Procesado de datos LiDAR en QGIS con SPDlib

Roberto Antolín[1], Pete Bunting[2] y Juan Suarez[1]

(1) Forest Research, Northern Research Station, Roslin EH25 9SY, UK
(2) Department of Geography and Earth Sciences, Aberystwyth University, Aberystwyth, Ceredigion, SY23 3DB, UK

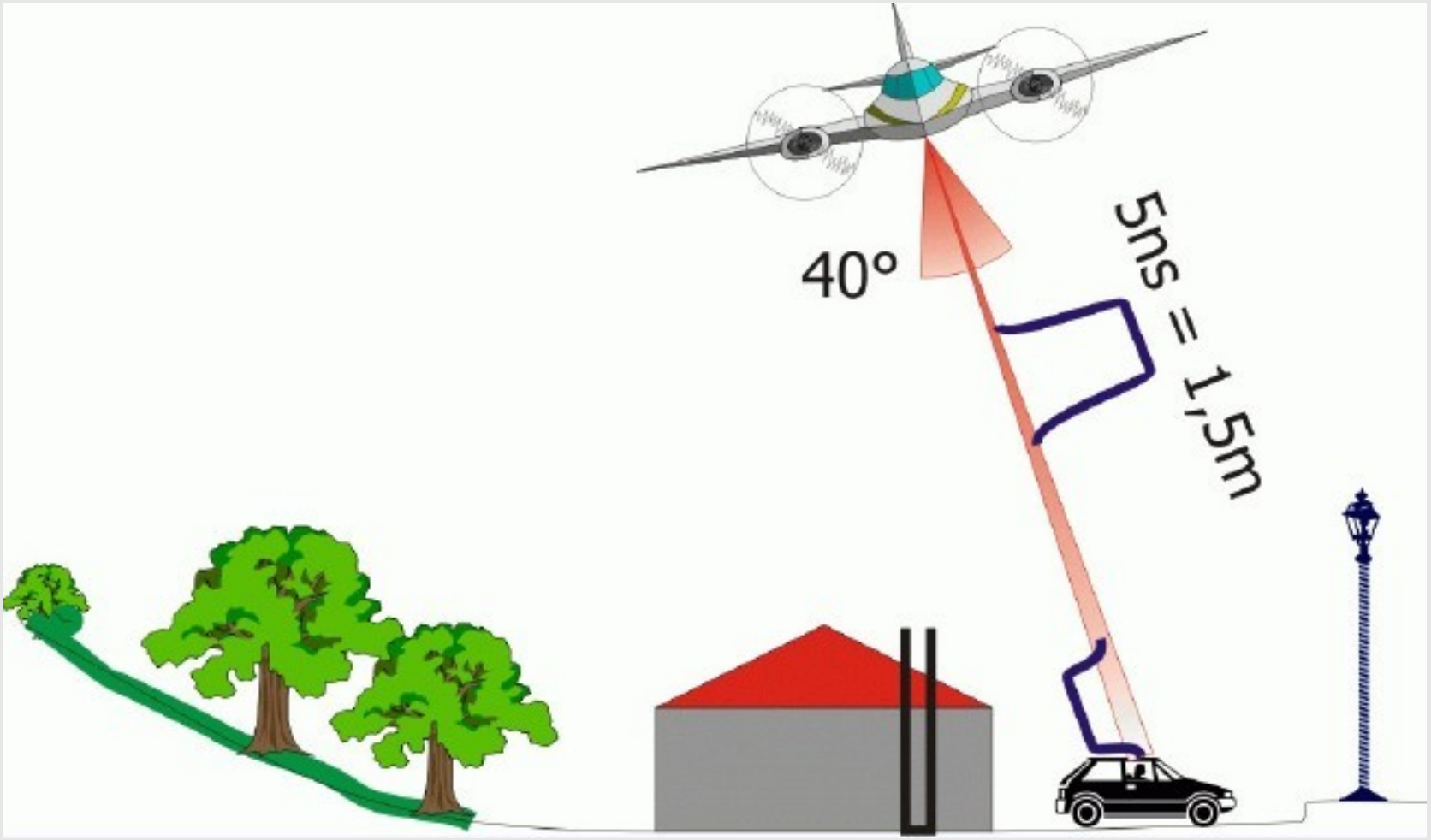
E-mail: Roberto.antolin@forestry.gsi.gov.uk
Twitter: @Tolanss

