive Morphological Filter*".

d) Filtro de paso

Conocido como "*Pass Through*", este filtro segmenta la nube en función de un campo determinado y los valores de corte pueden estar dentro o fuera de un rango determinado.

En tonos rojizos la nube completa y en blanco la segmentación de la línea blanca, en función de sus valores de intensidad.

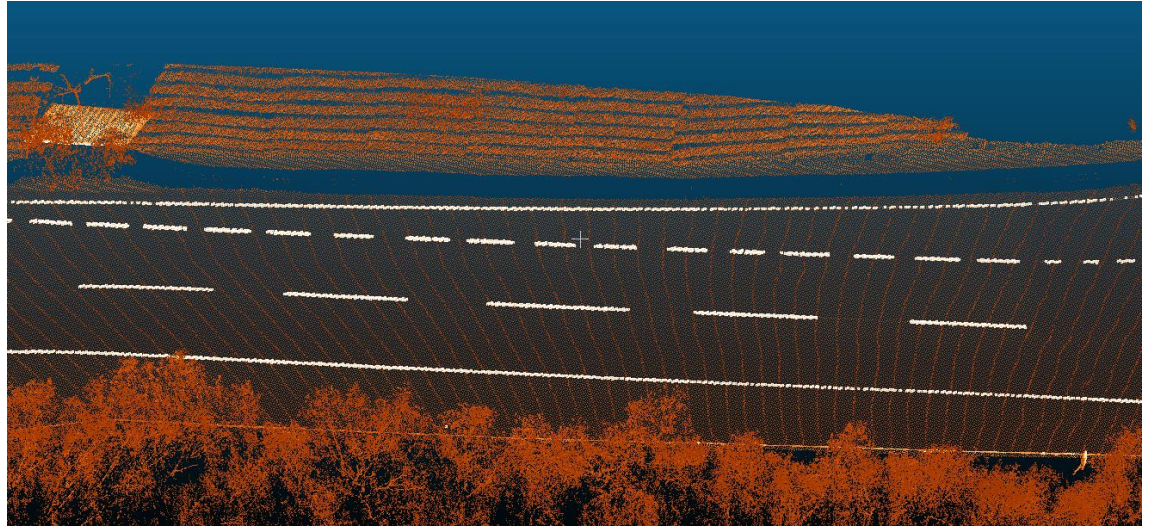


Figura 13. Filtro "Pass Through".

- e) Filtro de extracción de contorno de nubes
Llamado "Concave Hull", es filtro extrae los puntos del
contorno cóncavo de la nube de puntos.

En intensidad de escala de grises la nube de puntos y en
rojo, los puntos de contorno extraídos con el filtro.

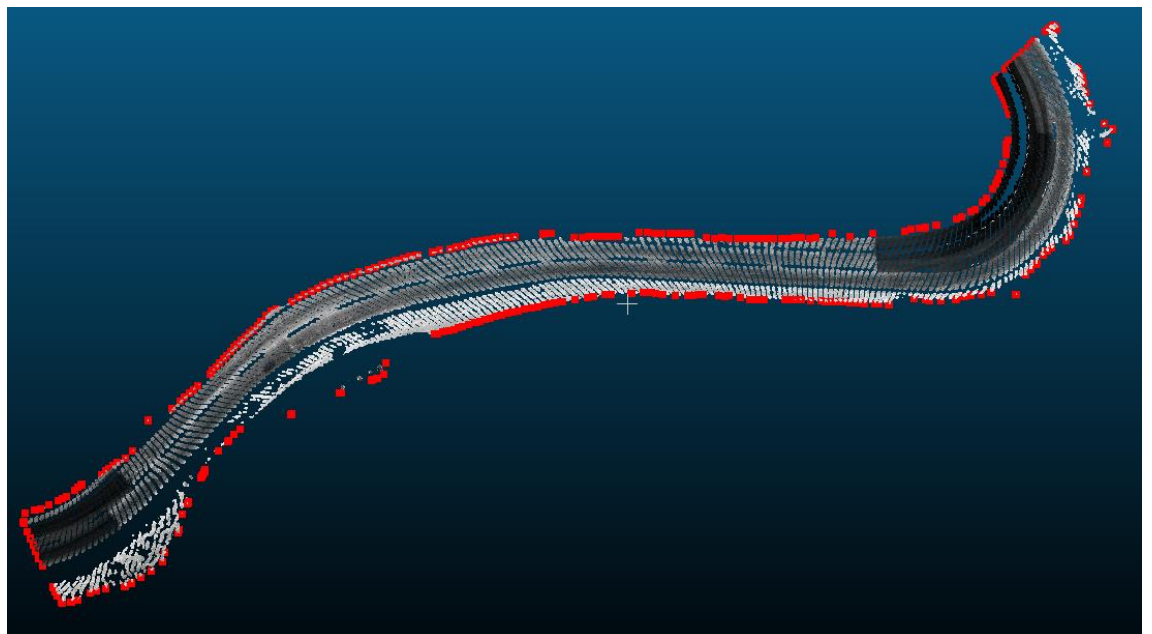


Figura 14. Filtro "Concave Hull".



5.1.2 Detección de elementos

a) Extracción de línea blanca

Para la extracción de la línea blanca, se han utilizado tres filtros, el filtro de paso "*Pass Through*", el filtro de reducción de ruido "*Statistical Outlier Removal*" y el filtro de reducción de densidad "*Voxel Grid*". Para aplicar el filtro de paso se han analizado los valores de intensidad de la nube de puntos, determinando que los valores de intensidad de la línea blanca son de 30000 a 60000.

Tras aplicar este filtro, se aplica el filtro de eliminación de ruido, con los valores 0.3 como multiplicador de la desviación estándar y 5 como número de puntos vecinos.

Para el filtro "*VoxelGrid*" se ha utilizado una distancia de un punto cada metro.

b) Extracción de contorno de aglomerado

Para obtener esta segmentación se han utilizado tres filtros diferentes, primero se ha utilizado el filtro de paso "*Pass Through*" con el intervalo de valores de intensidad de 1000 a 6000, tras analizar que los valores de intensidad del aglomerado se encuentran en este rango de valores.

Posteriormente se ha utilizado el filtro de eliminación de ruido "*Statistical Outlier Removal*" para eliminar puntos aislados fuera de la zona de interés.

Finalmente, con el filtro de extracción de contorno de nubes "*Concave Hull*".

c) Modelo digital del terreno

Para el modelo digital del terreno se ha utilizado la nube de puntos "terreno" y se ha utilizado el filtro de reducción de número de puntos "*Voxel Grid*", para reducir el volumen de la nube, dejando un punto cada 0.75 metros. Si bien es una densidad de puntos bastante pobre, para realizar un MDT es más que suficiente.

6. Resultados

Para poder trabajar con las nubes de puntos en esta herramienta, es necesario que el formato de estas nubes de puntos sea *.PCD. Es por esto, que el primer paso es importar las nubes en el software *Cloud Compare*.

Al importar una nube de puntos a *Cloud Compare*, este software realiza una transformación, desplazando la nube para que las coordenadas XYZ de los puntos no seas números tan altos. El software no obliga a realizar esta transformación. Sin embargo, si no se realiza esta transformación, la nube de puntos pierde calidad.

Es por ello, que a esta nube si se le ha aplicado la transformación.

