



Universidad de Oviedo

**EPM**  
ESCUELA  
POLITECNICA DE  
MIERES



# DETECCIÓN AUTOMÁTICA DE CAMINOS FORESTALES A PARTIR DE UNA NUBE DE PUNTOS LiDAR

---

TRABAJO FIN DE MÁSTER

MÁSTER EN TELEDETECCIÓN Y SISTEMAS DE INFORMACIÓN GEOGRÁFICA

Autor: Isabel Gómez Pérez

Tutor: Celestino Ordóñez Galán

Cotutor: Carlos Cabo Gómez

Julio, 2019

## ÍNDICE

<b>RESUMEN</b> .....	<b>1</b>
<b>ABSTRACT</b> .....	<b>2</b>
<b>1. INTRODUCCIÓN</b> .....	<b>3</b>
1.1. Tecnología LiDAR .....	4
1.1.1 LiDAR aerotransportado .....	5
1.2. LiDAR en la detección de pistas forestales .....	10
<b>2. OBJETIVOS</b> .....	<b>11</b>
<b>3. MATERIALES Y MÉTODOS</b> .....	<b>12</b>
3.1. Zona de estudio .....	12
3.2. Fuentes de datos .....	14
3.3. Softwares empleados .....	14
3.4. Metodología para la detección de pistas forestales .....	16
3.4.1 Preprocesamiento de datos LiDAR.....	17
3.4.1.1. Voxelización.....	18
3.4.1.2. Generación MDE y mapa de pendientes .....	21
3.4.1.3. Generación imagen RGB e Intensidad .....	23
3.4.2 Detección automática de las pistas forestales .....	25
3.4.3 Restauración de la información basado en Tensor-Voting .....	37
3.4.4 Vectorización de las pistas forestales detectadas.....	41
<b>4. RESULTADOS</b> .....	<b>42</b>
4.1. Detección de pistas forestales.....	42
4.2. Unión de tramos inconexos mediante Tensor Voting.....	44
4.3. Evaluación en la detección de las pistas forestales.....	46
<b>5. CONCLUSIONES</b> .....	<b>50</b>
<b>6. BIBLIOGRAFÍA</b> .....	<b>51</b>
<b>ANEXO: CÓDIGO MATLAB</b> .....	<b>57</b>

## ÍNDICE DE FIGURAS

Figura 1: Esquema de trabajo de un sistema LiDAR aerotransportado. Fuente: (Albacete, 2011).....	6
Figura 2: Partes de un sistema LiDAR. Fuente: <a href="http://www.modusrobotics.com/lidarsystems/">www.modusrobotics.com/lidarsystems/</a> ...	6
Figura 3: Interacción de un pulso láser con la vegetación (Volsselman y Maas 2010).....	8
Figura 4: Zona de estudio (mapa de situación). .....	13
Figura 5: Zona de estudio (plano topográfico). .....	13
Figura 6: Diagrama de la metodología empleada en la detección de pistas forestales.....	17
Figura 7: Esquema gráfico de la posición de 8 puntos de ejemplo en voxelización. Fuente: Cabo et al. (2014).....	19
Figura 8: Generación del código de voxelización, a) se muestra el proceso de codificación, b) muestra el código de los puntos, c) muestra el vector ordenado d) los vectores de correspondencia entre puntos de la nube y vóxeles. ....	19
Figura 9: Nube de puntos más bajos. ....	21
Figura 10: Modelo Digital de Elevaciones obtenido a partir de la nube de puntos LiDAR.....	22
Figura 11: Filtro para el cálculo del gradiente en $\nabla_x$ (O-E) y en $\nabla_y$ (S-N).....	22
Figura 12: Mapa de pendientes, a partir del MDE. ....	23
Figura 13: Imagen RGB generada a partir de la nube de puntos LiDAR con color.....	24
Figura 14: Imagen de intensidad. ....	25
Figura 15: Histograma ideal (bimodal). ....	26
Figura 16: Histograma imagen RGB (izquierda) e histograma intensidad (derecha).....	27
Figura 17: Histograma mapa de pendientes .....	27
Figura 18: Resultado de la umbralización a la derecha en la imagen RGB y a la izquierda la imagen intensidad. ....	28
Figura 19: Resultado del proceso de umbralización del mapa de pendientes. ....	29
Figura 20: Imagen intensidad umbralizada (izquierda) y después de eliminar componentes conectados menores a 4 píxeles (derecha). ....	30
Figura 21: Imagen de pendientes umbralizada (izquierda) y después de eliminar componentes conectados menores a 4 píxeles (derecha).....	30

Figura 22: Imagen que muestra la población cercana, selección de las áreas mayores a 900 píxeles e imagen sin las áreas de no interés. ....	31
Figura 23: Funcionamiento de la dilatación. ....	33
Figura 24: Funcionamiento de la erosión. ....	34
Figura 25: Funcionamiento del cierre. ....	35
Figura 26: Imagen binaria intensidad entrante, operación de dilatación y operación de cierre. ....	35
Figura 27: Imagen binaria pendientes entrante, operación de dilatación y operación de cierre. ....	35
Figura 28: Resultado de la operación de esqueletonización. ....	36
Figura 29: Resultado de la eliminación de las ramificaciones que se producen después de la esqueletonización. ....	37
Figura 30: Visualización del tensor en 2 dimensiones. La representación de un tensor de unión (bola) y la representación de un tensor de curva (barra). ....	39
Figura 31: Mapa de saliencia o prominencia. ....	40
Figura 32: Imagen final algoritmo TV. ....	40
Figura 33: Resultados de la detección de caminos forestales mediante Matlab. ....	42
Figura 34: Caminos no detectados, probablemente por la reducción de ruido en imágenes. ....	42
Figura 35: Visualización de las pistas forestales detectadas. ....	43
Figura 36: Zonas donde los caminos se encuentran ocluidos por la vegetación (izquierda). Detección de caminos en la misma zona (derecha). ....	44
Figura 37: Resultados sobre las discontinuidades con el algoritmo TV. ....	45
Figura 38: Pistas forestales detectadas (rojo) y tramos de conexión (azul) tras la aplicación del algoritmo TV. ....	45
Figura 39: Visualización de las pistas forestales detectadas y las pistas reales digitalizadas. ....	46
Figura 40: Criterios de valoración en la detección de pistas. El primero caso considerado con TP, el segundo ejemplo de FP y el último ejemplo de FN. ....	47
Figura 41: Caminos no detectados en zonas de vegetación frondosa, y zonas de repoblado donde se produce la detección. ....	49

## RESUMEN

La necesidad de disponer de una buena cartografía de los caminos forestales es de suma importancia, debido a que estas infraestructuras permiten el acceso a los montes, lo que a su vez es necesario para realizar labores de conservación de los montes, la gestión integral en el aprovechamiento de sus recursos, el uso social y recreativo de estos espacios, labores de prevención y extinción de incendios, y labores de custodia y defensa.

En la actualidad, no se dispone de cartografía digitalizada de la red viaria forestal en Asturias, por lo que a través de este trabajo se evaluará la posibilidad del uso de la tecnología LiDAR (Light Detection and Ranging), como recurso para la obtención de la misma. Partiendo exclusivamente de la nube de puntos LiDAR se obtiene un Modelo Digital de Elevaciones (MDE), una imagen a color, una imagen de intensidad y un mapa de pendientes de la zona de estudio, que se combinan mediante técnicas procesamiento de imágenes para extraer las pistas forestales. La metodología propuesta se evalúa en un monte de la Comunidad Autónoma de Asturias. Para ello, los resultados obtenidos se comparan con una cartografía digitalizada manualmente a partir de una ortofoto, y se proporcionan tres indicadores (integridad, exactitud y calidad) para cuantificar el grado de ajuste entre ambos conjuntos de datos.

## ABSTRACT

The need of having a good cartography of forest roads is of the utmost importance, because these infrastructures allow access to hills, which in turn is necessary to carry out perform forestry conservation works, the integral management in the use of its resources, the social and recreational use of these spaces, and do tasks of prevention and extinction of fires and work of custody and defense.

At present, there is no digitized cartography of the forest track network in Asturias, so through this work will evaluate the possibility of using LiDAR (Light Detection and Ranging) technology as a resource for obtaining the same. Starting exclusively from the LiDAR points cloud, we obtain a Digital Elevation Model (DEM), a color image, an intensity image and a slope map of the study area are obtained, which are combined by image processing techniques to extract the forest tracks. The proposed methodology is evaluated in a mountain of the Autonomous Community of Asturias. For this, the results obtained are compared with a manually digitized cartography from an orthophoto, and three indicators are provided (integrity, accuracy and quality) to quantify the degree of adjustment between both sets of data.

# 1. INTRODUCCIÓN

La red de caminos y pistas forestales forman parte de la infraestructura de los montes, de las que depende el acceso seguro al bosque para una mejor gestión y aprovechamiento de los montes, facilitan la prevención y extinción de incendios forestales, siendo además un auténtico componente vertebrador del territorio.

El mantenimiento y construcción de estos caminos lleva englobado un elevado coste, tanto ambiental como económico. Por todo ello, se considera que la planificación de la red de pistas forestales es de vital importancia para llevar a cabo una gestión sostenible y eficiente del monte (Ferraz et al. 2016). En este sentido, Laschi et al. (2016) reconocen en su estudio que el primer paso a dar en esta dirección es el reconocimiento de la estructura y características de la red de pistas a fin de evaluar el estado actual de accesibilidad a las diferentes áreas y las necesidades reales de uso de las pistas forestales.

La carencia de un inventario detallado y actualizado de la red viaria forestal de la mayoría de las Comunidades Autónomas es uno de los principales obstáculos para desarrollar una correcta planificación y gestión de los montes.

A través de este estudio se pretende realizar una detección automática de pistas forestales, a partir de una nube de puntos LiDAR (Light Detection and Ranging), con la finalidad de dar el primer paso propuesto por los autores referidos en los párrafos anteriores.

En el ámbito de la detección automática de pistas, la tecnología LiDAR, se presenta como un complemento efectivo a las técnicas clásicas de teledetección y resulta especialmente valiosa en zonas forestales, porque permite construir modelos digitales de elevaciones (MDE) a partir de los que se pueden realizar inventarios de caminos forestales detallados y precisos (Gallay et al. 2013). White et al. (2010), mostró la idoneidad de LiDAR para trazar mapas de caminos forestales, frente a los datos tradicionales de detección remota, como las imágenes de satélites y la fotografía aérea, ya que los sensores pasivos no pueden penetrar en el dosel denso.

En España se dispone de datos LiDAR de acceso libre que se distribuyen a través del Centro de Descarga del Centro Nacional de Información Geográfica (CNIG) dentro del Plan Nacional de Ortofotografía Aérea (PNOA). Actualmente está disponible para Asturias los datos LiDAR del vuelo realizado en 2012. Las nubes de puntos han sido capturadas mediante vuelos con sensor LiDAR y una densidad de puntos de 1-0,5 puntos/m<sup>2</sup>.

### 1.1. Tecnología LiDAR

También pueden encontrarse referencias al LiDAR como “laser scanning” (LS) o LaDAR (Laser Detection and Ranging) (Wehr and Lohr, 1999). Se trata de un sistema activo de teledetección, es decir, utiliza su propia fuente de energía, y se basa en el mismo principio básico utilizado por la tecnología radar, aunque empleando energía láser (Light Amplification by Stimulated Emission of Radiation) en vez de energía emitida en la región microondas. La mayoría de los sistemas LiDAR empleados en aplicaciones topográficas y forestales, trabajan en la región del espectro electromagnético correspondiente al infrarrojo cercano (IRC), normalmente entre las longitudes de onda de 1040 nm y 1065 nm, de modo que una particularidad principal de los sistemas LiDAR es que están afectados por las condiciones atmosféricas (nieve, nubes, etc.).

La aparición de los escáneres láser está arraigada a la historia de la tecnología LiDAR. En 1975, la NASA y otras organizaciones desarrollaron un sistema LiDAR oceanográfico aerotransportado para medir la concentración de clorofila y otras sustancia biológicas y químicas. Entre los años 1970 y 1980, se utilizó LiDAR para estudios batimétricos, forestales y otras aplicaciones (Solodukhin et al. 1985, Nelson et al. 1984, Schreier et al. 1985), que establecieron los principios básicos del uso de láseres con el objeto de la detección remota. A pesar de ello, no fue hasta el año 1990 cuando se llevaron a cabo los primeros experimentos con instrumentos de escaneado láser. Los sistemas de medida LiDAR comenzaron a ser unidos a sistemas ópticos-mecánicos de escaneado/barrido, elementos de posicionamiento global y unidades de medida inercial, estableciéndose así los primeros escáneres láser aéreos (Bufton 1989, Flood y Gutelius 1997, Hyyppä et al. 2008).

Entre los sistemas LiDAR, hay una diferenciación en función de la situación del escáner láser en la plataforma. Autores como Hyppä et al. (2013), realizan una clasificación entre tres categorías, los escáneres láser aerotransportados (ALS), escáneres láser terrestres (TLS) y escáneres láser móviles (MLS). Cuando se refiere al sistema LiDAR del PNOA, se trata de sensores láser aerotransportados.

#### 1.1.1 LiDAR aerotransportado

Su funcionamiento es teóricamente simple, un dispositivo láser realiza medidas mediante un barrido consecuencia de un movimiento oscilatorio y del avance del propio avión. En este barrido, todo objeto que encuentre en su trayectoria ofrecerá una reflexión del rayo y, en consecuencia, una medida representada, después de múltiples cálculos, en un punto con sus coordenadas XYZ (Figura 1). Los objetos quedan definidos por el número de puntos que han reflejado en sus superficies, normalmente llamado “nube de puntos”, cuya definición fue propuesta por Jon Mills y David Barber en 2003: “la nube de puntos es un conjunto de coordenadas XYZ en un sistema de referencia común, que le muestra al espectador una comprensión de la distribución espacial de un objeto en un sitio. También se puede incluir información adicional como un valor de intensidad o un valor RGB”. Esta particularidad representa la principal ventaja del LiDAR y, a la vez, su principal inconveniente. Para transformar esta nube de puntos en un documento compresible, útil para trabajos posteriores debe tratarse mediante los procesos adecuados para separar los puntos de los diferentes objetos de interés, de aquellos que puedan ser innecesarios.

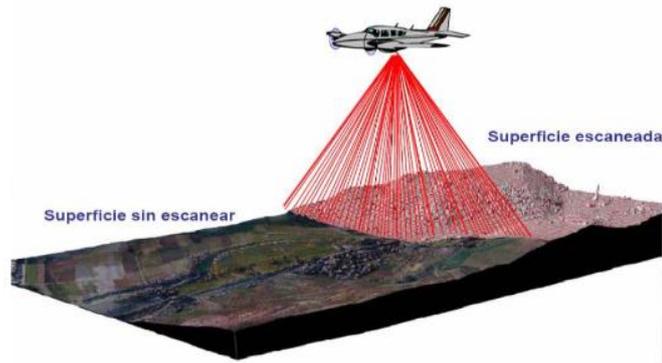


Figura 1: Esquema de trabajo de un sistema LiDAR aerotransportado. Fuente: (Albacete, 2011)

Las partes que componen un sistema LiDAR aerotransportado (Figura 2) incluyen generalmente un sensor láser, un receptor GPS (Global Positioning System) para medir la posición precisa de la plataforma y un sistema de navegación inercial (INS, Inertial Navigation System) o un GNSS/IMU (Unidades de Medidas Inerciales) que mide la distancia y la orientación de un punto respecto a su origen y sistema de referencia (Conama,2016). Estos sensores se integran dentro del propio vehículo aéreo para obtener información del movimiento del propio vehículo y su localización.

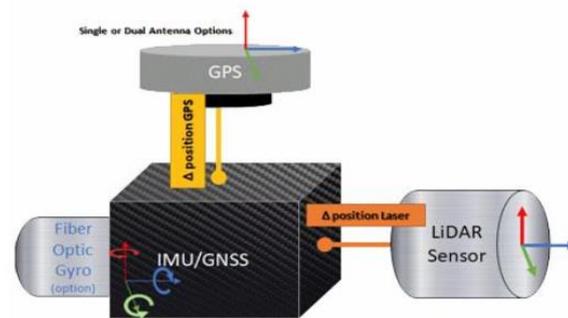


Figura 2: Partes de un sistema LiDAR. Fuente: [www.modusrobotics.com/lidarsystems/](http://www.modusrobotics.com/lidarsystems/)

La información recibida por el sensor láser combinada con la información de la posición proporcionada por los sistemas (GNSS/IMU) hace que sea posible la transformación de estas medidas en puntos tridimensionales reales del objeto (Romero et al. 2009).

El uso de datos provenientes de ALS, se ha incrementado en las últimas décadas, debido a la gran utilidad de esta tecnología para la producción de Modelos Digitales

de Elevaciones (MDE) de alta resolución y al papel que juegan estos modelos en muy variados ámbitos que van desde la generación de modelos tridimensionales, hasta la planificación de auxilio y ayuda en caso de desastres naturales (Fernández-Rivas et al, 2010).

Una de las ventajas de la tecnología LiDAR aerotransporta, procede de su capacidad para atravesar las cubiertas vegetales (Albacete, 2011), por tanto, en la posibilidad de medir la altura del terreno en áreas completamente cubiertas por arbolado o matorral no denso, lo cual no es posible con técnicas de medición clásicas como la fotogrametría. Esta es una de las principales motivaciones por las que los sistemas LiDAR se están imponiendo a la hora de generar modelos de elevaciones destinados a usos que no requieren más de 10-15 cm de precisión Z. Generalmente y debido a la alta densidad de puntos generada por el sensor LiDAR, incluso si solo un pequeño porcentaje de puntos alcanza el suelo suele ser suficiente para crear un Modelo Digital del Elevaciones (Landa et al. 2013).

La densidad de puntos de la nube es la manera de expresar la resolución espacial de un conjunto de datos LiDAR en un número de puntos por  $m^2$ . Son diversos los autores los que evaluaron la influencia de la densidad de puntos LiDAR en el ámbito del reconocimiento de objetos de interés (Sherba et al. 2014, Triglav et al. 2010), en ambos estudios no establecen una densidad óptima de puntos, pero sugieren valores superiores a los 4 puntos/ $m^2$ .

La principal característica de la tecnología LiDAR aerotransportada en zonas forestales es que permite capturar múltiples retornos por cada pulso emitido (Figura 3). De manera que, el rayo láser choca con la copa del árbol, parte de este se refleja y vuelve al avión (primer retorno), y otra parte sigue atravesando la vegetación hasta llegar al suelo (último retorno), por ser el árbol una superficie “no dura y porosa”, varios retornos vuelven al avión (retornos intermedios), se llama múltiples retornos y pueden registrar hasta cinco retornos por pulso o ilimitados según el sensor utilizado (Lefsky et al. 2002).

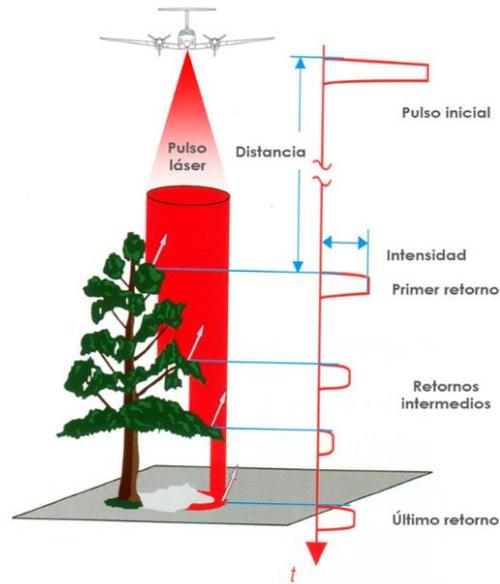


Figura 3: Interacción de un pulso láser con la vegetación (Volselman y Maas 2010)

Con esta característica, en zonas de vegetación puede pensarse que el último retorno siempre corresponde a la reflexión producida por el suelo, pero en la práctica, la densidad de la cubierta vegetal juega un papel decisivo. Según autores como Chasmer et al. (2006), sólo el 50% de los últimos retornos en zonas forestales proceden del suelo, por lo que sería necesario utilizar técnicas de clasificación para determinar que retornos pertenecen realmente a la superficie del suelo. Otros autores como Watt et al. (2013), concluyen en la dificultad que presenta esta tecnología para penetrar en la densa estructura vegetal que oculta al suelo y, por consiguiente, la proporción de retornos pertenecientes al suelo en estas zonas es muy reducida.

Los ALS no solo puede proporcionar información sobre la estructura forestales, sino que también mediciones de la vegetación y los objetos artificiales. También son útiles para cartografiar objetos largos y estrechos como son las pistas forestales o líneas eléctricas, que apenas pueden detectarse en imágenes ópticas (Maltamo et al. 2014).

La intensidad de los datos LiDAR es el relación entre la cantidad de energía procedente de los objetos o del terreno captada por el sensor y la cantidad de energía total emitida por el sensor (Singh et al., 2010). Los valores de intensidad se ven

afectados por la orografía del terreno, las características del vuelo y del sensor y también por las condiciones atmosféricas (Song y Civco, 2004). La imagen de intensidades permite realizar distinciones entre superficies como, por ejemplo, la identificación de carreteras (Fernández y Siabato, 2011).

Autores como Jeong-Heon Song et al. (2006) realizaron estudios fundamentándose en productos derivados de datos LiDAR como la imagen de intensidad, desarrollando el autor una metodología enfocada a la diferenciación de usos del suelo, partiendo de la premisa de que cada elemento posee una refracción singular en base al material que está compuesto, lo que permitió desarrollar un método en el cual se realizó una primera clasificación del suelo según su uso en base a la intensidad de los retornos LiDAR. En las conclusiones a su trabajo, mostraron que la intensidad LiDAR estaba influenciada por mucho ruido, además de no ajustarse exactamente a las leyes de la reflectividad, puesto que, debido a la eliminación del ruido mediante métodos de suavizado, se pierde parte de la información.

Otro inconveniente que aparece al tratar con nubes de puntos LiDAR es el enorme tamaño de los datasets (Elseberg *et al.*, 2013), interfiriendo en la eficiencia de los algoritmos de procesamiento de la nubes de puntos. Esto llega a convertir en inviables o no rentables algunos métodos por el tiempo de procesado que requerirían. Algunos algoritmos publicados en los últimos años proponen y utilizan métodos, estrategias o estructuras de simplificación de las nubes de puntos. Dentro de estas estas estructuras cabe destacar, las basadas en la implantación de una malla de ortoedros o cubos, como voxels (Cabo *et al.*, 2014), los octrees (Elseberg *et al.*, 2013; Meagher, 1982) o los kd-trees (Nuchter *et al.*, 2007). La principal diferencia entre estas estructuras se encuentra en el tamaño y forma de cubos u ortoedros y las relaciones de vecindad.

## 1.2. LiDAR en la detección de pistas forestales

Existe una variedad de estudios disponibles sobre detección de caminos forestales a partir de la tecnología LiDAR aerotransportada, surgidos en los últimos años. White et al. (2010), realizaron un estudio en la zona forestal de Santa Cruz (California), concluyendo que a partir de MDEs, derivados de LiDAR, se pueden revelar redes de carreteras ubicadas bajo el denso dosel del bosque, por la capacidad de los sensores activos de penetrar en la vegetación, frente a los datos procedentes de satélites o imágenes aéreas. Autores como Azizi et al. (2014), desarrollaron una metodología para la extracción de pistas forestales, a partir de una clasificación supervisada, empleando datos LiDAR de alta densidad, y proporcionando un enfoque para la realización de inventarios viales completos y precisos en la gestión forestal.

Con relación a la influencia de la densidad de la nube de puntos de partida, Sherba et al. (2014), evaluaron la calidad reduciendo la densidad de su nube de puntos de 1,2 puntos/m<sup>2</sup> a 0,3 puntos/m<sup>2</sup>, sugiriendo valores de densidades para la detección en zona forestales de 4 puntos/m<sup>2</sup>.

Kiss et al. (2015), realizaron estudios evaluando la calidad de la detección de caminos en zonas forestales, donde obtuvieron al menos un 66% de caminos forestales detectados correctamente.

Ferraz et al. (2016) abordaron la problemática de la detección y caracterización de los caminos en entornos boscosos a grandes escalas (>1000 km<sup>2</sup>) empleado datos LiDAR de alta densidad, y demostraron una detección de caminos correctos mayores al 80%, con un 10-15% de los caminos detectados erróneamente, utilizando métodos completamente automáticos.

## 2. OBJETIVOS

El objetivo general de este estudio es establecer una metodología sencilla de clasificación no supervisada, para detectar de forma automática y precisa pistas forestales empleando exclusivamente datos LiDAR del PNOA.

Los objetivos específicos perseguidos con este trabajo son:

- Detección de las pistas forestales de una nube de puntos generada con datos LiDAR mediante técnicas de procesamiento de imágenes.
- Utilización de un algoritmo basado en la restauración de la información faltante conocido como *Tensor Voting*, para lograr una continuidad en la red pistas detectadas.
- Evaluar la efectividad en la metodología de la detección de pistas forestales mediante el uso de datos LiDAR para disponer de información estandarizada y digitalizada de los caminos forestales que dan servicio a los montes.

### 3. MATERIALES Y MÉTODOS

#### 3.1. Zona de estudio

Para evaluar la metodología propuesta se ha utilizado una zona de estudio que se encuentra en el noroeste de Asturias (Figura 4), entre los concejos de Pravia y Muros de Nalón. La parroquia más cercana es Somado, perteneciente al concejo de Pravia. Limita al noroeste con el concejo de Muros de Nalón, al este y al sur con la parroquia de Santines, al suroeste con la de Villafría y al oeste y noroeste con la de la Piñera (Cudillero).

La zona forestal objeto de estudio es el Monteagudo (Figura 5), es un monte en los que la especie de mayor volumen (74% del total aprovechado) es el eucalipto (*Eucalyptus*), en el que también predomina el aprovechamiento de *Pinus pinaster* y *Pinus radiata*. Es un monte de Utilidad Pública que comprende los concejos de Pravia y Muros de Nalón con una superficie total de 179 ha.

En la zona de estudio también se encuentran los montes Peñaforada y El Carbón pertenecientes al Concejo de Pravia cuya parroquia más cercana es Los Cabos. En estos montes también se pueden visualizar caminos forestales y cuya especie predominante es el eucalipto y no son montes de Utilidad Pública.



Figura 4: Zona de estudio (mapa de situación).

ZONA DE ESTUDIO EXTRACCIÓN DE PISTAS FORESTALES

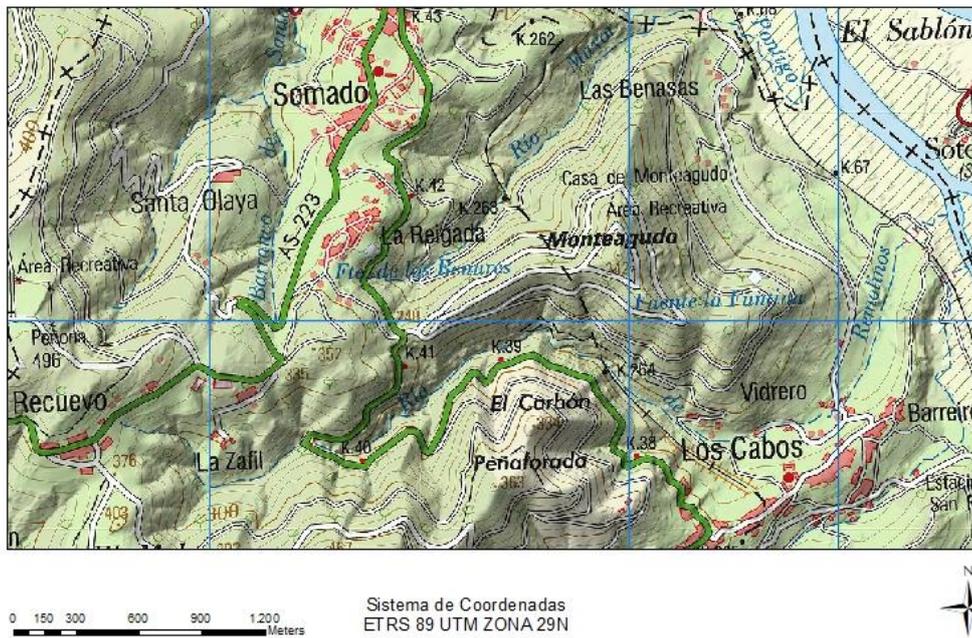


Figura 5: Zona de estudio (plano topográfico).

### 3.2. Fuentes de datos

Los datos de partida de los que se establece una metodología para la detección de pistas forestales y evaluación de los resultados son:

- Datos LiDAR de la zona de estudio correspondiente al vuelo del PNOA (Plan Nacional de Ortofotografía Aérea) realizado en 2012 (1ª Cobertura). Estos datos son obtenidos a través del Centro de Descarga del CNIG en ficheros digitales de 2x2 km de extensión y una densidad de puntos de 1- 0,5 puntos/m<sup>2</sup>. Los archivos que se presentan en formato LAZ (formato de compresión de los ficheros LAS), con alturas elipsoidales y sistema geodésico de referencia ETRS89 en proyección UTM y en huso 29 N. La organización encargada de la creación, distribución y actualización de este formato es ASPRS (*American Society for Photogrammetry & Remote Sensing*). Este formato constituye un formato estándar de intercambio de nubes de puntos tridimensionales. Almacena por cada punto de nube valores X, Y, Z, además de los valores de intensidad, retorno, número de retorno, ángulo de escaneado, dirección del escaneado, dirección de vuelo e información GPS, logrando también almacenar campos definidos durante el procesamiento de la información (ASPRS, 2013).

- Ortofotografía del PNOA e imagen ráster georreferenciada del Mapa Topográfico Nacional 1:25.000 (MTN25), correspondiente a la hoja 13 de la zona de estudio, obtenidos a través del Centro de Descarga del CNIG, como producto de apoyo en la digitalización de una red real de los caminos forestales. Son imágenes ráster en formato ECW (Enhanced Compresión Wavelet), propiedad de Hexagon Geospatial.

### 3.3. Softwares empleados

Para llevar a cabo la extracción automática de pistas forestales y su validación, nos hemos apoyado en tres programas informáticos cuyas características esenciales, y el uso que le hemos dado en este trabajo, se indican en las siguientes líneas.

## - **CloudCompare**

Es un software libre de procesamiento de nube de puntos 3D. Originalmente se diseñó para realizar comparaciones entre dos nubes de puntos densas (como las obtenidas por un escáner láser), o entre una nube de puntos y una malla triangular. Posteriormente se ha extendido a un software de procesamiento de nube de puntos más genérico, que incluye muchos algoritmos avanzados como registro, remuestreo, cálculo de estadísticas y normales, entre otros.

Se trata de un software destinado hacer frente al procesamiento de enormes nubes de puntos, por lo general, más de 10 millones de puntos y hasta 120 millones con 3 gigabytes de memoria.

Mediante este software se carga la nube de puntos completa en formatos LAZ, descargada del CNIG y se exporta en formato texto, para llevar a cabo la extracción de los objetos de la nube puntos más completa en un formato accesible en Matlab.

## - **Matlab**

Matlab es el nombre abreviado de “Matrix Laboratory”, es un software matemático para realizar cálculos numéricos con vectores y matrices y para aplicaciones científicas y de ingeniería. Matlab posee un lenguaje de programación propio (lenguaje M), muy próximo a los habituales en el cálculo numérico (Fortran, C, etc.). Su interés radica en los cientos de funciones tanto de propósito general como especializadas que posee, así como una amplia variedad para la visualización gráfica en 2D o 3D. Entre sus múltiples posibilidades se encuentran la manipulación de matrices, la representación de datos y funciones y la implementación de algoritmos.

A través de este software se realiza el procesamiento de la nube de puntos LiDAR, la obtención de las imágenes o capas de entrada y la detección automática de las pistas forestales de la zona de estudio.

## - **ArcGIS**

Es un software en el campo de los Sistemas de Información Geográfica (SIG) para visualizar, administrar, crear y analizar datos geográficos. Producido y comercializado por ESRI (Environmental Systems Research Institute).

En concreto se utiliza ArcMap 10.3, para la georreferenciación y vectorización final de las imágenes obtenidas en la detección automática de pistas forestales, empleando el módulo ArcScan. También se utiliza para la digitalización aproximada de una red real de caminos forestales con el fin de evaluar la precisión en la detección de caminos, por la localización de entidades bajo indicadores de Integridad, Exactitud y Calidad.

### 3.4. Metodología para la detección de pistas forestales

La metodología llevada a cabo en la detección de pistas forestales se puede dividir en cuatro partes (Figura 6):

- 1) Preprocesamiento de la nube de puntos mediante algoritmos de voxelización. Generación del MDE, mapa de pendientes, imagen RGB e imagen de intensidad.
- 2) Detección automática de las pistas forestales mediante clasificación no supervisada utilizando operaciones morfológicas y de filtrado de imágenes sobre las pistas forestales.
- 3) Restauración de información faltante o deteriorada basada en el algoritmo *Tensor Voting (TV)*.
- 4) Vectorización de las pistas forestales detectadas mediante el software *ArcGis*, para validar los resultados de la metodología propuesta.