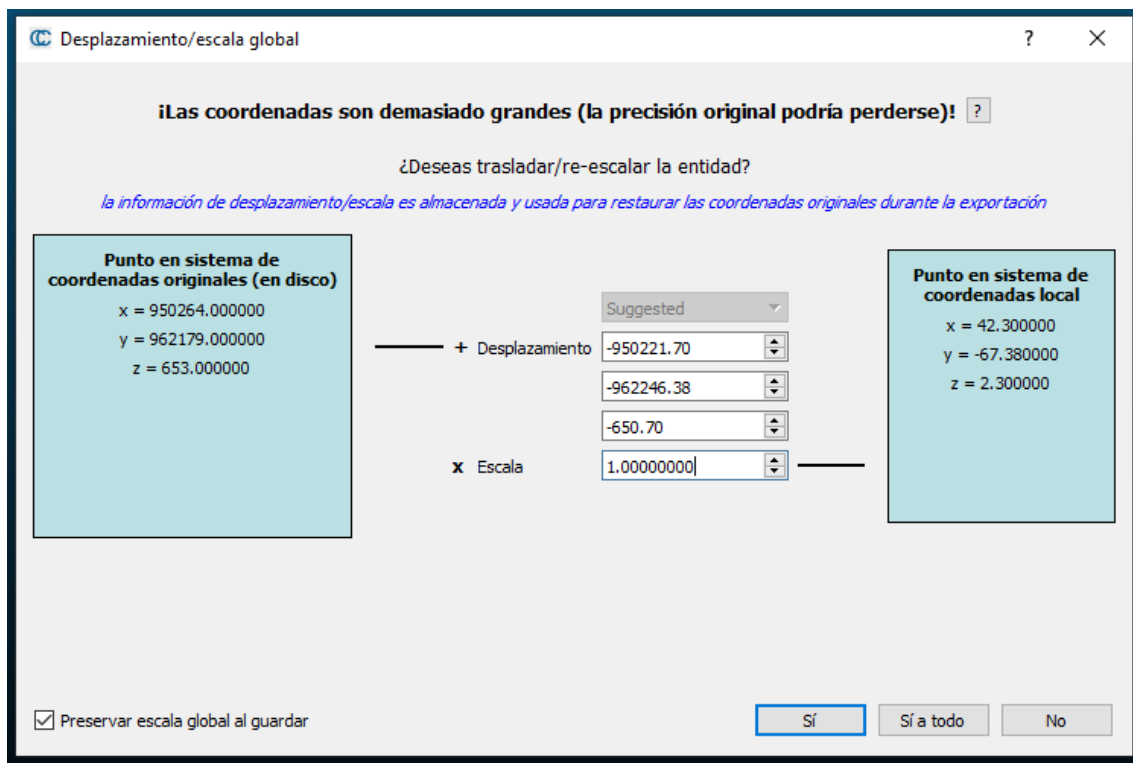


Figura 15. Traslación *Cloud Compare*.

Es importante conservar estos desplazamientos porque a los resultados se les aplicará una traslación para que la nube de puntos vuelva a su posición real.

Para observar el resultado de los filtros se ha utilizado una sobre una nube de puntos de aproximadamente 270 metros de carretera.

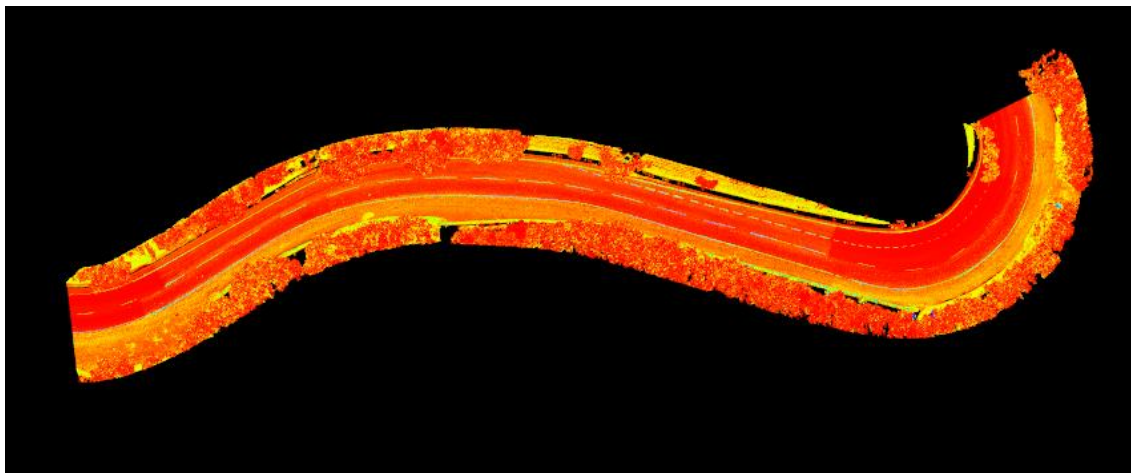


Figura 16. Nube de puntos de prueba.

El primer filtro que aplicado es el de reducción de número de puntos “Voxel Grid”, este se utiliza para la reducción de número de puntos, esta herramienta crea cuadrículas 3D del tamaño que se le indique. Una vez que se tiene esta caja (voxel) se reducen todos los puntos quedándose solo con el centroide.

En este caso, se han realizado varias pruebas y se ha utilizado una resolución de 0.15m, esto quiere decir que se utilizará un punto cada 0.15m.

Tras la utilización de este filtro el número de puntos se redujo de 15 251 530 puntos a 1 620 587 puntos, prácticamente el 10%, siendo aun así la densidad suficiente para poder realizar la extracción de elementos correctamente.

El tiempo de procesado de este filtro fue de aproximadamente 15 minutos.

La imagen de arriba es la nube reducida y la de abajo la original.

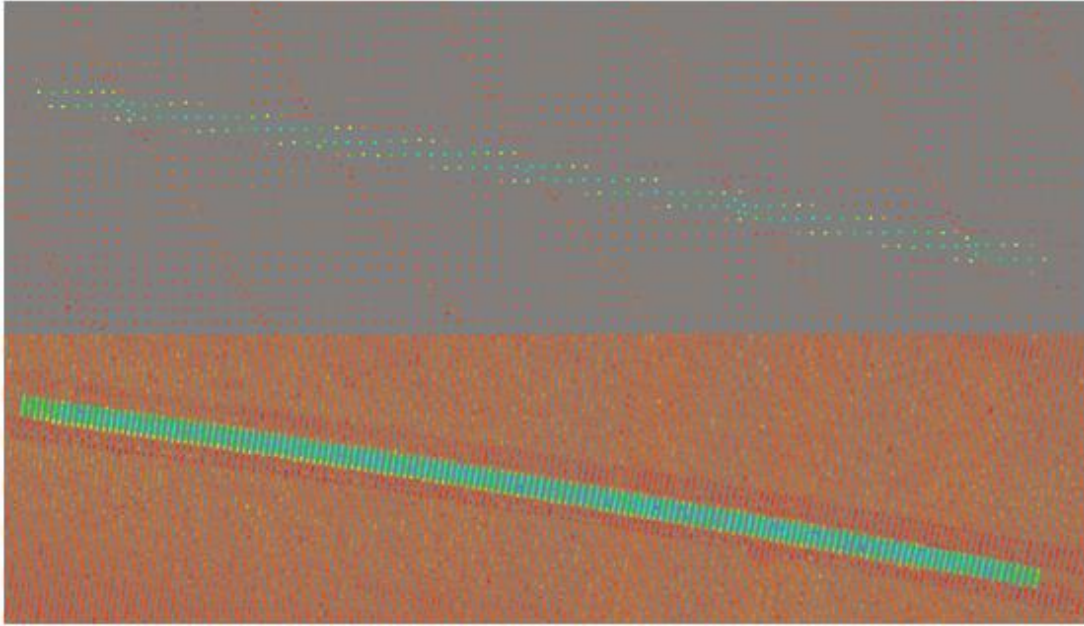


Figura 17. Nube de puntos reducida.

Posteriormente, con la nube ya reducida, se procede a aplicar el filtro “*Progressive Morphological Filter*” para obtener dos nubes de puntos: “terreno” y “no terreno”.