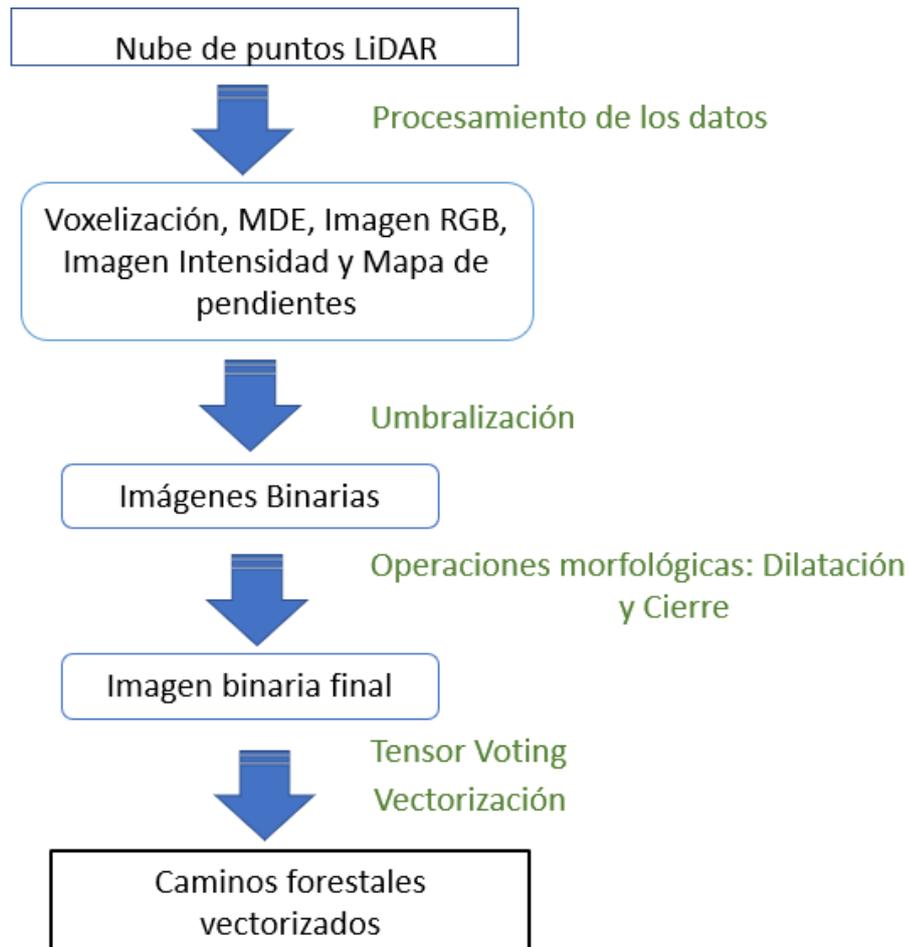


Figura 6: Diagrama de la metodología empleada en la detección de pistas forestales.

### 3.4.1 Preprocesamiento de datos LiDAR

Dado que el almacenamiento y procesamiento de los conjuntos de datos LiDAR de alta calidad, pueden ser difícilmente abordables, con los actuales sistemas de información geográfica de escritorio, existe la necesidad de contar con técnicas para el procesamiento y gestión del almacenamiento de conjuntos de datos masivos LiDAR.

Para ello primero se realiza un proceso de simplificación de la nube de puntos mediante un método de voxelización. Posteriormente se genera un MDE, el mapa de pendientes derivado del mismo, y la imagen RGB y de intensidad de la zona de estudio.

#### 3.4.1.1. Voxelización

La nube de puntos de las mediciones LiDAR obtenida en la zona de estudio es una colección de puntos muy grande, con un total de 2.985.207 puntos. Por esta razón, a menudo es necesario una simplificación que favorezca su tratamiento. Por tanto, un procesamiento de los datos LiDAR que se desarrollan en este trabajo con el software Matlab es un método de voxelización, que permite la generación de una versión reducida de la nube de puntos original. De manera, que se realiza una reducción de los requisitos informáticos al ajustar todos los datos en una red tridimensional regular que no presenta una pérdida de información significativa si la escala se elige correctamente. Sin embargo, los datos originales se conservan y se relacionan directamente con la versión reducida (Cabo, 2017).

El espacio se divide en una cuadrícula tridimensional regular. Como resultado, cada punto individual está dentro de una unidad cúbica volumétrica (es decir, vóxel). De esta modo, la nube de puntos original se simplifica significativamente utilizando una discretización espacial y simplemente almacenando los datos necesarios en las siguientes etapas, aunque los datos iniciales se conserven. El vóxel es la unidad cúbica que compone un objeto tridimensional, constituye la unidad mínima procesable de una matriz tridimensional, y es, por tanto, el equivalente del píxel en un objeto 2D.

El método de voxelización utilizado se basa en la codificación de las coordenadas planimétricas XY de los puntos escaneados en un solo código y su posterior decodificación. El código es un número entero de 5 dígitos como mínimo a 7 dígitos como máximo, que consta de los dos componentes de tres dígitos. Cada componente representa una de las dos coordenadas planimétricas XY (Figura 7 y 8). Las coordenadas se transforman en unidades de vóxel y se reducen al origen de la nube de puntos utilizando valores mínimos de XY (ec. 1). Después, las dos componentes se ensamblan en un código único (Cabo et al., 2014).

Código: YYYYXX

$$XXX = \text{entero} \frac{X - X_{\min}}{\text{Tamaño del voxel}}; \text{YYY} = \text{entero} \frac{Y - Y_{\min}}{\text{Tamaño del voxel}} \times 10000 \quad (\text{ec.1})$$

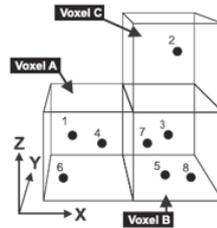


Figura 7: Esquema gráfico de la posición de 8 puntos de ejemplo en voxelización. Fuente: Cabo et al. (2014).

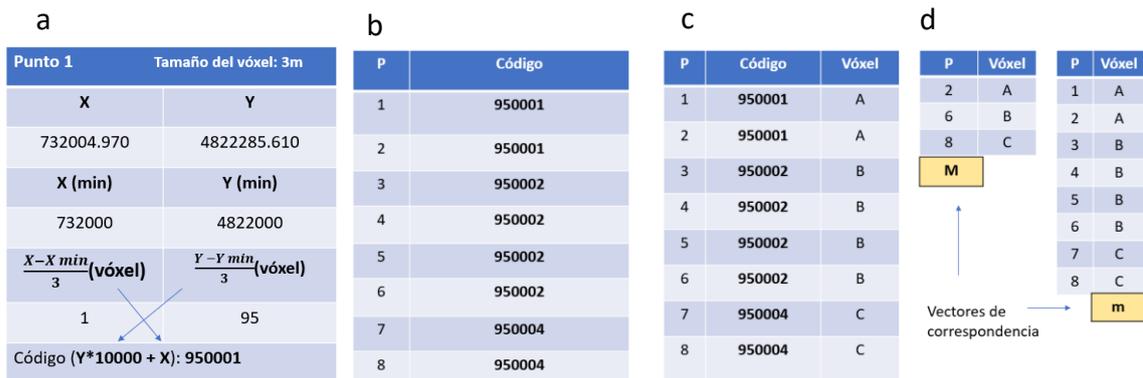


Figura 8: Generación del código de voxelización, a) se muestra el proceso de codificación, b) muestra el código de los puntos, c) muestra el vector ordenado d) los vectores de correspondencia entre puntos de la nube y vóxeles.

Los valores del código se almacenan en un solo vector (**m**). Su longitud es igual al número de puntos en la nube de puntos original (N). Dado que los puntos originales que pertenecen al mismo voxel tienen el mismo código, los elementos del vector se ordenan por su valor numérico. De esta manera, todos los puntos del mismo voxel se colocan juntos. Del vector (Figura 7.c) que contiene valores ordenados, los elementos únicos (es decir, los valores de los códigos que aparecen al menos una vez, pero que pueden estar repetidos si el voxel contiene más de un punto) se extraen y se

transfieren a otro vector (**M**). La longitud del nuevo vector ( $n < N$ ) es igual al número de vóxeles que contienen uno o más puntos.

Por lo que se generan dos vectores (vectores de correspondencia), el vector **M** que almacena los índices del último punto de cada vóxel ( $n$  elementos), mientras que el vector **m** que almacena los índices de los vóxeles que corresponden a cada punto original ( $N$  elementos). Ambos vectores se relacionan a través del identificador de cada punto (**P**).

Finalmente, se revierte el proceso de codificación. Los códigos se dividen y los datos necesarios para las etapas subsiguientes en el espacio de vóxeles calculan: a) las coordenadas de los centroides de los vóxeles que contienen al menos un punto y b) el número de puntos dentro de cada vóxel. De la misma manera, al usar los vectores de correspondencia (es decir, **M** y **m**), cada vóxel está directamente relacionado con sus puntos, y cada punto está relacionado con su vóxel, por lo que los puntos dentro de cualquier vóxel podrían extraerse directamente si se desea.

El proceso de voxelización con el software *Matlab*, se inicia importando el archivo en formato de texto, con tipo de salida "*Numeric Matrix*", de manera que tenemos una matriz de la nube de puntos LiDAR de dimensión 2.985.207 filas y 7 columnas (X, Y, Z, R, G, B, Intensity), cuyas filas se refieren al valor de cada punto obtenido.

La función de voxelización consta de varios pasos. En primer lugar, para ello se crea un nuevo mínimo para restar a las coordenadas de cada columna XY, de esta manera se reducen al origen de coordenadas la nube de puntos.

El segundo paso es generar el código píxel, ordenando los elementos de la matriz, tratando las columnas como vectores y ordenando cada columna, de manera que también redondea la nube de puntos, menos el valor mínimo anteriormente calculado y se divide entre el tamaño de vóxel elegido, que es el tamaño del lado del píxel (3 m). La coordenada Y se multiplica por 10000 para que sumar XY se genere un código único de cada píxel.

El tercer paso es calcular las coordenadas codificadas únicas, es decir, eliminar las repeticiones que se producen cuando más de un punto de la nube de puntos corresponde al mismo vóxel y con un indicador de ocurrencia, indicando que debe contener los índices últimos a los valores repetidos encontrados en la matriz.

Por último, se deshace la codificación teniendo en cuenta el orden en el que fue hecha la codificación. La nube de puntos importada al inicio tenía un tamaño de 2.985.207 puntos y después de la voxelización se redujo a un tamaño de 440.691 puntos.

### 3.4.1.2. Generación MDE y mapa de pendientes

#### - MDE

A partir de la nube de puntos LiDAR de la zona de estudio se genera el Modelo Digital de Elevaciones (Figura 10). Primero se pixeliza la nube de puntos, es decir se ordena los puntos en una retícula regular de lado del píxel del terreno y se ordena los puntos originales según el orden del código de pixelación. A partir de un proceso iterativo se recorren los píxeles buscando el punto más bajo (altura mínima, Z) de cada vóxel, teniendo en cuenta los puntos que corresponderían al terreno (Figura 9).

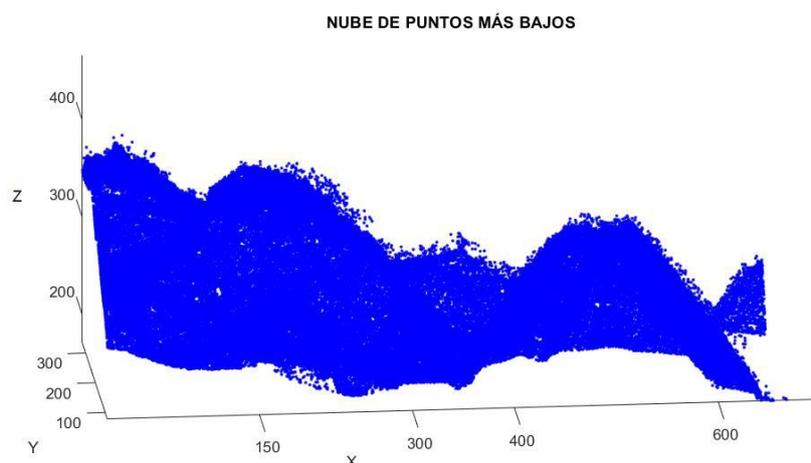


Figura 9: Nube de puntos más bajos.

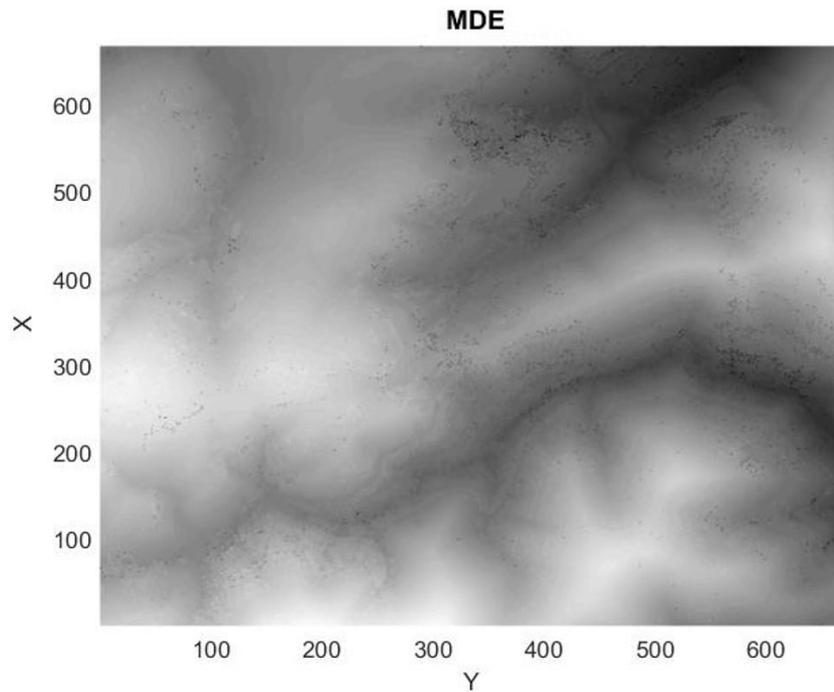


Figura 10: Modelo Digital de Elevaciones obtenido a partir de la nube de puntos LiDAR.

- Mapa de pendientes

Para poder generar el mapa de pendientes se ha utilizado el MDE generado anteriormente con la nube de puntos más bajos de cada píxel (Figura 12). Al MDE se le aplica un tipo especial de filtros en el dominio espacial. En concreto se crea un filtro direccional, de manera que a cada punto  $(x, y)$  del MDE se le calcula separadamente el gradiente de cada punto en el sentido Oeste-Este ( $\nabla_x$ ) y otro en sentido Sur-Norte ( $\nabla_y$ ).

El cálculo del gradiente en cada punto se realiza de manera que tenga en cuenta los 8 vecinos más próximos al punto (Figura 11).

<b>0</b>	<b>0</b>	<b>0</b>
<b>-1/4</b>	<b>0</b>	<b>1/4</b>
<b>0</b>	<b>0</b>	<b>0</b>

<b>0</b>	<b>1/4</b>	<b>0</b>
<b>0</b>	<b>0</b>	<b>0</b>
<b>0</b>	<b>-1/4</b>	<b>0</b>

Figura 11: Filtro para el cálculo del gradiente en  $\nabla_x$  (O-E) y en  $\nabla_y$  (S-N).

Posteriormente a la creación de los filtro direcciones, se filtra la matriz creada para el MDE con las matrices de los gradientes para cada sentido. Por último, se calcula la pendiente de cada píxel mediante la siguiente operación (ec. 2):

$$P = \sqrt{\nabla x^2 + \nabla y^2} \quad (\text{ec.2})$$

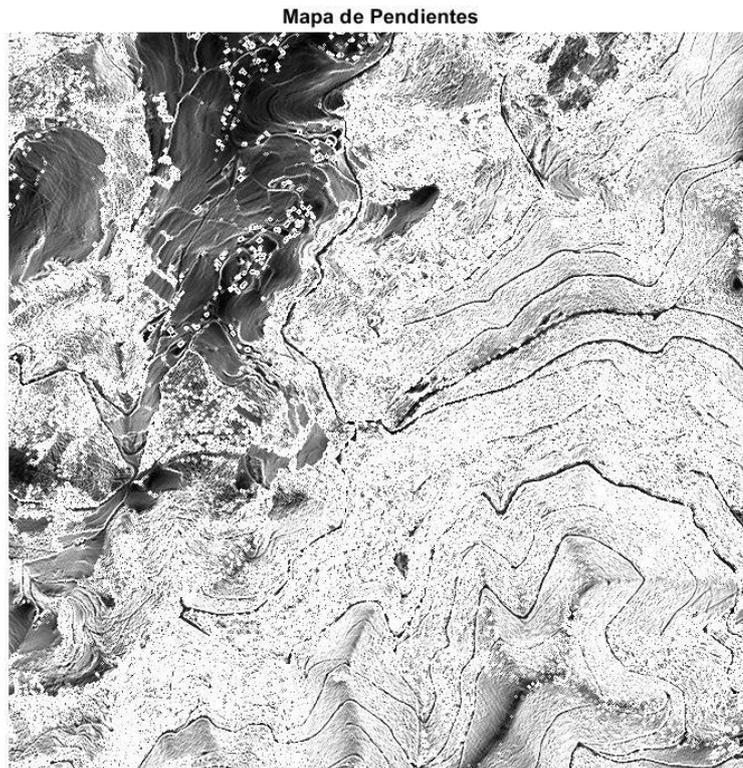


Figura 12: Mapa de pendientes, a partir del MDE.

#### 3.4.1.3. Generación imagen RGB e Intensidad

##### - Imagen RGB

A partir de la nube de puntos pixelada, es decir, ordenados los puntos originales en una retícula regular según el orden del código de voxelización se genera la imagen RGB, mediante un proceso iterativo se recorren los píxeles promediando los niveles digitales de los puntos que están dentro de cada píxel, en cada correspondiente canal de color.

Por último, para poder crear la imagen teniendo sus componentes por separado, se utiliza una concatenación de los tres canales de color (Figura 13).

### IMAGEN RGB



Figura 13: Imagen RGB generada a partir de la nube de puntos LiDAR con color.

#### - Imagen de intensidad

La imagen de intensidad se genera de forma similar a como se hizo con la imagen RGB, con la diferencia de que sólo hay un canal de información, la intensidad, y no tres. La Figura 14 muestra la imagen de intensidad en niveles de gris.