En la siguiente imagen se puede observar en colores de intensidad en escala de grises, la nube “terreno” y en colores de intensidad en escala verde-rojo la nube “no terreno”

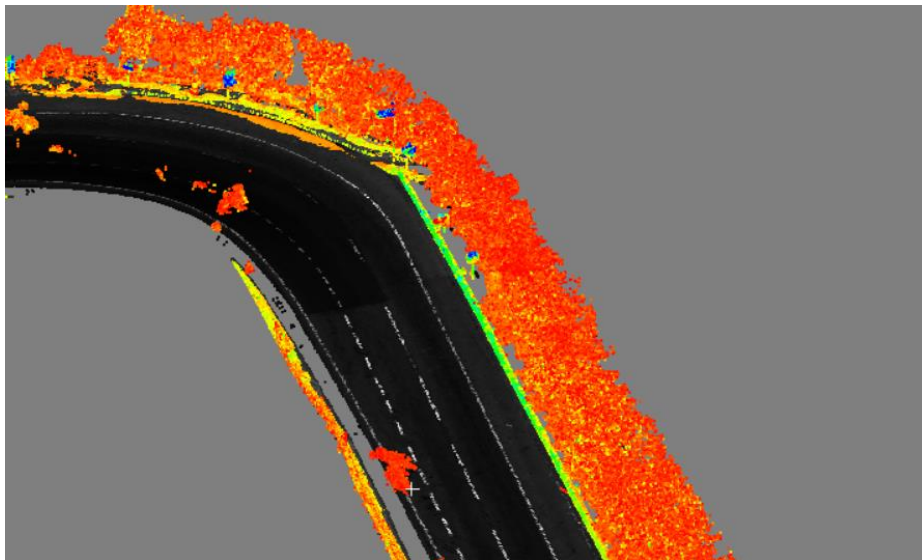


Figura 18. Nubes de puntos “Terreno” y “No terreno”.

La segmentación de la nube ha sido bastante correcta y el resultado es el esperado.

Para obtener este resultado, el tiempo de procesado fue de 2 horas y 45 minutos, siendo este el filtro que más tiempo tomaba.

Ahora, partiendo de la nube "Terreno" se puede extraer la línea blanca y el borde de aglomerado. El MDT ya se podría generar debido a que ya se tiene el terreno, habría que reducir la nube para que al crear el MDT no tener un exceso de triángulo (por ejemplo, dejar un punto cada metro).

Para extraer la línea blanca se ha utilizado un filtro de paso o "*Filtro Pass Through*", el cual consiste en segmentar la nube de puntos en función de un rango de valores, para estas nubes de puntos se vio que la línea blanca se encuentra entre los valores 30000 a 60000 de intensidad, por lo cual se eligió este intervalo.



Figura 19. Segmentación línea blanca.

Como se puede observar en la imagen, la segmentación ha sido bastante buena, sin embargo, quedan algunos puntos exteriores que no pertenecen a la línea blanca. Es por eso, que luego se le aplica el filtro de reducción de ruido "*Statistical Outlier Removal*", para eliminar los puntos aislados,

El resultado fue el siguiente, como se puede observar queda un resultado más depurado y limpio, sólo con la línea blanca.

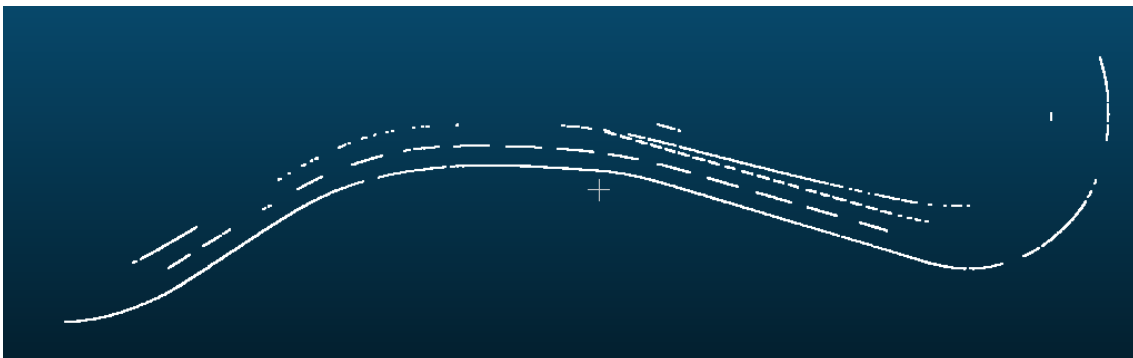


Figura 20. Segmentación línea blanca sin ruido.

Para aplicar el filtro de reducción de ruido se han aplicado los siguientes parámetros:

- El número de puntos vecinos a analizar es de 5, eliminando así los puntos aislados.
- Y el valor multiplicador de la desviación estándar en 0.3; esto significa, que todos los puntos que tengan una desviación estándar mayor que 0.3 por la desviación estándar media, se consideran valores atípicos y serán eliminados.

Finalmente, se utilizó el filtro de reducción de densidad de nube de puntos "Voxel Grid", para eliminar punto en paralelo dentro de línea blanca, y obtener un sólo punto dentro de la línea.

Arriba la nube de puntos antes de utilizar el filtro de reducción de densidad y abajo después. Como se puede observar se eliminan puntos en paralelo. Ya que, en caso de utilizarlos para dibujar una polilínea, podría crear una polilínea con muchos picos.

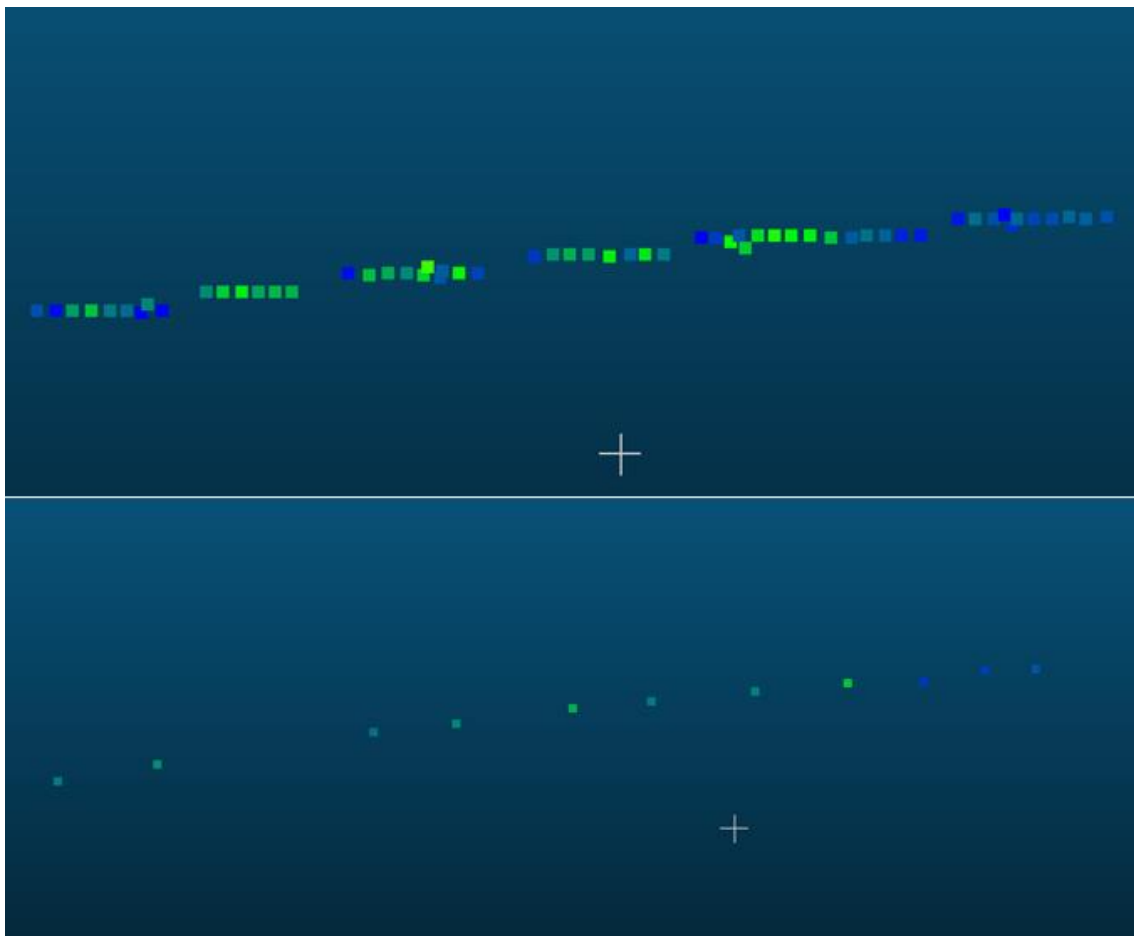


Figura 21. Línea blanca tras aplicarle el filtro "VoxelGrid"

El último paso, cuando ya se tenía la segmentación de la nube con la que se va a trabajar, es realizarle una traslación. Se realiza con la función "Affine3f", esta función en realidad es para realizar una transformación con rotación y traslación, como en este caso no hay rotación, sólo será una traslación. Como se puede comprobar, estos valores coinciden con los que indicaba Cloud Compare que iba a desplazar la nube, pero con el signo cambiado.

```
Method #2: using an Affine3f
  1    0    0 950222
  0    1    0 962246
  0    0    1 670.07
  0    0    0    1
```

Figura 22. Traslación de la nube de puntos *.PCD.

El tiempo para obtener la línea blanca fue de aproximadamente 5 minutos.

Posteriormente se ha extraído el aglomerado, se ha utilizado el filtro de paso "Pass Through" con los valores de intensidad entre 1000 a 10000.

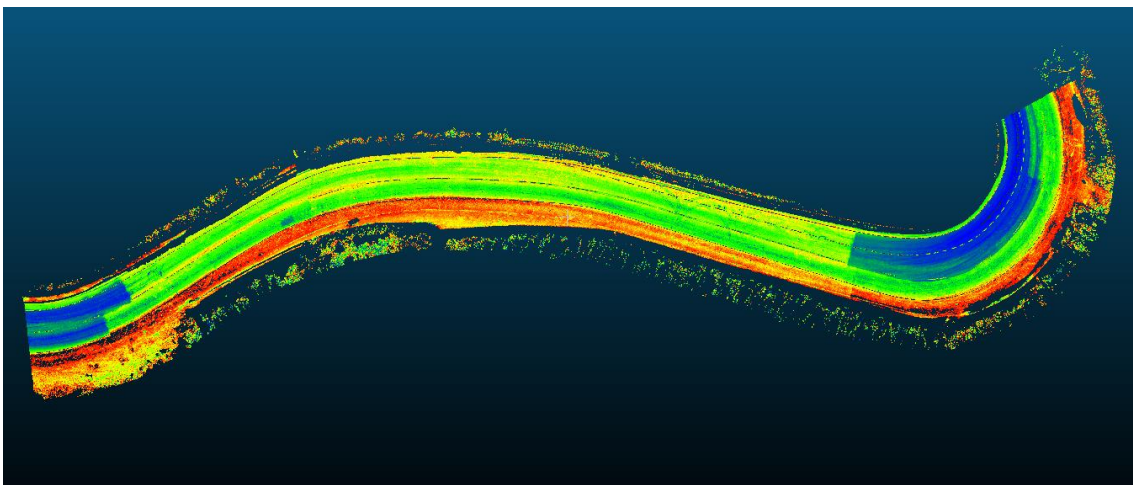


Figura 23. Segmentación asfalto.

Como se puede observar, esta nube tiene mucho ruido por los bordes, es por ello que se le aplica un filtro de reducción de ruido "Statistical Outlier Removal", pero este filtro se aplica dos veces iterando el resultado tras aplicarle el primer filtro, se hace de esta manera ya que tras numerosas pruebas es como mejor eliminaba el ruido. Siendo el resultado el siguiente:

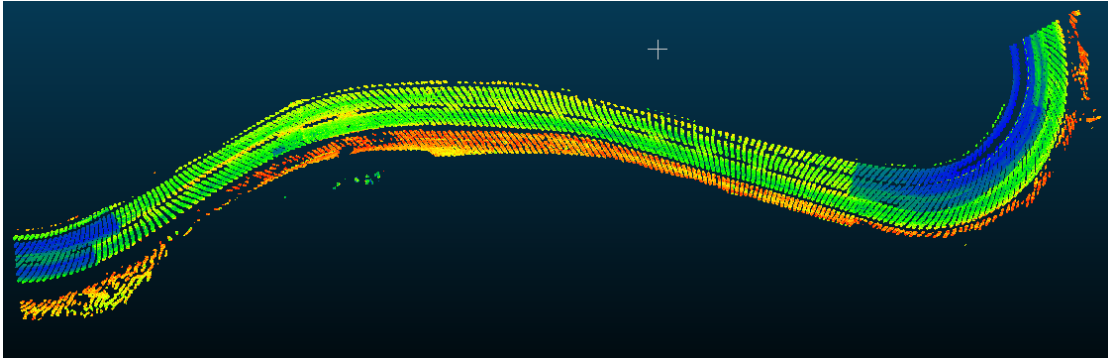


Figura 24. Segmentación asfalto sin ruido.

La primera vez que se aplica el filtro de reducción de ruido se utilizan los valores de 50 puntos vecinos y 0.1 de multiplicador de desviación estándar, la segunda vez también son 50 puntos vecinos, pero con 0.03 como multiplicador, haciéndolo de esta manera se elimina la gran mayoría de ruido alrededor del asfalto.

Si bien hay menos cantidad de puntos, son suficientes para definir correctamente el contorno.

El contorno se extrae con el filtro "Concave Hull", el cual extrae el contorno cóncavo de la nube de puntos, este contorno coincide con el contorno del borde de asfalto.



Figura 25. Contorno de asfalto.

Antes de exportar los puntos, se les aplica la traslación para que los puntos se desplacen a sus coordenadas reales.

Este tiempo de procesado fue de 15 minutos.

Posteriormente, estos puntos se exportan como un archivo ASCII, en formato *.PCD.



```
1 # .PCD v0.7 - Point Cloud Data file format
2 VERSION 0.7
3 FIELDS x y z intensity
4 SIZE 4 4 4 4
5 TYPE F F F F
6 COUNT 1 1 1 1
7 WIDTH 394
8 HEIGHT 1
9 VIEWPOINT 0 0 0 1 0 0 0
10 POINTS 394
11 DATA ascii
12 950263.56 962183.75 651.27466 Asf
13 950263 962185.44 651.25653 Asf
14 950262.81 962186.25 651.25214 Asf
15 950262.75 962187.38 651.26337 Asf
16 950262.81 962187.62 651.26807 Asf
```

Figura 26. Formato de fichero *.PCD.

Este archivo se va a importar con MDT de Aplitop, MDT es una aplicación que funciona sobre Autocad.

Para importar estos puntos se va a crear una plantilla de importación que se adapte a la estructura de los datos. En este caso, que los puntos empiezan en la fila 12 y que la estructura de los puntos es "X-Y-Z-Codigo".

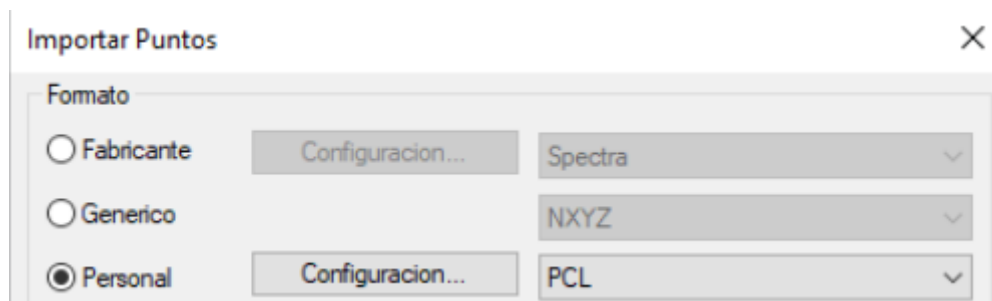


Figura 27. Importar puntos MDT.

Tras definir el formato de importación "PCL", se importan los puntos.