#### IMAGEN INTENSIDAD

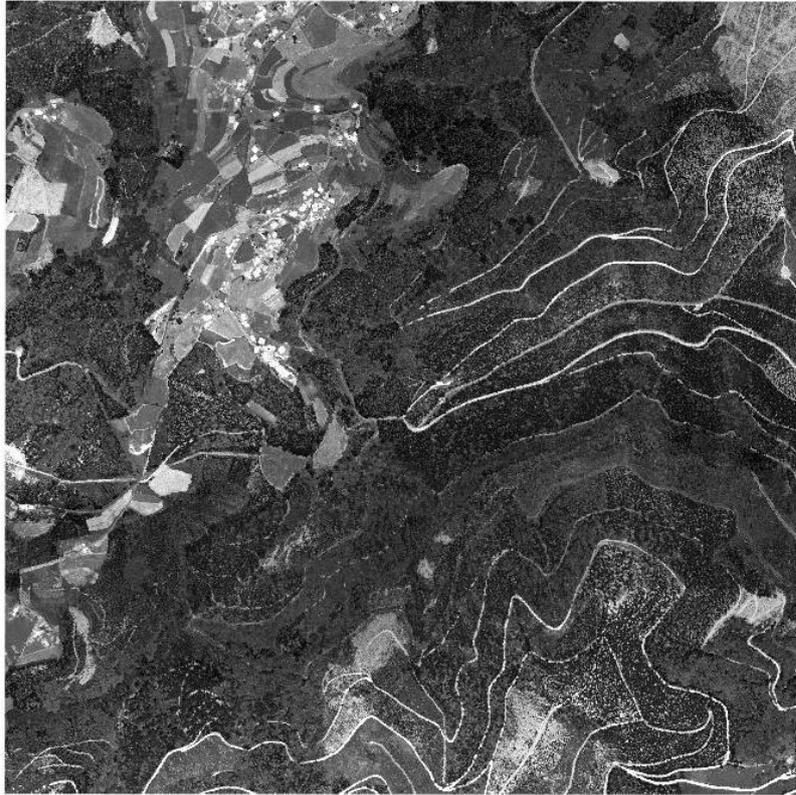


Figura 14: Imagen de intensidad.

#### 3.4.2 Detección automática de las pistas forestales

Para la detección de las pistas forestales se realizan una clasificación no supervisada de las imágenes anteriormente obtenidas, con el fin de extraer los ejes pertenecientes a los caminos y eliminar el máximo ruido posible contenido en las imágenes, de esta manera se consigue el propósito de resaltar las zonas objeto de estudio.

##### A) Umbralización.

La umbralización de imágenes es uno de los métodos para la segmentación de imágenes digitales. Las técnicas de umbralización buscan obtener un valor de umbral que permita binarizar la imagen separando adecuadamente el fondo (background) y el objeto a separar (foreground). Este método tiene en cuenta el nivel de gris de un

píxel para decidir si el mismo pertenece o no al objeto de interés, para ello se debe encontrar el rango de niveles de gris que caracterizan dicho objeto.

El problema de la umbralización es encontrar el valor del umbral adecuado entre los valores de grises en la imagen que permita una óptima separación entre el objeto y el fondo. Las imágenes siempre presentan ruido y, como consecuencia de esto, sus histogramas son difíciles de analizar, es decir, presentan histogramas sin una marcada separación de regiones. El caso ideal sería un histograma de intensidad de una imagen (nivel de gris) bien marcado con dos picos o lóbulos, en los que se pueden separar dos zonas o regiones y el umbral (T) será aquel valor que se encuentre en el valle entre ambas (Figura 15).

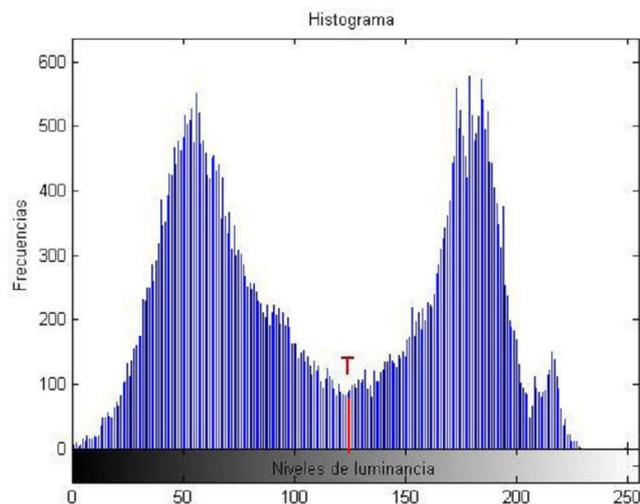


Figura 15: Histograma ideal (bimodal).

Para ello se obtienen previamente los histogramas de la imagen RGB, intensidad e imagen de la pendiente (Figura 16 y 17).

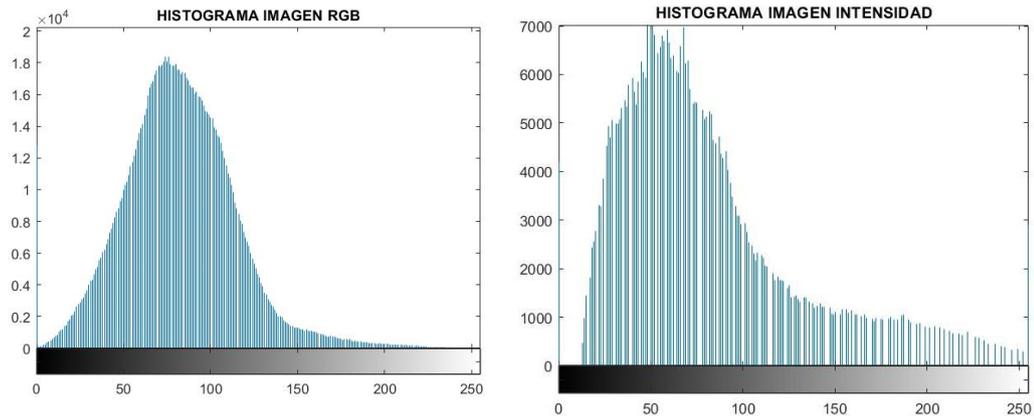


Figura 16: Histograma imagen RGB (izquierda) e histograma intensidad (derecha).

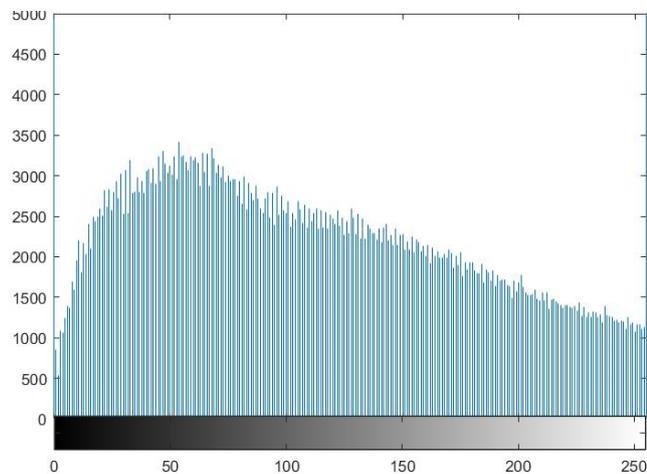


Figura 17: Histograma mapa de pendientes

Como se puede observar en los histogramas de las diferentes imágenes (Figura 16 y 17), no presentan en alguna zona de la distribución una concavidad en la curva como en el histograma bimodal. Por tanto, a partir de un análisis visual de las distintas imágenes se comprueba el valor de los píxeles en las zonas donde se encuentran los caminos forestales para cada una de las imágenes, y tienen valores aproximadamente constantes (altos en imagen RGB y de intensidad y bajos en imagen de pendientes) y son diferenciables de otras estructuras que no se quieren resaltar.

Por lo que para la imagen RGB e imagen intensidad se realiza una función de umbralización con un percentil alto de 97.5% y 90%, respectivamente, el cuál asigna

1 a los valores de la imagen por encima de ese percentil, que corresponden, en general, a las pistas forestales. Al resto de las celdas se les asigna valor 0 (Figura 18).

Previamente a la umbralización, la imagen RGB se convierte a escala de grises, y se ajustan valores de de contraste especificando los límites.

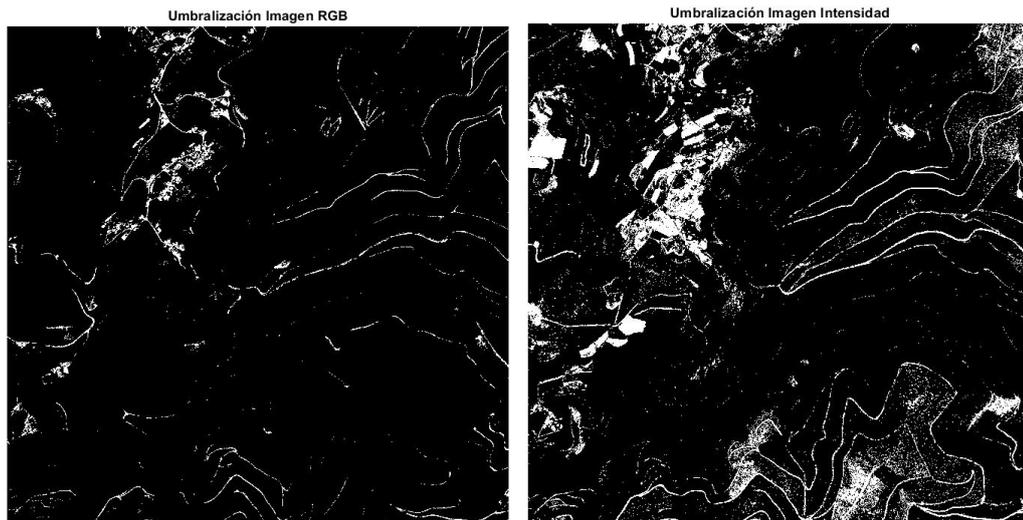


Figura 18: Resultado de la umbralización a la derecha en la imagen RGB y a la izquierda la imagen intensidad.

Para umbralizar la imagen de pendientes se ha utilizado un percentil bajo de 20%, asignando 1 a las celdas con valores de la imagen por debajo de ese percentil. Estas celdas corresponden, en gran medida a las pistas forestales, ya que tienen una pendiente baja en relación con los taludes del terreno adyacente. El resto de celdas toman valor 0 (Figura 19).

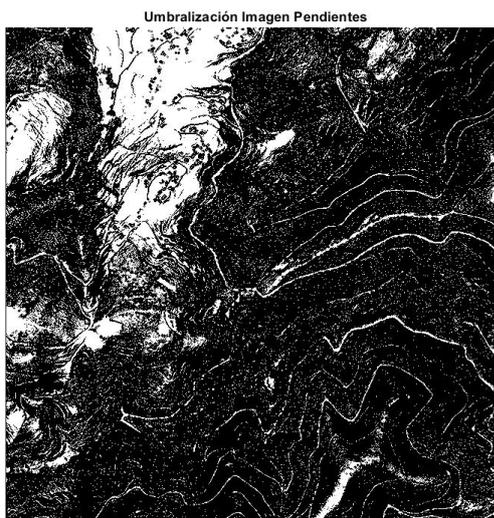


Figura 19: Resultado del proceso de umbralización del mapa de pendientes.

Tras el proceso de umbralización, en todas las imágenes se observa la pérdida de muchas partes de los caminos forestales de interés, esto es debido a diferentes causas, como que los caminos estén cubiertos por vegetación densa o que no exista suficiente contraste entre las caminos con las áreas de alrededor, por lo que al final del trabajo se propone un algoritmo de restauración de la información faltante para obtener mejores resultados.

#### B) Filtrado de las imágenes.

En esta etapa se pretende resaltar de forma selectiva información contenida en las imágenes después del proceso de umbralización que sea de interés para la correcta extracción de los caminos forestales o, por el contrario, eliminar aquella que no los sea.

Después de la umbralización existen numerosos píxeles aislados que no pertenecen a los caminos y por tanto su extracción no es de interés para el estudio. Para su eliminación se utilizan funciones para el borrado de objetos conectados, de manera que se elimina de una imagen binaria todos los objetos conectados que tiene menos del número de píxeles marcado por el usuario en la operación, produciendo otra imagen binaria. En nuestro caso se aplica a la imagen de intensidad y pendientes una eliminación de agrupaciones de píxeles de 4 píxeles (Figura 20 y 21), porque producen ruido y así se consigue aislar mejor las pistas forestales.

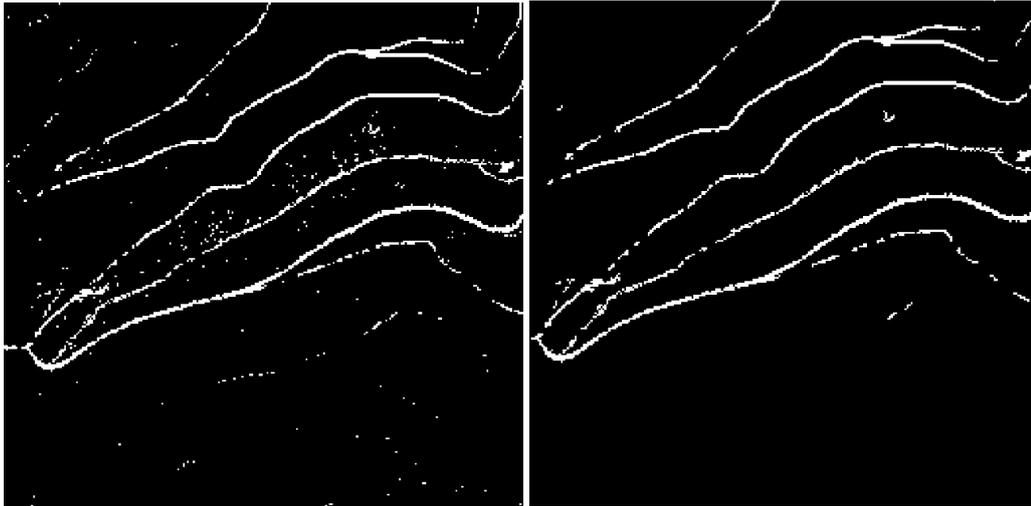


Figura 20: Imagen intensidad umbralizada (izquierda) y después de eliminar componentes conectados menores a 4 píxeles (derecha).

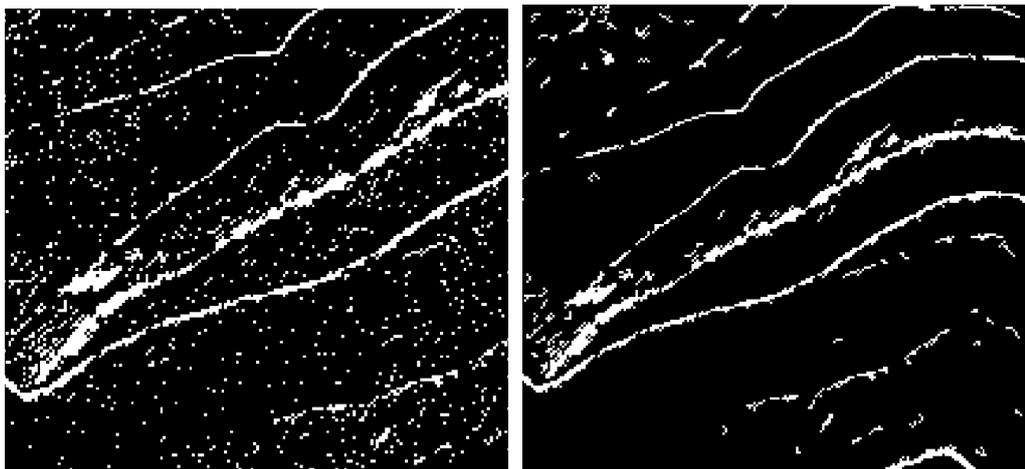


Figura 21: Imagen de pendientes umbralizada (izquierda) y después de eliminar componentes conectados menores a 4 píxeles (derecha).

Además, estas dos imágenes después de la umbralización muestran las zonas pertenecientes a la población, próximas a la zona de estudio que no son objeto de interés en la extracción, debido a que presentan características de intensidad y pendiente similar a los caminos forestales. Por tanto, se procede a su eliminación mediante la búsqueda de componentes conectadas a una imagen binaria, con una conectividad de vecindad de 8 píxeles. A continuación, se mide las características de los objetos en la imagen con conectividad por su área. Por último, se crea una imagen binaria la cual contenga sólo los objetos o regiones que cumplan un determinado

criterio. Se impone como criterio un área mayor o igual a 900 píxeles, de esta manera se consigue, eliminar las máximas regiones pertenecientes a la población sin eliminar las pistas forestales de interés (Figura 22).

El resto de las estructuras visibles y que no son objeto de estudio, se eliminan durante el proceso de vectorización.



Figura 22: Imagen que muestra la población cercana, selección de las áreas mayores a 900 píxeles e imagen sin las áreas de no interés.