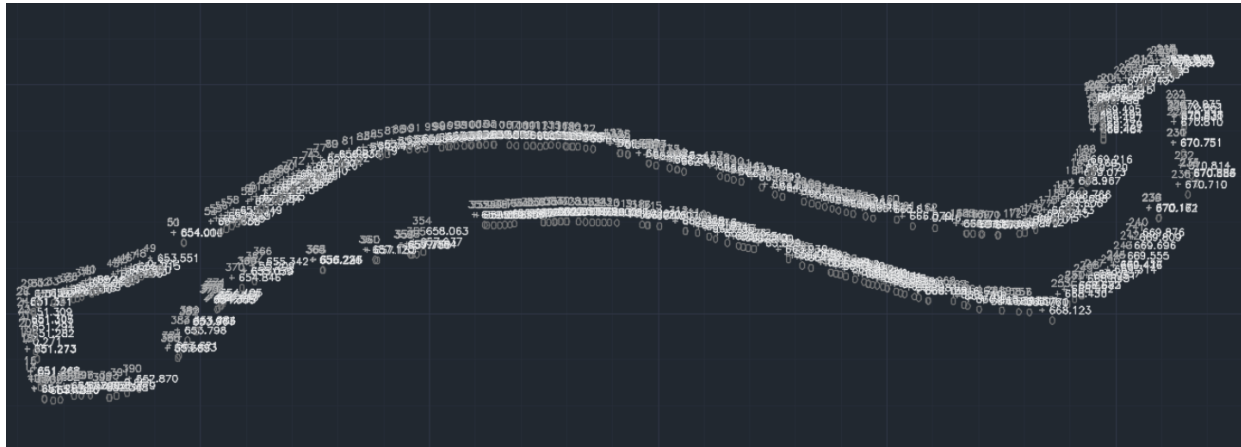


Figura 28. Puntos importados en Autocad/MDT.

Se han importado todos los puntos y como todos tienen el mismo código, se puede utilizar la herramienta de MDT "Dibujar por código", para utilizar esta herramienta hay que definir un código y una capa en la que se va a dibujar, en este caso la capa fue "Asfalto" y se dibujaba a partir de los puntos con código "Asf" y se obtiene la siguiente polilínea.

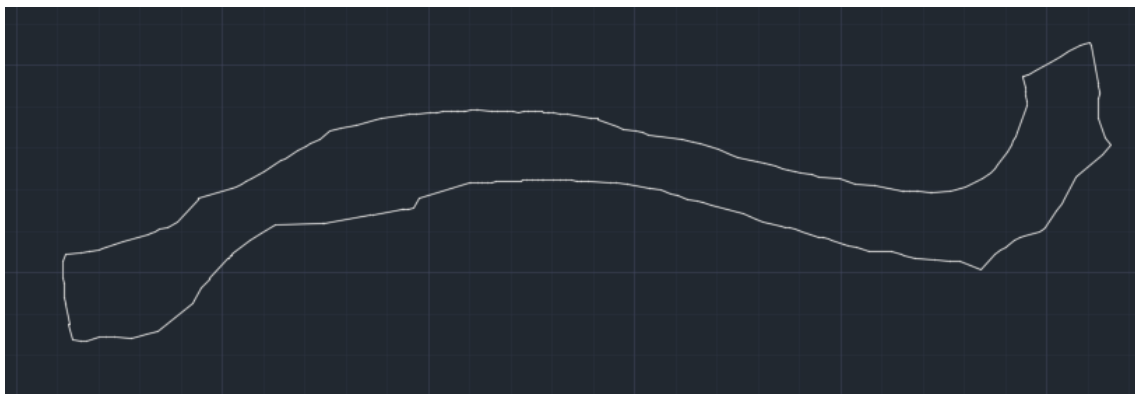


Figura 29. Polilínea del contorno de asfalto.

El resultado es el esperado, una polilínea con le contorno de la nube de puntos.

Finalmente, se obtiene una superficie, que es el MDT, para obtenerlo se ha partido de la nube "Terreno", y se le ha aplicado el filtro "VoxelGrid" para reducir la densidad de punto. La idea es obtener una cuadrícula de 0,75 m x 0.75 m. Esta densidad se considera suficiente para mantener los detalles de la nube y a su vez poder obtener un curvado fiable. Tras aplicar este filtro el numero de puntos es de 17323 puntos.

Se les aplica la translación a estos puntos y se exportan.

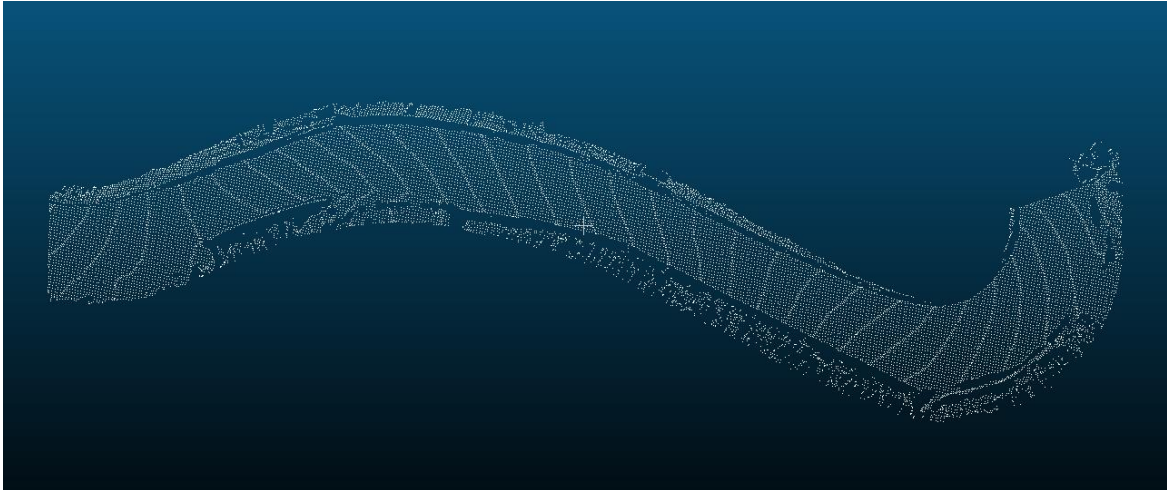


Figura 30. Puntos reducidos para obtener MDT.

Posteriormente, se importan estos puntos, de la misma manera que se han importado los puntos del contorno, y se crea una superficie con estos puntos.

Fichero Origen	
Mínimo	Máximo
X: 950260.625	X: 950520.188
Y: 962179.813	Y: 962264.625
Z: 650.788	Z: 681.298
17323 puntos. Aproximadamente 0.8 puntos/m ²	
Categorías...	
Superficie Destino	
Porcentaje de puntos a importar	100.0
<input type="checkbox"/> Distancia mínima entre vértices	0.000
<input type="checkbox"/> Contorno	Designar <
<input type="checkbox"/> Ventana	Designar <
<input type="checkbox"/> Longitud máxima	0.000
Aproximadamente 17323 vértices en la superficie	

Figura 31. Crear superficie en Autocad/MDT.

La superficie obtenida es la siguiente:

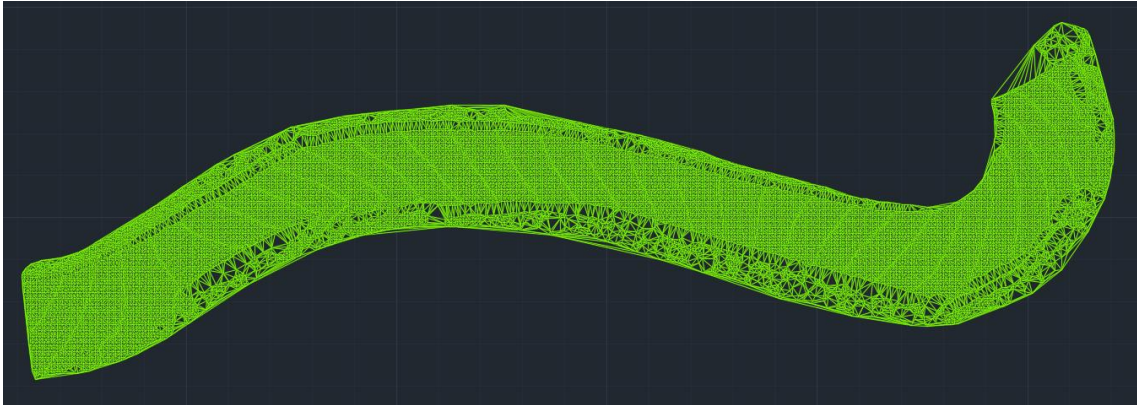


Figura 32. Superficie - MDT.

Tras obtener la superficie, se puede exportar como un formato de superficie (LandXML, por ejemplo) o se podría obtener un curvado, en este caso se va a obtener un curvado para, posteriormente compararlo con el curvado obtenido de la delineación del trabajo realizado en 2020.

El curvado se ha definido con curvas simples cada metros y curvas maestras cada cinco metros, lo que se esperaría en una cartografía escala 1:1000.

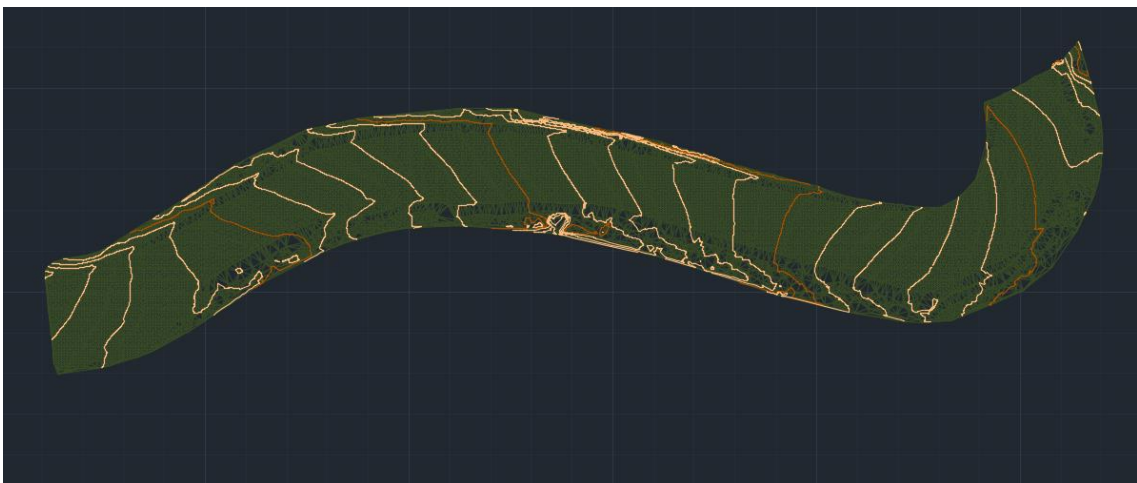


Figura 33. Curvado.

7. Comparación de resultados

Como se mencionó anteriormente, estas nubes de puntos pertenecen a un trabajo realizado a principios de 2020.

Por lo cual se pueden comparar si realmente las nubes segmentadas ayudarían a reducir la carga de trabajo, viendo que tan fiables son comparándolas con las polilíneas que se dibujaron para ese trabajo.

Primero se va a comparar la extracción de la línea blanca obtenida, contra la línea blanca dibujada en su momento.

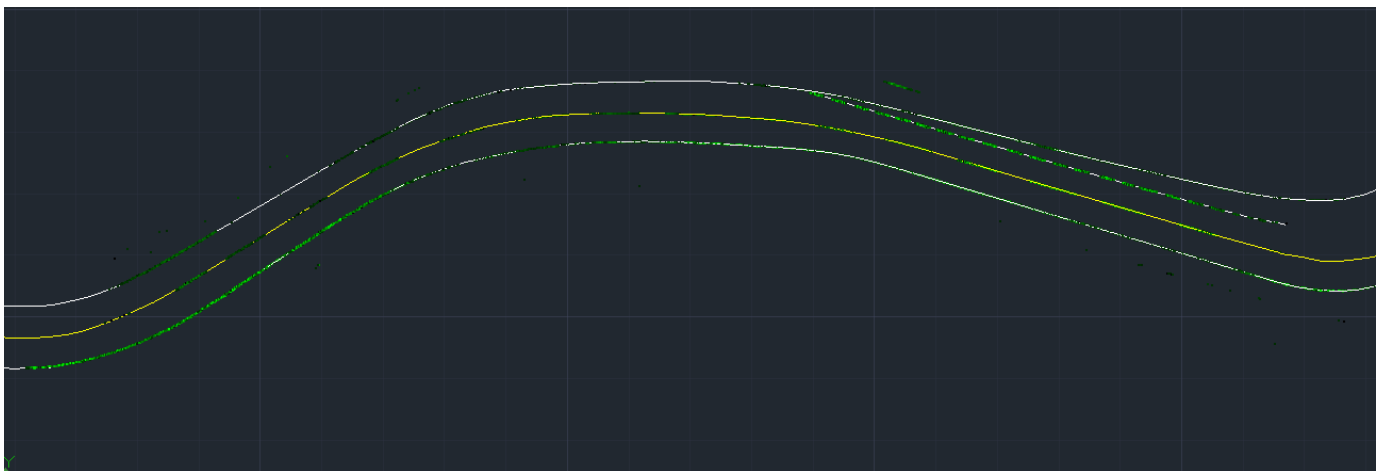


Figura 34. Comparación de nube de puntos con delineación.

La nube de puntos se ve en un tono verde, y se puede apreciar que la extracción coincide bastante bien con la línea blanca dibujada.

Lo siguiente que se puede comparar es la polilínea del borde de asfalto obtenida con la delineada.

En la siguiente imagen se puede ver en rojo la delineación original y en gris, la obtenida a partir de los puntos de contorno, por el superior se ajusta bastante bien. Sin embargo, en la zona inferior varía un poco.

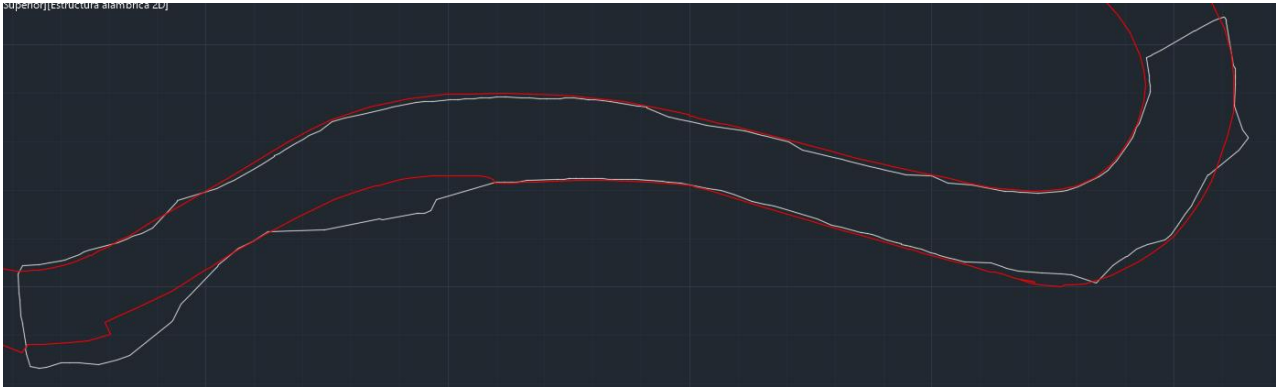


Figura 35. Comparación de bordes de asfalto.

En cuanto a las superficies de los Modelos Digitales del Terreno obtenidos, son bastante similares, y se va a comparar ambos curvados, para ver si las curvas de nivel coinciden en ambas delineaciones.

Curvado del trabajo realizado en 2020:

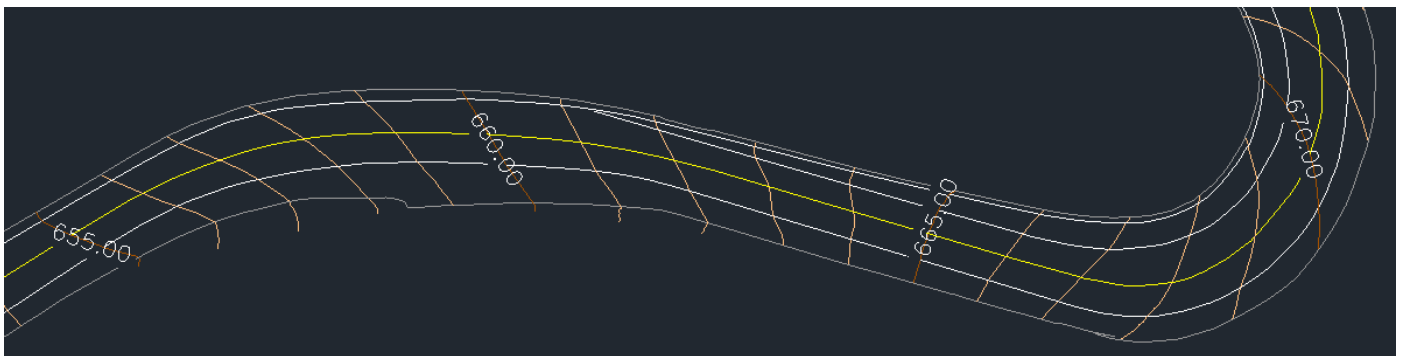


Figura 36. Delineación, trabajo 2020.