### C) Operadores morfológicos.

Para el tratamiento de imágenes binarias la operación morfológica, morfología o matemática morfológica se utiliza como una técnica de procesado que depende de las formas de las imágenes. El valor de cada píxel en la imagen de salida se resulta de la comparación del píxel correspondiente de la imagen de entrada con sus píxeles adyacentes. Estas transformaciones permiten, dada una imagen inicial, reducir ruidos, realzar la imagen o determinados aspectos de la misma, detectar bordes o la propia segmentación de la imagen en diferentes clases.

Las operaciones morfológicas fundamentales son la dilatación y la erosión, y en ellas se basan otros operadores morfológicos como la apertura y el cierre. En las operaciones morfológicas se aplica un elemento estructurante a la imagen binaria de entrada y genera una imagen binaria de salida. Este elemento estructurante se desplaza sobre la imagen y la transforma, de manera que se crean nuevos píxeles o se eliminan.

Para este trabajo se realizó una dilatación seguida de un cierre porque son las dos técnicas que resultan más útiles para resaltar los caminos forestales y se consigue una continuidad para trazar el eje de los mismos.

- Dilatación

Es una operación morfológica básica que expande las zonas positivas de la imagen, provocando su agrupación si se encuentran cerca de otras regiones positivas y aumentando su área en caso contrario. Los puntos pertenecientes al fondo que se encuentran rodeados de regiones positivas son engullidos por éstas, produciéndose un efecto global de aumento en el brillo de la imagen final con respecto a la inicial. La dilatación de la imagen binaria A por el elemento estructurante B se muestra cómo  $(A \oplus B)$  y se define como:

$$A \oplus B = \{x | (\hat{B})_x \cap A \neq \emptyset\}$$

Se obtiene en base a la reflexión del elemento estructurante  $(\hat{B})$  con respecto a su origen y un desplazamiento x de B sobre el conjunto A.

La salida de la dilatación es el conjunto de puntos barridos por el centro del elemento estructurante mientras algún punto de B coincide con algún punto de A. Se puede interpretar también la dilatación como el resultado de reemplazar cada píxel blanco de la imagen original por una réplica del elemento estructurante (Figura 23).

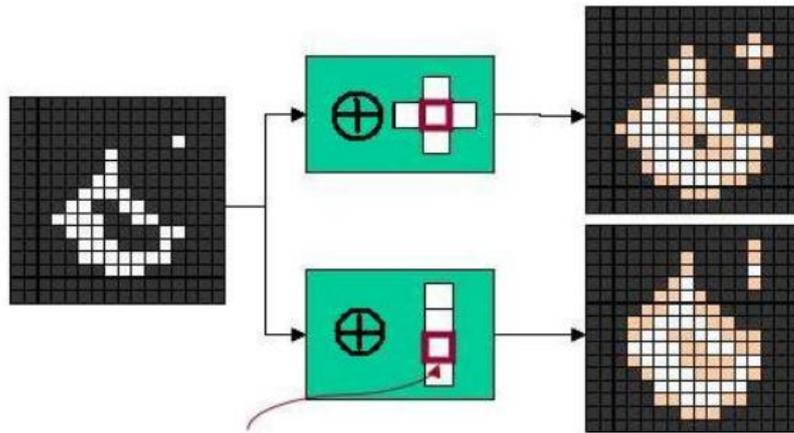


Figura 23: Funcionamiento de la dilatación.

En la figura 23, los píxeles que cambian de negro a blanco por efecto de la dilatación aparecen en tono marrón claro.

- Erosión

La erosión es la otra operación morfológica básica. Al contrario que en la dilatación su efecto es contraer las regiones positivas a favor de las regiones del fondo, disminuyendo el brillo de la imagen de salida con respecto a la entrada. Las regiones de píxeles positivos rodeadas de fondo encogerán, y los píxeles positivos aislados desaparecerán, además de que las regiones de fondo que estuvieran cerca se fusionarán mientras los píxeles aislados de fondo aumentarán. La erosión de la imagen binaria A por el elemento estructurante B se muestra cómo  $(A \ominus B)$  y se define como:

$$A \ominus B = \{x \mid B_x \subseteq A\}$$

Se obtiene en base al desplazamiento del elemento estructurante  $B_x$ , de manera que se cumple que este está completamente incluido dentro del conjunto A.

La salida de la erosión es el conjunto de puntos barridos por el centro del elemento estructurante mientras se cumpla que todos los puntos de B estaban contenidos en A. La erosión se encarga de eliminar los grupos de píxeles donde el elemento estructurante no entra.

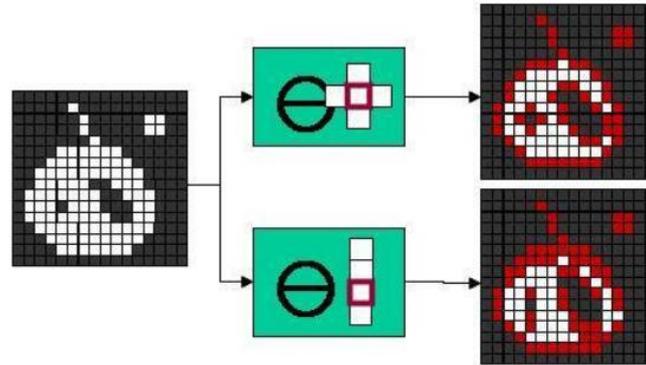


Figura 24: Funcionamiento de la erosión.

En la figura 24 los píxeles que cambian de blanco a negro por el efecto de la erosión aparecen en rojo.

- Cierre

El cierre es la otra operación morfológica, basada en las operaciones fundamentales, es una dilatación seguida de erosión. Se usa para rellenar los huecos en el interior de las estructuras (dilatación) y permite unir elementos cercanos. Así conseguimos unir las discontinuidades de las pistas forestales detectadas, debido a que pequeños píxeles aislados que pertenecen a los caminos forestales al ser ampliados en la dilatación no pueden ser eliminados en la erosión. Por tanto, logramos restaurar y unir los caminos anteriormente detectados. El cierre de la imagen A por el elemento estructurante K se muestra cómo  $(A \bullet K)$  y se define como:

$$A \bullet K = (A \oplus K) \ominus K$$

Donde  $(A \oplus K)$  es la operación de dilatación y  $\ominus K$  es la erosión con el mismo elemento estructurante. Se desplaza el elemento estructurando  $K$  por la frontera del conjunto de manera que redondea concavidades grandes (Figura 25).

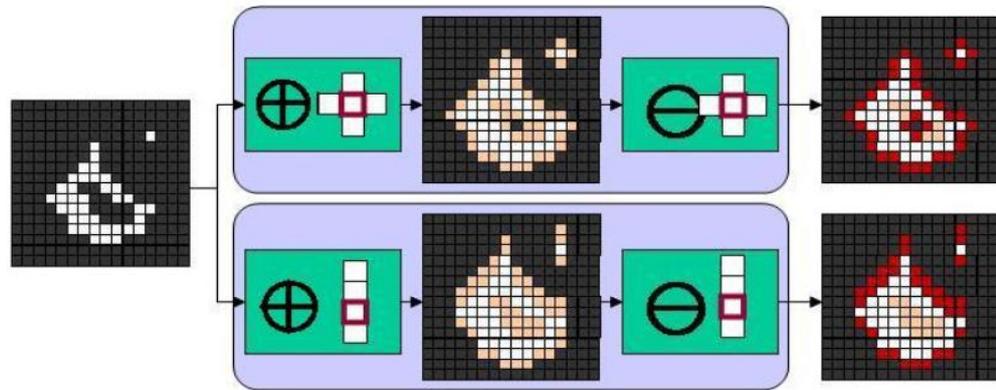


Figura 25: Funcionamiento del cierre.

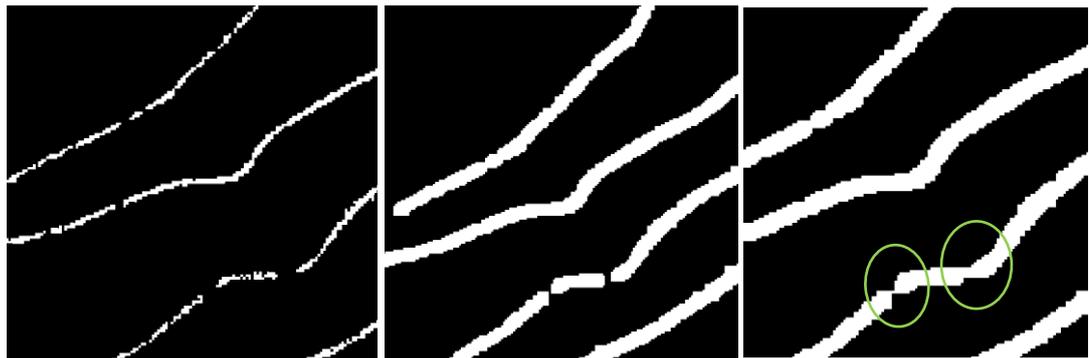


Figura 26: Imagen binaria intensidad entrante, operación de dilatación y operación de cierre.

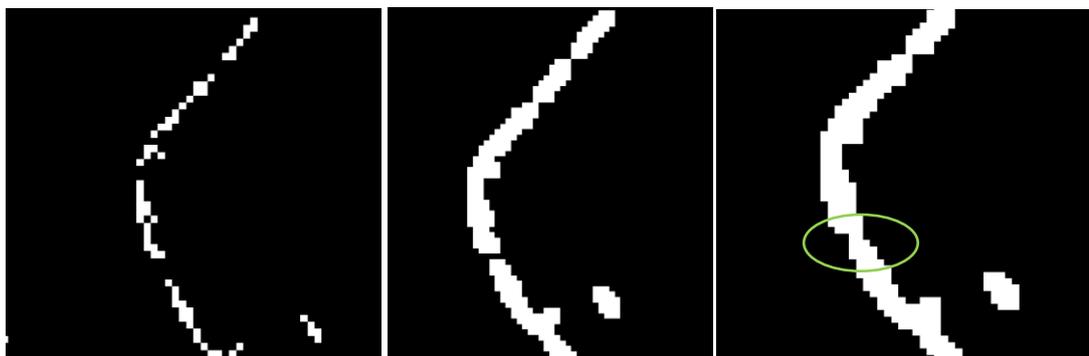


Figura 27: Imagen binaria pendientes entrante, operación de dilatación y operación de cierre.

En las figuras 26 y 27 se puede observar como la operación de dilatación expande o rellena las discontinuidades en los píxeles y provoca su agrupación. Y con la

operación de cierre, dilatación seguida de erosión, se consigue cerrar los huecos o concavidades que no pueden contener al elemento estructurante.

En cada una de las imágenes que se ha aplicado la metodología, se obtienen pistas diferentes y puesto que son imágenes binarias, se combinan los resultados obtenidos por separado mediante una unión, en un único resultado. De esta manera, se extrae una imagen binaria final con la detección de pistas a partir de nube de puntos LiDAR.

- **Esqueletonización.**

La esqueletonización es un proceso de quitar de un patrón, caminos forestales en nuestro caso, la mayor cantidad de píxeles posibles sin afectar la forma general del patrón, es decir, después de que los píxeles hayan sido quitados, el patrón debe ser reconocido.

Una vez se obtiene la imagen final de los caminos forestales con el máximo ruido posible eliminado se procede a esqueletizar los mismos, de manera que permite extraer el esqueleto de un objeto en una imagen eliminando los píxeles contiguos a objetos, pero sin romper la continuidad de éstos (Figura 28). Este comando extrae los caminos forestales con una anchura de 1 píxel.

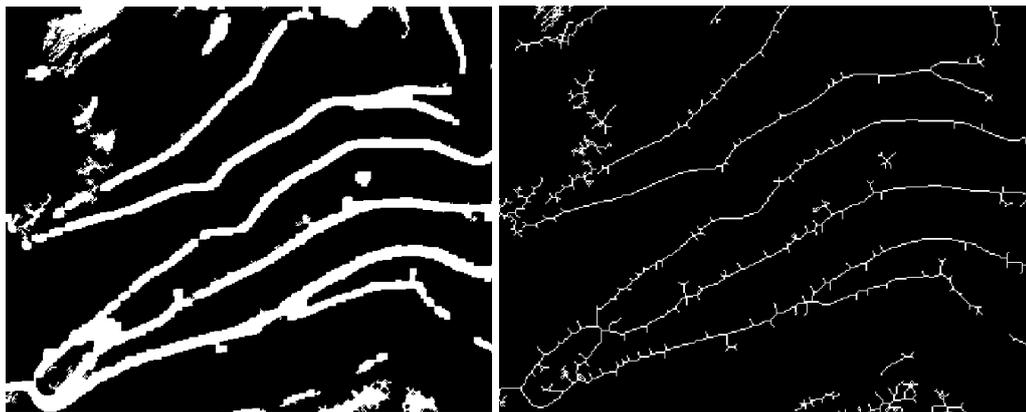


Figura 28: Resultado de la operación de esqueletonización.

Como se puede ver en la figura 28, la esqueletización produce muchas ramificaciones debidas a las irregularidades en los contornos de los caminos. Estas ramificaciones no aportan información en las pistas y dificultan su identificación, por ello es necesario un proceso de poda o eliminación de ramas. Para ser suprimidas, se

obtienen los puntos de intersección y los puntos donde terminan las ramas y se calcula la distancia entre ellos, comenzando a medir desde el extremo o punto final y encontrando todos los píxeles que están en el punto de ramificación más cercano, para por último eliminar esos píxeles del esqueleto original (Figura29).

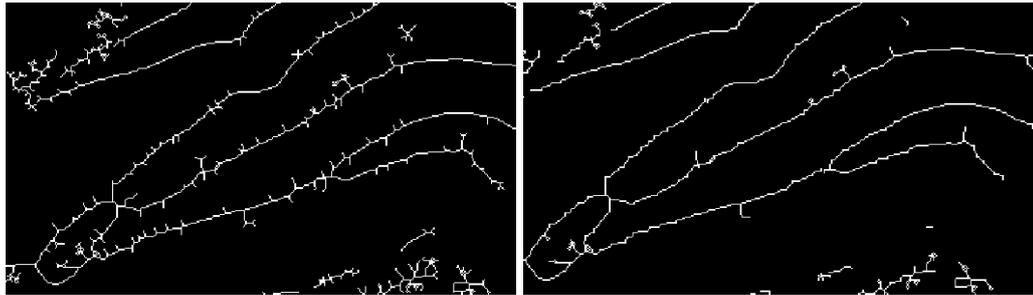


Figura 29: Resultado de la eliminación de las ramificaciones que se producen después de la esqueletonización.

La imagen binaria final con el esqueleto de las vías forestales detectadas (Figura 29) se va a utilizar como imagen de entrada para el algoritmo de restauración Tensor-Voting, dado que este algoritmo ofrece mejores resultados con un adelgazamiento de la estructura a analizar.

### 3.4.3 Restauración de la información basado en Tensor-Voting.

Durante el proceso de extracción de los caminos forestales se puede advertir la pérdida de muchas partes de los mismos. Este hecho puede estar originado por diferentes causas, como: que los caminos estén cubiertos por vegetación densa, que no se encuentre suficiente contraste de color con las áreas de alrededor, o que, al ser segmentados, el tramo que se detecta tenga un número de píxeles menor al inicial, con la consiguiente eliminación no deseada de parte de dichos tramos.

Los algoritmos Tensor Voting (TV) buscan capturar principios de la percepción visual humana para revelar información sobre estructuras perceptivamente coherentes de manera robusta, aún en contextos altamente dañados o ruidosos (Medioni et al. 2004). Es una poderosa herramienta de procesamiento de imágenes que ayuda a limpiar, rellenar huecos y reforzar curvas basándose en los principios de

Gestalt de cierre y continuidad (Trevor Forbes Linton, 2009). El sistema TV toma una imagen binaria y genera un mapa de unión y otro de curvas (Figura 31). El mapa de curvas ofrece la posibilidad de que una curva pase por cada píxel. Del mismo modo, el mapa de unión expresa la probabilidad de una unión en un píxel. Los mapas de unión y curva se valoran de 0 a 1.

En el caso particular de la detección de pistas forestales cada píxel clasificado como camino da lugar a una estimación de una curva a través de ese píxel. Esta curva se utiliza para emitir votos en otros píxeles donde deberían pasar las curvas. Por lo que cada píxel está asociado con un tensor que codifica la orientación del píxel o la orientación más probable de una curva en ese píxel.

El tensor se utiliza para representar la dirección y la significancia de los píxeles en la imagen, se define como (ec.3):

$$T = \lambda_1 \hat{e}_1 \hat{e}_1^T + \lambda_2 \hat{e}_2 \hat{e}_2^T = (\lambda_1 - \lambda_2) \hat{e}_1 \hat{e}_1^T + \lambda_2 (\lambda_1 \hat{e}_1 \hat{e}_1^T + \lambda_2 \hat{e}_2 \hat{e}_2^T) \quad (\text{ec.3})$$

Donde  $\lambda_1$  y  $\lambda_2$  son los valores propios del tensor y  $\hat{e}_1$  y  $\hat{e}_2$  son los vectores propios del tensor. Los tipos de estructura de los tensores se definen como  $\hat{e}_1 \hat{e}_1^T$  para un *stick tensor* (tensor de palo o de curva) el cual tiene una orientación perfectamente definida que representa las características de la curva y cuyos valores propios cumplen  $\lambda_1 > \lambda_2$  y como  $\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T$  para un *ball tensor* (tensor de bola o de unión), que no tiene una orientación definida, que representa las características del nodo o puntos de intersección y cuyos valores propios cumplen  $\lambda_1 = \lambda_2$  (Figura 30). La expresión  $\lambda_1 - \lambda_2$  indica el tamaño del tensor, y por tanto la prominencia. La saliencia o prominencia es la importancia percibida de las estructuras (curvas y uniones) y se mide por la magnitud  $\lambda_1$ .