Curvado obtenido tras extraer el suelo y disminuir la densidad de puntos:

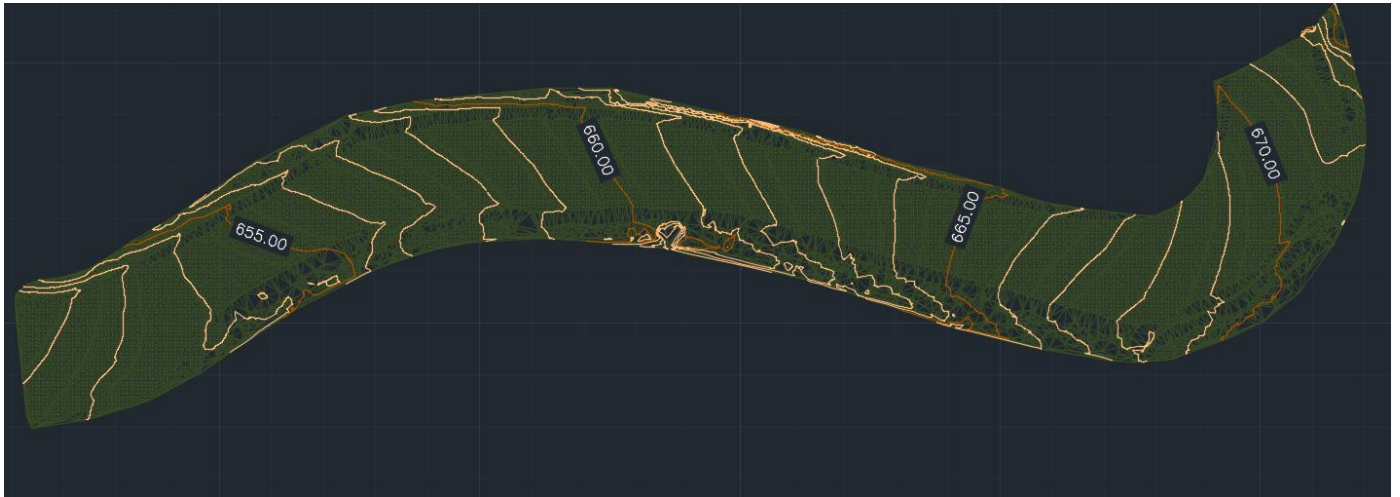


Figura 37. Curvado TFM.

Las cotas de las curvas de nivel maestras coinciden en ambos, se va a superponer ambos curvados y se observa que coinciden las curvas en su totalidad, por lo cual se podría dar por bueno el MDT.

En las siguientes imágenes se ve en naranja y marrón la curva simple y maestra del curvado obtenido de la nube de puntos y en amarillo y azul el curvado obtenido de la delineación.



Figura 38. Comparación de curvados.



Figura 39. Comparación curvados, detalle.

También se ha realizado otra comparativa, con el software 3DReshaper. Este software permite comparar superficies, por lo que se han seleccionado ambas superficies y se han comparado. Como las diferencias de puntos más grandes era de 0,5 metros, se han elegido tres intervalos, de 0 a $\pm 0,04\text{m}$ (verde), de $\pm 0,04\text{m}$ a $\pm 0,10\text{m}$ (amarillo) y de $\pm 0,10\text{m}$ a $\pm 0,50\text{m}$ (rojo).

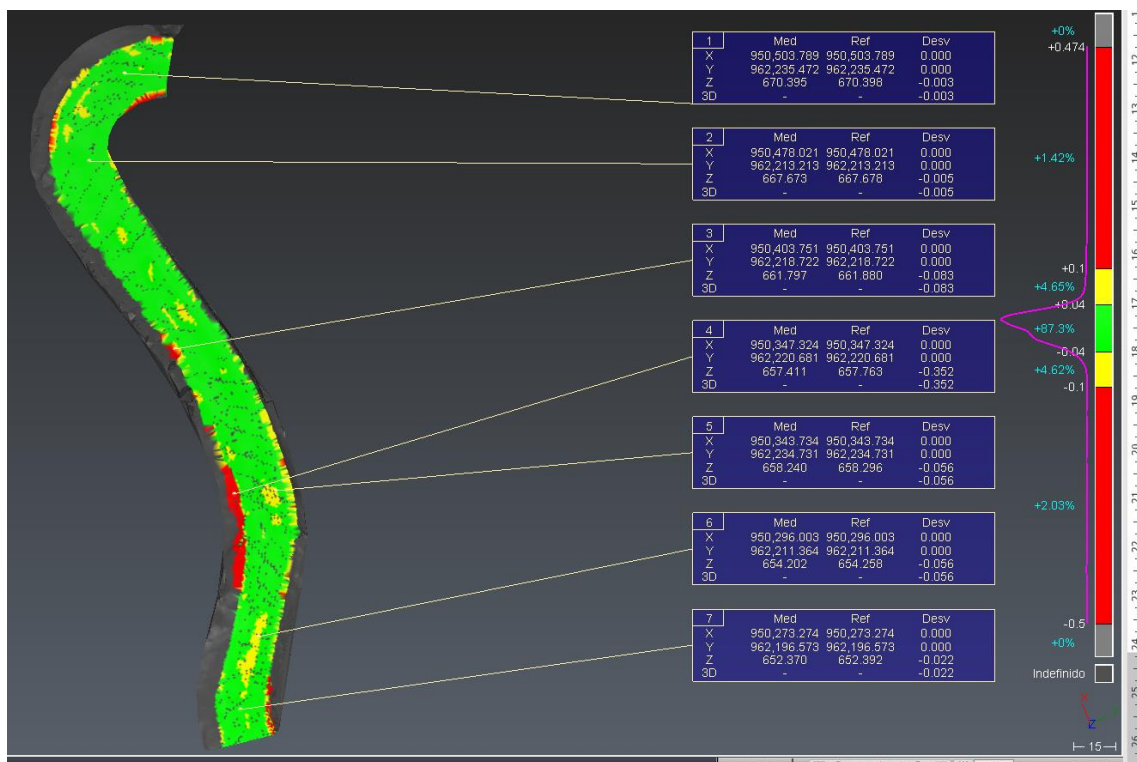


Figura 40. Comparación de superficies.

Como se puede observar el resultado es bastante correcto, ya que el 87,30% de los puntos se encuentran con diferencias por debajo de 0,04m.

Y si observamos las mayores diferencias, en rojo, se encuentran en los bordes, esto se puede deber a restos de vegetación o bionda no eliminados correctamente. Los puntos amarillos, que generalmente se encuentran por el centro, se pueden deber a que con la superficie de la nube de han elegido todos los puntos, mientras que la otra superficie fue obtenida a través de las líneas de rotura digitalizadas, por lo que, si en la carretera había baches o sobresaltos, por el deterioro, estos no se hubieran visto reflejados.

Porcentaje	Intervalo (m)
87.30%	De 0 a ± 0.04
9.27%	De ± 0.04 a ± 0.1
3.43%	De ± 0.1 a ± 0.5

Tabla 1. Diferencias de puntos entre ambas superficies.



8. Conclusiones

En este TFM se ha desarrollado un algoritmo que ayude de una forma sencilla y práctica a reducir el trabajo que conlleva trabajar con una de nubes LiDAR, en este caso, capturada con un sistema MLS.

Si bien el resultado ha sido el esperado, ya que en el caso de la línea blanca se ha podido extraer bastante bien, no se ha podido encontrar una manera semiautomática de que esta se dibuje para obtener una polilínea.

En el caso del borde de asfalto, sí que se ha conseguido una metodología, apoyándose en MDT, gracias a la cual es posible extraer el borde de asfalto, ajustándose al resultado esperado.

En cuanto al MDT y al curvado, se ha obtenido de forma bastante sencilla y comparando curvados, se ajusta y coincide con el resultado del trabajo real, obteniendo un 87.30% de puntos a una distancia inferior a 0.04 metros.

Este trabajo se enfoca como un punto de partida para continuar sumando filtros y metodologías que sirvan para simplificar el trabajo con nubes de puntos.

El archivo *.CPP (cloud_viewer.cpp) con el código generado se ha subido a GitHub, como archivo público, en el siguiente enlace: <https://github.com/jorgessaenz/Trabajo-Fin-de-Master>.



9. Desarrollo futuro y mejoras

La línea más interesante para seguir desarrollando este TFM sería conseguir que además de extraer la línea blanca, se pudiera dibujar de una forma sencilla como con el contorno de asfalto. Es más complicado porque a diferencia del contorno de asfalto, en la línea blanca hay tres grupos de nubes de puntos implicados, las líneas exteriores y la del medio.

Sería interesante conseguir separar estos tres grupos para posteriormente, poder dibujarlos por códigos.

Otra mejora interesante, para ampliar este TFM sería crear distintos perfiles de trabajo, en función del equipo utilizado. Para poder tener, por ejemplo, un perfil con datos del Leica Pegasus Two, en cuanto a valores de intensidad óptimos para extracción y por otro lado tener creado un perfil del Trimble MX7 para poder trabajar con datos de este sistema MLS.

De esta manera utilizar como punto de partida lo conseguido y seguir mejorándolo y ampliándolo con nuevas herramientas y funciones.



10. Referencias

- Bitenc, M., Lindenbergh, R., Khoshelham, K., Van Waarden, A.P.(2011).
Evaluation of a LIDAR Land-Based Mobile Mapping System for Monitoring Sandy Coasts.
Remote Sens., 3 (7) (2011), pp. 1472-1491
<https://www.mdpi.com/2072-4292/3/7/1472>
Último acceso: 01/09/2021
- Foucher, P., Sebsadji, Y., Tarel, J. P., Charbonnier, P. & Nicolle, P. (2011).
Detection and recognition of urban road markings using images
Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems, 1747–52.
https://ieeexplore.ieee.org/abstract/document/6082840?casa_token=_Sge77MJg8QAAAAA:9SeFz00uvwQTbBqTYd2TIRpKOLmciabCdev3bHOf9CdAD95TvUfiYJ3Bj2aU3sp6YgSEVGZeiQo
Último acceso: 31/08/2021
- Gonzalez-Jorge, H., I. Puente, I., Riveiro, B., Martinez-Sanchez, J., Arias, P. (2013)
Automatic segmentation of road overpasses and detection of mortar efflorescence using mobile LiDAR data.
Opt. Laser Technol., 54 (2013), pp. 353-361
<https://www.sciencedirect.com/science/article/pii/S0030399213002392>
Último acceso: 02/09/2021
- He, G., Novak, K., & Feng, W. (1993).
On the Integrated Calibration of a Digital Stereo-Vision System.
International Archives of Photogrammetry and Remote Sensing , 29, 139–139.
<http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=05796394CD86186067D1F192C306B23F?doi=10.1.1.471.1414&rep=rep1&type=pdf>
Último acceso: 02/09/2021



- Jaakkola, A., Hyypä, J., Hyypä, H., Kukko, A. (2008).
Retrieval algorithms for road surface modeling using laser-based mobile Mapping.
Sensors, 8 (9) (2008), pp. 5238-5249
<https://www.mdpi.com/1424-8220/8/9/5238/htm>
Último acceso: 02/09/2021
- Martín Jiménez, J. A., Zazo, S., Arranz Justel, J. J., Rodríguez González, P., Gonzales Aguilera, D. (2018)
ISPRS Journal of Photogrammetry and Remote Sensing.
Volumen 146, pp. 334-346.
<https://doi.org/10.1016/j.isprsjprs.2018.10.004>
Último acceso: 01/09/2021
- Mc Elhinney, C., Kumar, P., Cahalane, C., McCarthy, T. (2010)
Initial results from european road safety inspection (Eursi) mobile mapping project. Proceedings of the ISPRS Commission V Mid-Term Symposium Close Range Image Measurement Techniques 38, pp. 440–445.
<https://mural.maynoothuniversity.ie/9271/>
Último acceso: 31/08/2021
- Porras Díaz, H., Sanabria Echeverry, D. Y & Ortiz Ferreira, J. A. (2018)
Tendencia mundial en tecnologías de sistemas de mapeo móvil implementadas con láser.
Avances: Investigación en ingeniería, 15 (1), 204-230
<https://doi.org/10.18041/1794-4953/avances.1.1361>
Último acceso: 02/09/2021
- Puente, I., Gonzalez-Jorge, H., Martinez-Sanchez, j., Arias P. (2013)
Review of mobile mapping and surveying technologies
Measurement, 46 (7) (2013), pp. 2127-2145
<https://www.sciencedirect.com/science/article/pii/S0263224113000730>
Último acceso: 02/09/2021
- Tao, C. V. (2000).
Mobile mapping technology for road network data acquisition.
Journal of Geospatial Engineering, 2, 2, 1–14.



<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.470.8627&rep=rep1&type=pdf>

Último acceso: 03/09/2021

- Tao, C. V., Chapman, M. a., & Chaplin, B. a. (2001).

Automated processing of mobile mapping image sequences.

ISPRS Journal of Photogrammetry and Remote Sensing, 55, 330–346.

[https://www.sciencedirect.com/science/article/pii/S092427160100026](https://www.sciencedirect.com/science/article/pii/S0924271601000260)

[0](#)

Último acceso: 03/09/2021

- Varela-Gonzalez, M., González-Jorge H., Riveiro B., Arias P. (2014)

Automatic filtering of vehicles from mobile LiDAR datasets

Measurement, 53 (2014), pp. 215-223

[https://www.sciencedirect.com/science/article/pii/S026322411400137](https://www.sciencedirect.com/science/article/pii/S0263224114001377)

[7](#)

Último acceso: 01/09/2021



11. Bibliografía

- Concave Hull Filter, **Point Cloud Library**
https://pointclouds.org/documentation/tutorials/hull_2d.html
Último acceso: 03/09/2021
- Docs, Introduction, **Point Cloud Library**
<https://pcl.readthedocs.io/projects/tutorials/en/latest/>
Ultimo acceso: 31/08/2021
- Leica Pegasus Two, **Leica Geosystems España**
https://leica-geosystems.com/es-es/products/mobile-sensor-platforms/capture-platforms/leica-pegasus_two
Último acceso: 01/09/2021
- Pass Through Filter, **Point Cloud Library**
<https://pointclouds.org/documentation/tutorials/passthrough.html>
Último acceso: 03/09/2021
- Progressive Morphological Filter, **Point Cloud Library**
https://pointclouds.org/documentation/tutorials/progressive_morphological_filtering.html
Último acceso: 03/09/2021
- Statistical Outlier Removal, **Point Cloud Library**
https://pointclouds.org/documentation/tutorials/statistical_outlier.html
Último acceso: 03/09/2021
- Using a matrix to transform a point cloud, **Point Cloud Library**.
https://pointclouds.org/documentation/tutorials/matrix_transform.html
Último acceso: 01/09/2021
- Voxel Grid, **Point Cloud Library**
https://pointclouds.org/documentation/tutorials/voxel_grid.html



Último acceso: 03/09/2021

- Waypoint Software User Guide, **Novatel**.
https://docs.novatel.com/Waypoint/Content/PDFs/Waypoint_Software_User_Manual_OM-20000166.pdf

Ultimo acceso: 31/08/2021