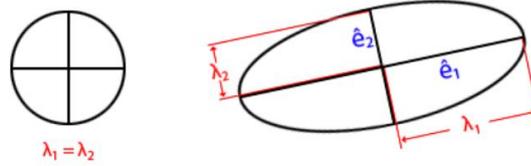


Figura 30: Visualización del tensor en 2 dimensiones. La representación de un tensor de unión (bola) y la representación de un tensor de curva (barra).

Posterior a la generación de los tensores, se procede a emitir los votos, de manera que cada píxel entrega su información a otros píxeles vecinos. La magnitud de este voto depende de la proximidad y la continuidad. Por tanto, la influencia de un tensor a otro debe decaer a medida que la distancia entre ellos aumenta. Por todo ello, la función de prominencia entre dos tensores que comparten la misma normal sobre un círculo cuya circunferencia atraviesa a los dos, se define como (ec. 4):

$$fp(s, k, \sigma) = e^{-\left(\frac{s^2 + ck^2}{\sigma^2}\right)} \quad (\text{ec. 4})$$

Donde  $s$  es la longitud del arco entre dos tensores,  $k$  es la curvatura,  $c$  es el grado de decadencia o degradación de la curva y  $\sigma$  es la escala de la votación, la cual determina el tamaño del vecindario (campo de votación). El parámetro  $c$  (ec. 5) se calcula en función de  $\sigma$  y se define de la siguiente manera:

$$c = \frac{-16 \cdot \log(0.1) \times (\sigma - 1)}{\pi^2} \quad (\text{ec. 5})$$

Después de la votación del tensor, cada píxel recopila todos los votos proyectados por el tensor en su vecindario y los integra en un nuevo tensor. Finalmente, el tensor se descompone (ec. 3) y se calcula el valor propio. Se obtiene la prominencia de cada puntos en la imagen, de manera que las características estructurales de la imagen se realizan calculando su prominencia. Si la prominencia de un píxel es  $(\lambda_1 - \lambda_2) > \lambda_2$ , el píxel es un punto en la curva. Si  $\lambda_1 = \lambda_2 > 0$ , el píxel es una unión.

Los votos son recolectados por una operación simple de escaneo de la imagen y la magnitud de cada voto es multiplicada por su correspondiente valor de prominencia. Esto genera un mapa de probabilidad mostrando las curvas y líneas que completarían la continuidad de los puntos en la imagen (Figura 31).

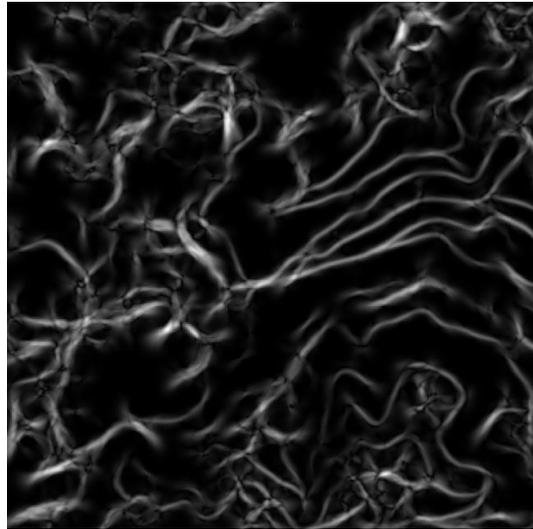


Figura 31: Mapa de saliencia o prominencia.

Finalmente, el algoritmo devuelve la imagen final (Figura 32), en la que se forman las líneas que conectan partes del esqueleto buscando la mejor ruta posible de continuidad.

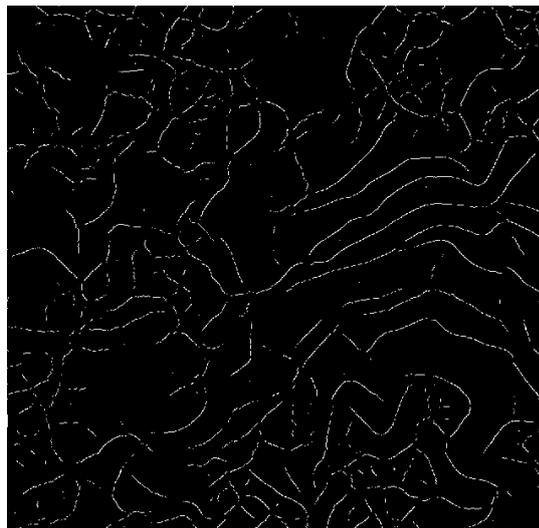


Figura 32: Imagen final algoritmo TV.

#### 3.4.4 Vectorización de las pistas forestales detectadas.

Con el objetivo de evaluar la capacidad de la metodología propuesta para detectar pistas forestales, se compararon los obtenidos en el área de estudio con los que resultan de realizar una digitalización aproximada de los caminos reales existentes en la zona de estudio.

La imagen final resultante de aplicar el algoritmo TV a la propuesta inicial de pistas forestales se vectorizó para obtener un mapa vectorial con las pistas forestales constituido por polilíneas.

Para ello se utilizó la vectorización automática con la herramienta *ArcScan* de ArcGis 10.3. La vectorización automática es el procedimiento que convierte los datos ráster en entidades vectoriales sin intervención del usuario, siempre que el ráster se encuentre clasificado en dos niveles (imagen binaria).

Durante este proceso, se vectorizan solo los caminos pertenecientes a la zona de montes y se limpian las zonas de población y carreteras nacionales, que se han detectado debido a que comparten características de color, intensidad y pendientes y que, por tanto, no son objeto de interés en el trabajo y no se han podido eliminar durante la detección automática.

## 4. RESULTADOS

### 4.1. Detección de pistas forestales

De los resultados obtenidos durante la detección automática, se puede determinar que de las tres capas que se obtienen para realizar la extracción de pistas forestales, la imagen de intensidad y el mapa de pendientes ofrecen mejores resultados con las técnicas de umbralización, pero a la vez producen más ruido (Figura 33), debido a que las zonas donde se encuentran los caminos comparten características de intensidad y pendiente con zonas de población que no son objeto del trabajo. Además, con las técnicas para suavizar la imagen se produce la pérdida de parte de la información, y la consecuente discontinuidad de los caminos forestales (Figura 34).

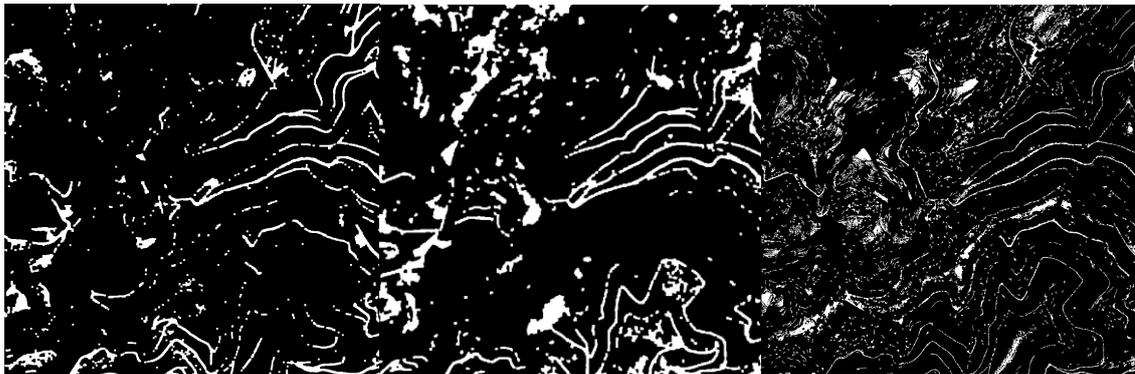


Figura 33: Resultados de la detección de caminos forestales mediante Matlab.



Figura 34: Caminos no detectados, probablemente por la reducción de ruido en imágenes.

Se utilizan operadores morfológicos de dilatación y cierre, con el propósito de rellenar las discontinuidades que se obtienen. De esta manera, se consigue la unión de ciertos tramos en los caminos forestales, pero esto conlleva que también se dilatan o expanden las zonas pertenecientes a la población, que no son objeto de estudio. Para eliminar estas áreas de población presentes en la imagen, se utilizan operadores de cálculo de los componentes conectados, a partir de un determinado criterio de área. Y en su posterior vectorización sobre la cartografía base, se completa su eliminación (limpieza), de manera que se realiza una post clasificación de las pistas entre los objetos que no son de interés (Figura 35).

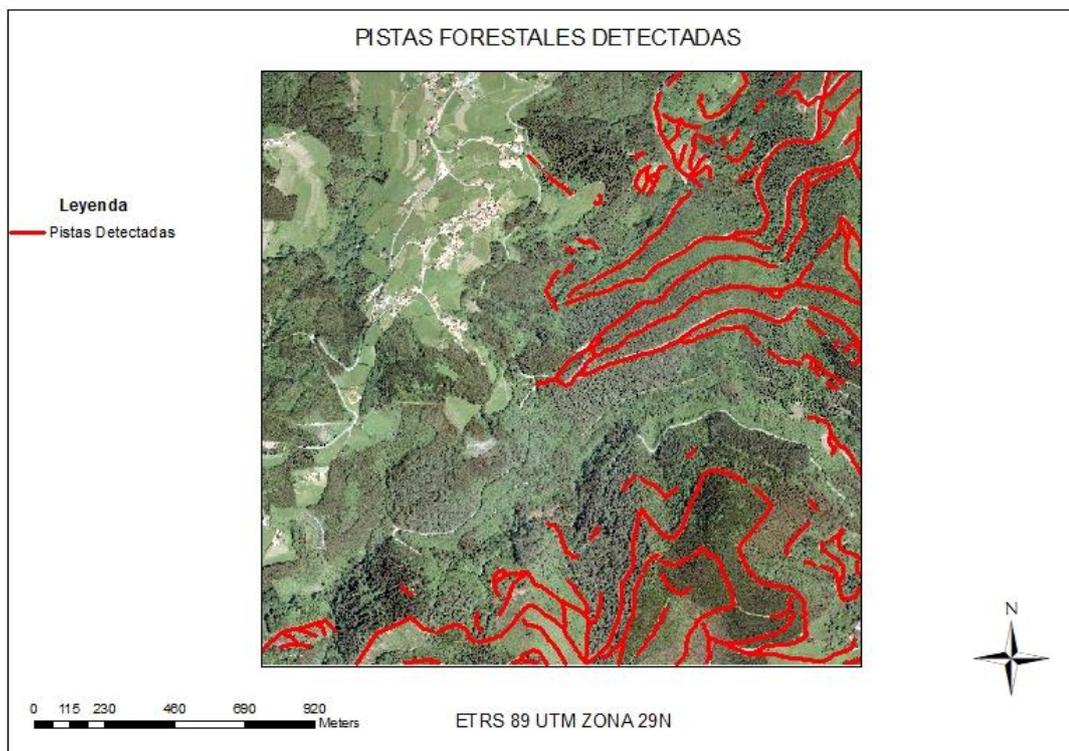


Figura 35: Visualización de las pistas forestales detectadas.

Por otra parte, se pueden observar zonas donde visualmente no se perciben los caminos, a causa de la densidad de la vegetación, pero si se detectan durante la metodología seguida (Figura 36). En estos casos se puede afirmar que, aun cuando el pulso láser se puede ver influenciado por la dificultad de atravesar la vegetación, llega a proporcionar cierta información del suelo.



Figura 36: Zonas donde los caminos se encuentran ocluidos por la vegetación (izquierda). Detección de caminos en la misma zona (derecha).

#### 4.2. Unión de tramos inconexos mediante Tensor Voting

La zona de estudio está compuesta mayoritariamente por frondosas, por lo que la detección de carreteras forestales a partir de una nube de puntos LiDAR se obtiene de manera discontinua. Teniendo en cuenta, además que, durante el proceso de detección, se realizan procesos de umbralización, filtrado de imágenes u operaciones morfológicas, con el fin de resaltar la característica de estudio, se producen pérdidas de la información, sobre todo en píxeles aislados.

La aplicación del algoritmo TV a la imagen resultante en la detección de pistas, se lograron resultados poco satisfactorios (Figura 37), pues se consigue unir ciertos tramos que tienen grandes discontinuidades, sin embargo, en otros produce discontinuidades.

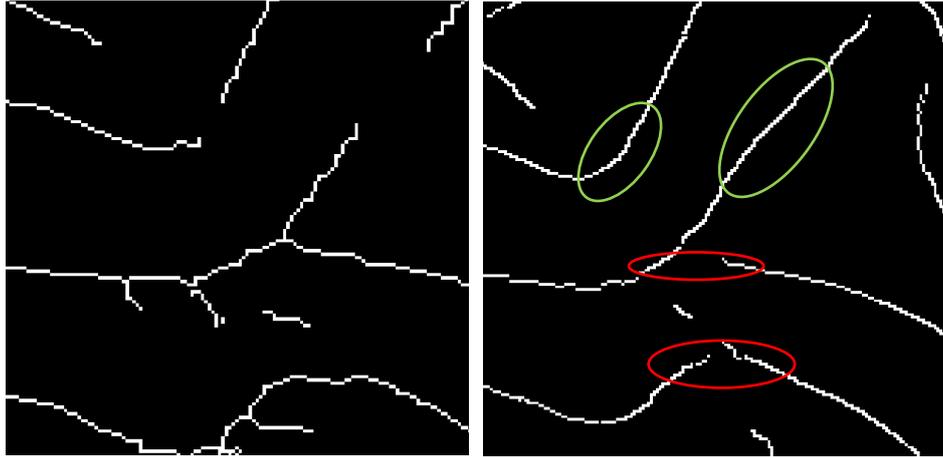


Figura 37: Resultados sobre las discontinuidades con el algoritmo TV.

Los tramos que se unieron tras la aplicación del algoritmo TV se combinaron a los resultados de la detección sin la aplicación del mismo, logrando mayor continuidad en algunos tramos de pistas forestales, como se puede ver en la figura 38.

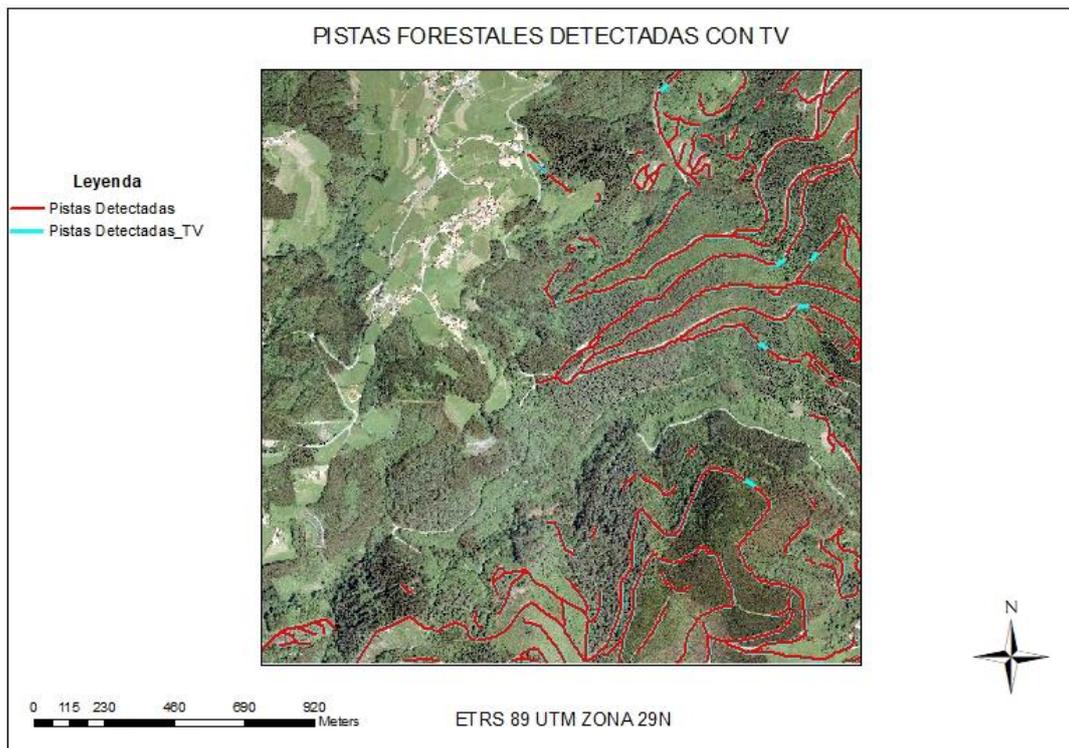


Figura 38: Pistas forestales detectadas (rojo) y tramos de conexión (azul) tras la aplicación del algoritmo TV.

#### 4.3. Evaluación en la detección de las pistas forestales

Para evaluar la eficiencia de los resultados obtenidos de manera cuantitativa, se realiza una digitalización aproximada de las pistas forestales reales, con ayuda de la imagen RGB generada y del ráster en formato imagen del Mapa Topográfico Nacional (MTN25) de la zona de estudio, en el que se representan algunos de los caminos forestales (Figura 39).

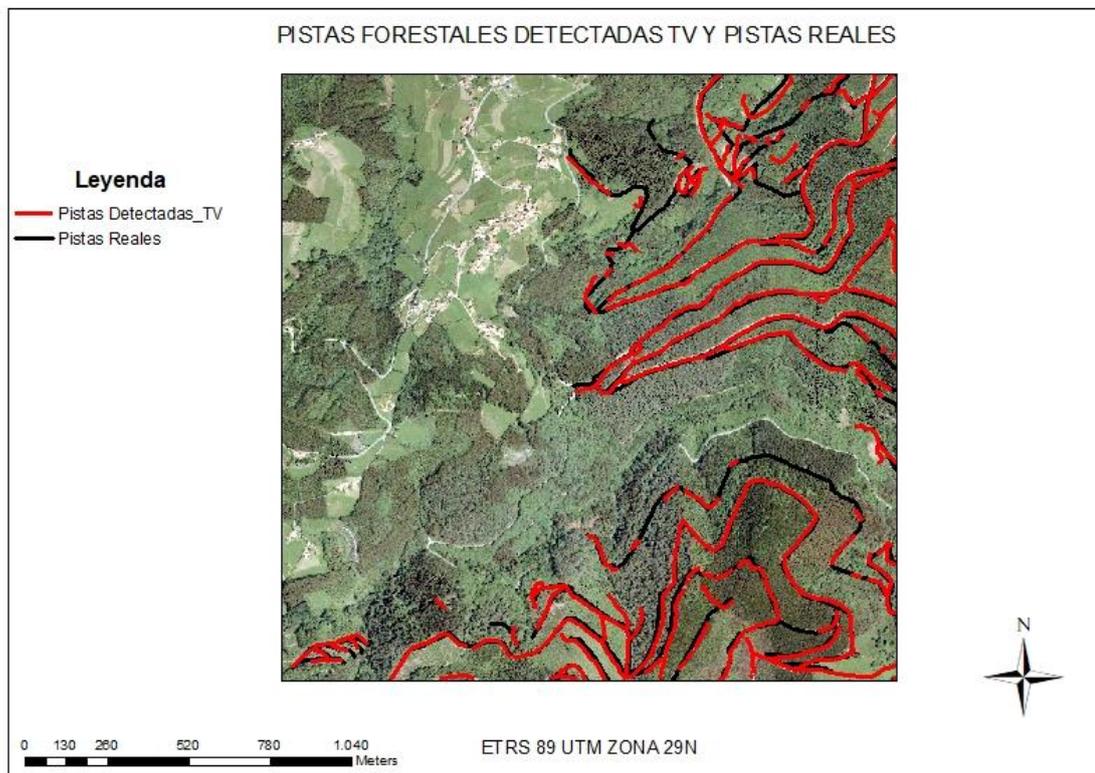


Figura 39: Visualización de las pistas forestales detectadas y las pistas reales digitalizadas.

Para comparar ambos mapas vectoriales, se aplicaron los métodos de evaluación descritos por diversos autores Heipke et al. (1997) y Wiedmann (2003). Para ello se genera un buffer de las líneas digitalizadas a partir de la imagen RGB, que será la referencia, en este caso se ha optado por un ancho de buffer de 10 m (área de influencia), considerando el ancho mínimo de las pistas es de 4 m, y contando con errores propios de la georreferenciación, vectorización y tamaño de píxel. Superponiendo el mapa de líneas con las pistas forestales detectadas, al mapa de las pistas digitalizadas con los buffer se calculan los siguientes parámetros (Figura 40):

- Verdaderos Positivos, (en inglés TP), aquellos segmentos de líneas detectadas que se ubican dentro del área de influencia, considerándose como pistas detectadas.
- Falsos Positivos (FP), aquellos segmentos detectados como pistas que quedan dentro del área de influencia, pero solo en cierta parte.
- Falsos Negativos (FN), aquellos segmentos de la red real que no se identificaron como pistas detectadas.

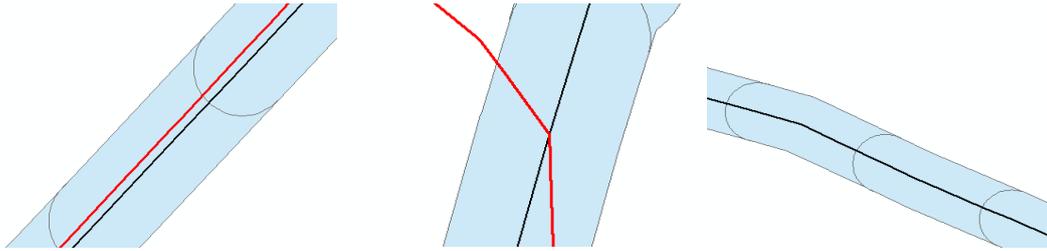


Figura 40: Criterios de valoración en la detección de pistas. El primero caso considerado con TP, el segundo ejemplo de FP y el último ejemplo de FN.

A partir de los criterios impuestos sobre el área de influencia y mediante la selección por localización, se calculan los siguientes indicadores de Integridad (Completeness, COM), que describe la longitud de pistas que se detectaron de la red real, mediante la relación entre la longitud de los datos coincidentes y la longitud de los datos de referencia, y Corrección o Precisión (Correctness, COR), que describe la longitud de pistas predichos que son verdaderos, mediante la relación entre las longitudes coincidentes del área de influencia y la longitud total detectada, reflejadas en la ecuación 6.

$$\text{COM} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{COR} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (\text{ec. 6})$$

A partir de los indicadores de Integridad y Corrección se puede obtener otro indicador, la Calidad (Quality, Q) que equilibra los resultados de los dos anteriores como se refleja en la ecuación 7.

$$Q = \frac{\text{TP}}{\text{TP} + \text{FN} + \text{FP}} \quad (\text{ec. 7})$$

Los resultados recogidos en la siguiente tabla obtenidos en la detección de pistas sin aplicar el algoritmo TV, arrojaron unos resultados de Integridad, Corrección y Calidad:

<b>EVALUACIÓN DE LA DETECCIÓN DE CAMINOS FORESTALES</b>		
Integridad (COM)	Corrección o Precisión (COR)	Calidad (Q)
0,79	0,77	0,64

Los resultados indican que de la longitud total de pistas existentes se detectan un 79% (COM) y de la longitud total de pistas detectadas un 77% (COR) son realmente pistas forestales. El indicador de calidad nos indica que del total de la longitud de pistas existente más las detectadas como tales que no los son, un 64% son realmente pistas forestales.

Los resultados recogidos en la siguiente tabla obtenidos con la aplicación del algoritmo TV, arrojaron los siguientes resultados de Integridad, Corrección y Calidad:

<b>EVALUACIÓN DE LA DETECCIÓN DE CAMINOS FORESTALES CON LA APLICACIÓN ALGORITMO TV</b>		
Integridad (COM)	Corrección o Precisión (COR)	Calidad (Q)
0,81	0,77	0,66

Los resultados indican que de la longitud total de pistas existentes se detectan un 81% (COM) y de la longitud total de pistas detectadas un 77% (COR) son realmente pistas forestales. El algoritmo TV, restaura la discontinuidades que se producen durante la detección automática, de manera que aumenta la longitud de verdaderos positivos y disminuye la longitud de falsos negativos, manteniéndose constante la longitud de pistas detectadas como falsos positivos.

El porcentaje de pistas no detectadas corresponde a pistas ocluidas por la densa vegetación de frondosas, la cual dificulta la penetración del pulso láser a través de ellas.

Y del porcentaje de pistas detectadas, algunas son errores de las zonas donde la vegetación es menos densa (replado) o donde existen áreas de suelo (Figura 41).



Figura 41: Caminos no detectados en zonas de vegetación frondosa, y zonas de replado donde se produce la detección.

## 5. CONCLUSIONES

En este trabajo se ha propuesto una metodología para detectar pistas forestales utilizando datos LiDAR. De los resultados obtenidos en las zonas de monte donde la vegetación está compuesta por frondosas como en nuestro caso el eucalipto, los pulsos láser de la tecnología LiDAR tienen la dificultad para llegar al suelo, por lo que supone un inconveniente en la detección de los caminos forestales. Esto origina que tanto el MDT, como la imagen RGB, la imagen intensidad y el mapa de pendientes generados presenten errores elevados debido a la falta de puntos de suelo. En zonas donde la vegetación es poco densa y se puede hacer una detección más efectiva de los caminos forestales; sin embargo, también se detectan como pistas zonas de repoblado de árboles y zonas de población próximas a la zona de estudio porque presentan valores de intensidad y pendiente semejantes a los caminos forestales.

Durante la realización de este trabajo se ha propuesto utilizar el algoritmo Tensor-Voting para procurar resolver la problemática de la detección en zonas donde la vegetación es muy densa. Esta metodología mejora ligeramente los resultados en cuanto a longitud de pistas detectadas del total de las pistas existentes, pero no mejora donde se han detectado pistas y realmente no existen (repoblado) porque los valores de intensidad son parecidos a los valores de los caminos.

Los resultados obtenidos en el presente trabajo, a partir de la metodología son similares a los obtenidos con otros estudios en zonas forestales (Ferraz et al. 2016, Kiss et al. 2015), a pesar de que los datos LiDAR disponibles del PNOA (1ª Cobertura) tienen una densidad de 1- 0,5 puntos/m<sup>2</sup>, inferiores a la de dichos autores.

En resumen, las nubes de puntos LiDAR de baja densidad y con vegetación frondosa se pueden utilizar como método para obtener una buena aproximación de las pistas forestales, que debe completarse con trabajo de campo en las zonas con gran densidad de arbolado en las que no existe penetración de los pulsos del escáner.

Como futuras líneas de investigación, basadas en el mismo objetivo, se propone adaptar el algoritmo Tensor Voting a las características de los datos LiDAR en pistas forestales.

## 6. BIBLIOGRAFÍA

- ALBACETE, A. S. *Procesamiento de datos LiDAR con ArcGIS Desktop 10*. Garmendia Salvador, Luis (dir), Mauro Gutiérrez, Francisco(codir). Trabajo Fin de Máster. Universidad Complutense de Madrid. Madrid, 2011.
- ALEGRE, E., SÁNCHEZ, L., FERNÁNDEZ, R. A., MOSTAZA, J. C. *Procesamiento Digital de Imágenes. Fundamentos y Prácticas con Matlab*. Universidad de León, Secretariado de Publicaciones y Medios Audiovisuales, 2003, ISBN 84-9773-052-6.
- ASPRS. The American Society for Photogrammetry & Remote Sensing, *LAS Specification Version 1.4-R13*, 2013, pp.1–28. <https://doi.org/faf> (Accedido: febrero 2019).
- AZIZI, Z., NAJAFI, A., SADEGHIAN, S. Forest Road Detection Using LiDAR Data. *Journal of Forestry Research*, 2014, 25 (4): 975–980.
- BECKER, G. 1997. *Integrated information systems as a basis of advanced harvesting planning*. Voluntary paper prepared for the XI World Forestry Congress, Antalya, Turkey, 13-22 Oct 1997.
- BUFTON, J. L. Laser altimetry measurements from aircraft and spacecraft. *Proceedings of the IEEE*, 1989, 77, pp. 463-477.
- CABO, C., ORDOÑEZ, C., GARCÍA-CORTÉS, S., MARTÍNEZ, J. An algorithm for automatic detection of pole-like Street furniture objects from Mobile Laser Scanner point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2017, 87, pp. 47-56.
- CABO GÓMEZ, C. Algoritmos para la detección automática de elementos en entornos viales a partir de datos de escáneres láser móvil. Ordóñez Galán, Celestino (dir) y García Cortés, Silverio (dir). Tesis Doctoral. Universidad de Oviedo. Departamento de Minería, Obra Civil, Medio Ambiente y Dirección de Proyectos. Febrero, 2017.
- CHASMER, L., HOPKINSON, C., TREIZ, P. Investigating laser pulse penetration through a conifer canopy by integrating airborne and terrestrial LiDAR. *Canadian Journal of Remote Sensing*, 2006, 32, pp. 116-125.

- ELSEBERG, J., BORRMANN, D., NÜCHTER, A. One billion point in the cloud-an octree for efficient processing of 3D laser scans. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2013, 76, pp. 76-88.
- FERNÁNDEZ, J., SIABATO, W. *Procesamiento y accesibilidad de datos LiDAR a través de aplicaciones distribuidas. Integración del estándar OpenGIS Web Processing Service y la Arquitectura GRID Computing*. Universidad Politécnica de Madrid, 2011, pp. 1-13. <https://es.scribd.com/document/61073185/tesis1>. (Accedido: febrero 2019).
- FERNÁNDEZ-RIVAS, J., SIABATO, W. *Arquitectura GRID Computing como medio para la democratización e integración de datos LiDAR*. ETSI Topografía, Geodesia y Cartografía, Universidad Politécnica de Madrid, 2010 pp. 1-14.
- FERRAZ, A., MALLENT, C., CHEHATA, N. Large-scale road detection in forested mountainous areas using airborne topographic lidar data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2016, 112, pp. 23-36.
- FLOOD, M., GUTELIUS, B. Commercial Implications of Topographic Terrain Mapping Using Scanning Airborne Laser Radar. *Photogrammetric Engineering and Remote Sensing*, 1997, 63, pp. 327-332.
- GALLAY, M. Section 2.1.4: *Direct Acquisition of Data: Airborne laser scanning*. In: Clarke, L.E & Nield, J. M. (Eds) *Geomorphological Techniques* (Online Edition). British Society for Geomorphology. 2013, ISSN 2047-0371.
- GONZALEZ, R., WOODS, R., EDDINS, S. *Digital Image Processing using Matlab*. Prentice-Hall, Inc, 2003, ISBN 0130085197.
- HEIPKE, C., MAYER, C., WIDEMANN, C. Evaluation of Automatic Road Extraction. *Chair for Photogrammetry and Remote Sensing*. 1977, pp. 1-11.
- HINKS, T., CARR, H., TRUONG-HONG, L., LAEFER, D.F. Point cloud data conversion into solid models via point-based voxelization. *Journal of Surveying Engineering*. January 2012, Belfield, Dublin 4, Ireland ,139(2), pp. 72-83

- HYYPÄ, J., HYYPÄ, H., LECKIE, D., GOUGEON, F., MALTAMO, M. Review of methods of small-footprint airborne laser scanning for extracting forest inventory data in boreal forest. *International Journal of Remote Sensing*, 2008, 29, pp. 1339-1366.
- KISS, K., MALINEN, J., TOKOLA, T. Forest road quality control using ALS data. *Canadian Journal of Forest Research*. 2015, 45(11), pp. 1636-1642.
- LASCHI, A., NERI, F., BREANCHETTI, N., MARCHI, E. A Methodological Approach Exploiting Modern Techniques for Forest Road Network Planning. *Croat. J. for. Eng*, 2016, 37 (2), pp. 319-330.
- LEFSKY, M. A., COHEN, W. B., PARKER, G., HARDING, D. Lidar Remote Sensing for Ecosystem Studies. *BioScience*, 2002, 52 (1), pp. 19-30.
- MABOUD, M., AMINI, J., HAHN, M., SAATI, M. Road Network Extraction from VHR Satellite Images Using Context Aware Object Feature Integration and Tensor Voting. *Remote Sens*, 2016, 8 (8), <https://doi.org/10.3390/rs8080637> (Accedido: mayo 2019).
- MALTAMO, M., NAESSET, E., VAUHKONEN, J. *Forestry Applications of Airborne Laser Scanning: Concept and Case Studies*. Managing Forest Ecosystems 27. Springer Dordrecht. 2014, ISBN: 978-94-017-8662-1.
- MARTÍNEZ BLANCO, M. *Evaluación y propuesta de metodologías de clasificación a partir del procesamiento combinado de datos LiDAR e imágenes aéreas georreferenciadas*. Sánchez Espeso, Javier (dir) y Bastarrika Izagirre, Aitor (dir). Tesis doctoral. Universidad de Cantabria. Área de Ingeniería Cartográfica, Geodesia y Fotogrametría, enero 2016.
- MEAGHER, D. Geometric modeling using octree encoding. Computer graphics and image processing. *Computer Graphics and Image Processing*, 1982, 19, pp. 129-147.
- MEDIONI, G., KANG, S. *Emerging Topics in Computer Vision*, 2004, pp. 203-219. <http://people.inf.ethz.ch/pomarc/pubs/KangMedioniBook.pdf> (Accedido mayo 2019).
- MEDIONI, G., LEE, M., TANG, C. Computational Framework for Segmentation and Grouping. *Elsevier Science Inc.*, 2000.

- MILLS, J., BARBER, D. *An Addendum to the Metric Survey Specifications for English Heritage- the collection and archiving of point cloud data obtained by terrestrial laser scanning or other methods*. 2003, Versión 11/12/2003. <http://www.english-heritage.org.hk/server/show/nav.9236> (Accedido: marzo 2019).
- MILLS, J., BARBER, D., SMITH-VOYSEY, S. Geometric validation of a ground-based mobile laser scanning system. *ISPRS Journal of Photogrammetry and Remote Sensing*. 2008, 63 (1), pp. 128-141.
- MOCEDO CRUZ, A. *Técnicas de clasificación automática de uso de suelos agrícolas y forestales basadas en imágenes digitales*. Santos Peñas, Matilde (dir) y Pajares Martinsanz, Gonzalo (dir). Tesis doctoral. Universidad Complutense de Madrid. Departamento de Arquitectura de Computadores y Automática, agosto 2012.
- NELSON, R., KRABILL, W., MACLEAN, G. Determining forest canopy characteristics using airborne laser data. *Remote Sensing of Environment*. 1984, 15, pp. 201-212.
- NUCHTER, A., LINGEMANN, K., HERTZBERG, J. Cached kd tree search for ICP algorithms. In 3-D Digital Imaging and Modeling. *3DIM`07. Sixth International Conference on IEEE*. 2007, pp. 416-426.
- Revista de Montes*. Landa, A. F., Rodríguez, F., López, D., Olabarria, J. R. G., Yudego, B. M., Lasala, D., Arjonilla, F. J., Molina, A. G. *Los sensores aerotransportados LiDAR y multiespectrales en la descripción y cuantificación de los recursos forestales*. 2013, 112, pp. 31-36. ISSN 0027-0105.
- ROMERO, A., ALONSO, F., Moreno, J., Belmonte, F., GARCÍA, G. *Evaluación de la producción de escorrentía como agente erosión de depósitos de residuos mineros en la Sierra Minera de Cartagena-La Unión: parte II, Utilización del sensor LiDAR para la generación de capas de información espacial con aplicación en la modelización hidrológica*. Informe científico-técnico. Instituto Euro Mediterráneo del Agua, Murcia, 2009.

- SCHREIER, H., LOUGHEED, J., TUCKER, C., LECKIE, D. Automated measurements of terrain reflection and height variations using airborne infrared laser system. *International Journal of Remote Sensing*. 1985, 6, pp. 101-113.
- SHERBA, J., BLESIOUS, L., DAVIS, J. Object-Based Classification of Abandoned Logging Roads Under Heavy Canopy Using LiDAR. *International Journal of Remote Sensing*. 2014, 6, pp. 4043-4060.
- SINGH, K. K., VOGLER, J. B., MEENTEMEYER, R. K. Estimation of land-use in a urbanized landscape using LiDAR intensity data: a regional scale approach. *A special joint symposium of ISPRS Technical Commission IV & AutoCarto*. 2010, pp. 1-4.
- SOLODUKHIN, V., ZUKOV, A., MAZUGIN, I. Possibilities of laser aerial photography for forest profiling. *Lesnoe Khozyaisto (Forest Management)*. 1977, 10, pp. 53-58.
- SOLOMON, C., BRECKON, T. *Fundamentals of digital image processing. A practical approach with examples in Matlab*. UK: Wiley-Blackwell, 2011. ISBN 978 0 470 84472 4.
- SONG, J-H., HAN, S-H., YU, K., KIM, Y. Assessing The Possibility of Land-cover Classification Using LiDAR Intensity Data. 2006. Disponible en: <https://www.isprs.org/proceedings/XXXIV/part3/papers/paper128.pdf>. (Accedido: enero 2019).
- SONG, M., CIVCO, D. Road Extraction Using SVM and Image Segmentation. *Photogrammetric Engineering & Remote Sensing*. 2004, 70 (12), pp. 1365-1371.
- SUN, K., ZHANG, J., ZHANG, Y. Roads and Intersections Extraction from High-Resolution Remote Sensing Imagery Based on Tensor Voting under Big Data Environment. *Wireless Communications and Mobile Computing*. 2019, 4, pp. 1-11.
- TEJA SURYA, A. Road Extraction From High Resolution Satellite Image. *Indian Institute of Information Technology*. 2017, Sricity, Chittoor, India.
- TREVOR FORBES, L. *Semantic Feature Analysis in Raster Maps*. Berzins, M. (dir). Doctoral thesis. University of the Utah. August, 2009.

- TRIGLAV, M., CROSILLA, F., KOSMATIN M. Theoretical LiDAR point density for topographic mapping in the largest scales. *Geodetski Vestnik*. 2010, 54 (3), pp. 403-416.
- VOSELMAN, G., MAAS, H. G. Airbone and Terrestrial Laser Scanning. *Whittles Publishing*, 2010, Dunbeath, Escocia, 318. pp. 28-35.
- WATT, S., ADAMS, T., GONZÁLEZ, S., ARACIL, A., MARSHALL, H., WATT, P. The influence of LiDAR pulse density and plot size on the accuracy of New Zealand plantation stand volumen equations. *New Zealand Journal of Forestry Science*. 2013, 43, 15.
- WEHR, A., LOHR, U. Airborne Laser Scanning- An Introduction and Overview. *ISPRS Photogrammetry and Remote Sensing*. 1999, 54 (2-3), pp. 68-82.
- WEIDEMANN, C. External Evaluation of Roads Networks. *International Archives of Photogrammetry and Remote Sensing*. 2003, 34 (3/W8), pp. 93-95
- WHITE, R. A., DIETTERICK, B. A., MASTIN, T., STROHMAN, R. Forest Roads Mapped Using LiDAR in Steep Forested Terrain. *Remote Sens*. 2010, 2(4), pp. 1120-1141. Disponible en: <https://doi.org/10.3390/rs2041120> (Accedido: enero 2019).

## ANEXO: CÓDIGO MATLAB

### - Voxelización

```
function [NubeVoxel, idOrdenPixel, idVueltaPix_XYZ]=PixelizarSueloArboles (NubeXYZRecorte, LadoPixelTerreno)

% Nuevo mínimo para restarle a las coordenadas
minNube=min(NubeXYZRecorte); %de cada columna

%Reducción de coordenadas XY a valores mínimos
NubeXYZRecorte(:,1)=NubeXYZRecorte(:,1)-minNube(1);
NubeXYZRecorte(:,2)=NubeXYZRecorte(:,2)-minNube(2);

%Se genera el código píxel ordenando (sort) y redondeando (floor)
%Tratando columnas como vectores con valor coord-min
%Tamaño vóxel=lado píxel= 2m

[CoordCodif, idOrdenPixel]=sort((floor(NubeXYZRecorte(:,2)/LadoPixelTerreno)+1)*10000+...
    floor(NubeXYZRecorte(:,1)/LadoPixelTerreno)+1);

%Se calculan las coordenadas codificadas únicas (se eliminan las repeticiones que se producen cuando más de un punto de la NubeXYZRecorte corresponde al mismo vóxel)

[CoordCodifUnicas, idVueltaPix_XYZ]=unique(CoordCodif, 'last');

%Se deshace la codificación (teniendo en cuenta el orden en el que CoordCodif fue hecho)
NubeVoxel(:,2)=uint32(floor(CoordCodifUnicas/10000));
NubeVoxel(:,1)=uint32(CoordCodifUnicas-
    floor(CoordCodifUnicas/10000)*10000);

end
```

### - Obtención MDT, imagen RGB e imagen intensidad

```
%Datos
LadoPixelTerreno=3;
PercentileMin=0;

%llamada a la función de voxelización
```

```

%Pixeliza la nube de puntos. Es decir: ordena los puntos en
una retícula regular de lado "LadoPixel"
%Ordena los puntos originales según el orden del código de
pixelización
[NubeVoxel,idOrdenPixel,idVueltaPix_XYZ]=PixelizarSueloArbo
les (NubeXYZRecorte,LadoPixelTerreno);
NubeXYZRecorte=NubeXYZRecorte(idOrdenPixel,:);

%Añado un cero al vector para que funcione desde el vóxel 1

UltimosDeCadaPixel=[0;idVueltaPix_XYZ];

% Recorre los píxeles buscando su punto más bajo
for i=1:(size(UltimosDeCadaPixel,1)-1)
    %Teniendo en cuenta que UltimosDeCadaPixel
    contiene el último punto de cada perfil (además de un uno
    al principio, que le añadimos antes)
    %Calculando su punto más bajo (Zmin)

    ZminNubePixel(i)=prctile(NubeXYZRecorte(UltimosDeCadaPixe
l(i)+1:(UltimosDeCadaPixel(i+1)),3),PercentileMin);
    %Aquí se calcula la intensidad máxima de los puntos que
    están dentro de cada píxel. Se puede sacar la mínima o la
    media sustituyendo "max" en la línea siguiente por "min"
    o "mean"

    IntMax(i)=max(NubeXYZRecorte(UltimosDeCadaPixel(i)+1:(Ulti
mosDeCadaPixel(i+1)),7));

    %Se calcula el valor medio para cada canal de color
    RMed(i)=mean(NubeXYZRecorte(UltimosDeCadaPixel(i)+1:(Ulti
mosDeCadaPixel(i+1)),4));

    GMed(i)=mean(NubeXYZRecorte(UltimosDeCadaPixel(i)+1:(Ulti
mosDeCadaPixel(i+1)),5));

    BMed(i)=mean(NubeXYZRecorte(UltimosDeCadaPixel(i)+1:(Ulti
mosDeCadaPixel(i+1)),6));
end

%Figura con la nube de los puntos más bajos de cada celda
(píxel)
figure(1)
axis equal
hold all
plot3(NubeVoxel(:,1),NubeVoxel(:,2),ZminNubePixel,'.')

%Figura con la nube de puntos con las intensidades más
altas para cada píxel
figure(2)

```

```

plot3(NubeVoxel(:,1),NubeVoxel(:,2),IntMax, '.')

% Convierte la lista de coordenadas en una matriz dispersa
PixelesBaseSparse=sparse(double(NubeVoxel(:,1)),double(Nube
Voxel(:,2)),ZminNubePixel);
% Lo mismo para las intensidades
PixelesIntensSparse=sparse(double(NubeVoxel(:,1)),double(Nu
beVoxel(:,2)),IntMax);
%Lo mismo para los colores
PixelesRSparse=sparse(double(NubeVoxel(:,1)),double(NubeVox
el(:,2)),RMed);
PixelesGSparse=sparse(double(NubeVoxel(:,1)),double(NubeVox
el(:,2)),GMed);
PixelesBSparse=sparse(double(NubeVoxel(:,1)),double(NubeVox
el(:,2)),BMed);

%Transforma la matriz dispersa en una matriz normal. Cada
casilla de la matriz resultante corresponde a un píxel, y
almacena la cota de su punto más bajo.
PixelesBase=full(PixelesBaseSparse);
PixelesInt=full(PixelesIntensSparse);
PixelesR=full(PixelesRSparse);
PixelesG=full(PixelesGSparse);
PixelesB=full(PixelesBSparse);

%pasamos canales a tanto por 1 dividiendo entre 255, porque
es matriz tipo double con valores mayores a 1, hay que
escalar para visualizar

Intensidad=PixelesInt/200;
R=PixelesR/255;
G=PixelesG/255;
B=PixelesB/255;

%Concatenación de los canales de color
ImagenColor=cat(3,R,G,B);

%Visualizar imágenes y sus histogramas
figure(3)
imshow(ImagenColor);
figure(4),imhist(ImagenColor);

figure(5),imshow(Intensidad);
figure(6),imhist(Intensidad);
i
%Para guardar
% imwrite(ImagenColor,'ImagenColor.tif')
% imwrite(Intensidad,'Intesidad.tif')

```

- Obtención mapa de pendientes

```
clc
clear all;
close all;
load 'PixelesBase.mat';

%Cálculo de la pendiente:
%el denominador=2xResolucion espacial MDE= 2xLadoPixel=4
%k1=filtro para la pendiente en sentido O-E=Px
k1=[0 0 0;-1/4 0 1/4;0 0 0];
%k2=filtro para la pendiente en sentido S-N=Py
k2=[0 1/4 0;0 0 0;0 -1/4 0];

%Filtra la matriz MDT con el filtro de las pendientes
Px=imfilter(PixelesBase,k1);
Py=imfilter(PixelesBase,k2);

%dimensiones nueva matriz
[m,n]=size(Px);
Px=Px(1:m,1:n);
Py=Py(1:m,1:n);

%pendiente total
P=sqrt((Px.^2)+(Py.^2));
figure(1),imshow(P),title('Mapa de Pendientes');

P1=im2uint8(P);
figure(2),imhist(P1);

%rotar imagen 90 grados
P=imrotate(P,90);
imwrite(P,'Mapa_Pendientes.tif');
```

- Filtrado y operaciones morfológicas: imagen RGB, imagen intensidad y mapa de pendientes

```
clear all;
close all;

%lee las capas de entrada imagen
I1=imread('ImagenColor.tif');
I2=imread('Intensidad.tif');
I3=imread('Mapa_Pendientes.tif');

figure(1),imshow(I1); title('Imagen RGB');
```

```

figure(2),imhist(I1);
figure(3),imshow(I2); title('Intensidad');
figure(4),imhist(I2);
figure(5),imshow(I3); title('Mapa de pendientes');
figure(6),imhist(I3);

%conversión a imagen de grises
J=rgb2gray(I1);
figure(7),imshow(J);
title('grayscale image');

%ajuste de contraste en la imagen de grises
K=imadjust(J,[0.3 0.7],[]);
figure(8),imshow(K);

%Umbralización:
%calcula el percentil del 97,5% de la imagen RGB
%Asigna uno a los valores por encima del percentil
perc = 97.5;
[percent IALTO1] = umbralalto(K,perc);
figure(9),imshow(IALTO1), title('Umbralización Imagen
RGB');

%calcula el percentil del 90% de la imagen Intensidad
%Asigna uno a los valores por encima del percentil
perc = 90;
[percent IALTO2] = umbralalto(I2,perc);
figure(10), imshow(IALTO2), title ('Umbralización Imagen
Intensidad');

%calcula el percentil del 20% de mapa pendientes
%Asigna uno a los valores por debajo del percentil
perc = 20;
[percslope SREC] = umbralbajo(I3,perc);
figure(11), imshow(SREC), title ('Umbralización Imagen
Pendientes');

%Intensidad y pendientes presentan mucho ruido de píxeles
aislados, se eliminan
IALTO2 = bwareaopen(IALTO2,4);
figure(12),imshow(IALTO2);
title('Elimina componentes aislados Imagen
Intensidad(pixel <4)');
SREC = bwareaopen(SREC,4);
figure(13),imshow(SREC);
title('Elimina componentes aislados Mapa Pendientes(pixel
<4)');

%Calcule el área de cada componente en intensidad y
pendientes para eliminar zonas de población:

```

```

CC2 = bwconncomp(IALTO2, 8);
CC3 = bwconncomp(SREC, 8);

a2 = regionprops(CC2, 'Area');
a3 = regionprops(CC3, 'Area');

%Quitar zonas de población, se tomó como criterio eliminar
áreas de 1500 píxeles, si siguen existiendo no se
vectorizan.
L2 = labelmatrix(CC2);
L3 = labelmatrix(CC3);
Z2 = ismember(L2, find([a2.Area] >= 900));
Z3 = ismember(L3, find([a3.Area] >= 900));

figure(14), imshow(Z2), title('Selección zonas de
población I');
figure(15), imshow(Z3), title('Selección zonas de
población P');

%Calcula la diferencia absoluta quitando zonas de
población
Int =imabsdiff(IALTO2,Z2);
figure(16), imshow(Int),title('Eliminación zonas de
población I')
Pend =imabsdiff(SREC,Z3);
figure(17), imshow(Pend),title('Eliminación zonas de
población P')

%dilatación
se1=strel('line',3,0);
se2=strel('line',3,90);
dilateBW1=imdilate(IALTO1,[se1 se2]);
figure(18),imshow(dilateBW1), title('Dilatación RGB');
dilateBW2=imdilate(Int,[se1 se2]);
figure(19),imshow(dilateBW2), title('Dilatación
Intensidad');
dilateBW3=imdilate(Pend,[se1 se2]);
figure(20), imshow(dilateBW3), title('Dilatación
Pendientes');

%cierre
closeBW1=imclose(dilateBW1, se2);
figure(21), imshow(closeBW1),title('Cierre RGB');
closeBW2=imclose(dilateBW2, se2);
figure(22), imshow(closeBW2),title('Cierre Intensidad');
closeBW3=imclose(dilateBW3,se2);
figure(23), imshow(closeBW3),title('Cierre Pendientes');
imwrite(closeBW1, 'Pistas_RGB.tif');
imwrite(closeBW2, 'Pistas_Intensidad.tif');
imwrite(closeBW3, 'Pistas_Pendientes.tif');

```

```

%combinamos imágenes binarias
%pistas finales
F1=or(closeBW1,closeBW2);
F=or(F1,closeBW3);
figure (24),imshow(F);title('Pistas Detectadas');
    imwrite(F,'Pistas_Detectadas.tif');

%esqueletoniza
%TV funciona mejor con el esqueleto
skel=bwmorph(F,'skel',Inf);
figure (25),imshow(skel);title('Esqueletonización');
B = bwmorph(skel, 'branchpoints');
E = bwmorph(skel, 'endpoints');
[y,x] = find(E);
B_loc = find(B);
Dmask = false(size(skel));
for k = 1:numel(x)
    D = bwdistgeodesic(skel,x(k),y(k));
    distanceToBranchPt = min(D(B_loc));
    Dmask(D < distanceToBranchPt) =true;
end
%eliminamos ramas
skelD = skel - Dmask;
skelP=logical(skelD);
figure(26),imshow(skelP), title('Limpieza de ramas');
imwrite(skelP,'Poda_skel.tif');

```