Resumen

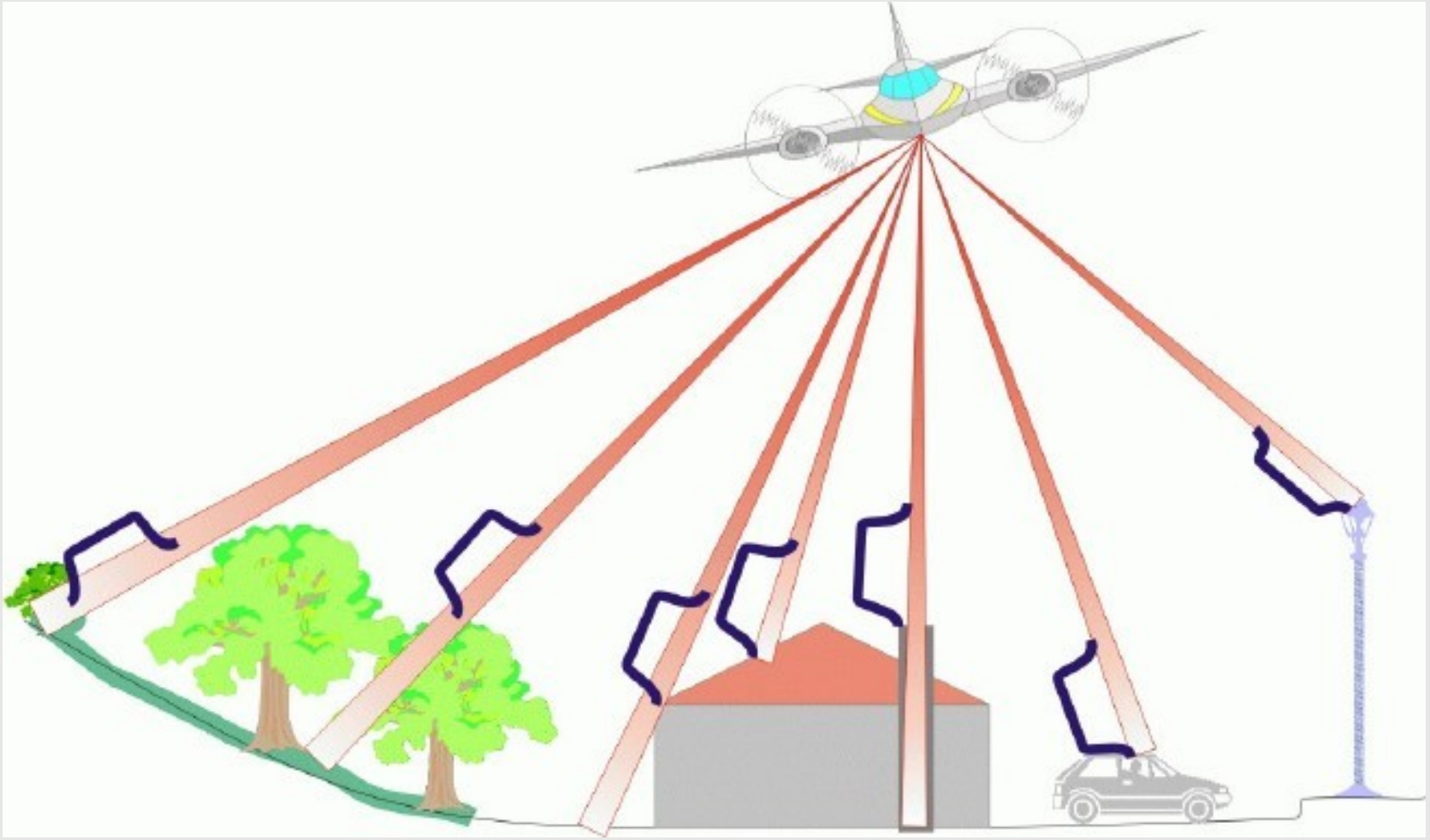
1. Motivaciones
   - Qué es el LiDAR?
   - Herramientas actuales LiDAR
2. SPDlib
   - Nuevo formato SPD: Por qué?
3. Plugin en QGIS
   - Flujo de trabajo LiDAR
   - Módulos
4. Conclusiones

MOTIVACIONES

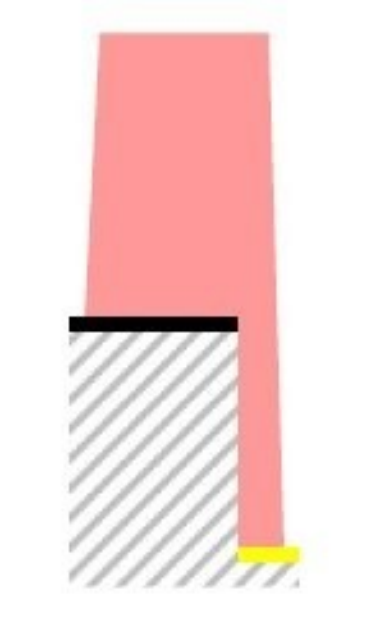
Qué es LiDAR?

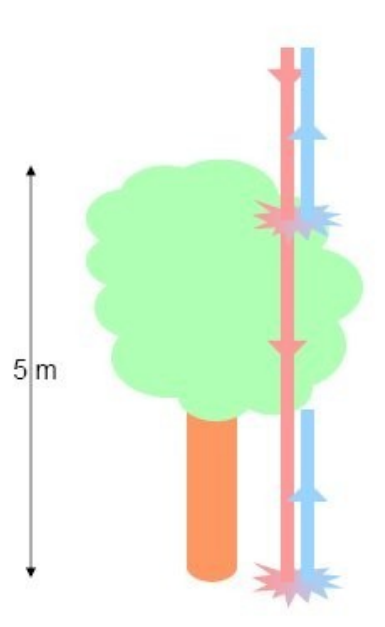


Qué es LiDAR?



Qué es LiDAR?





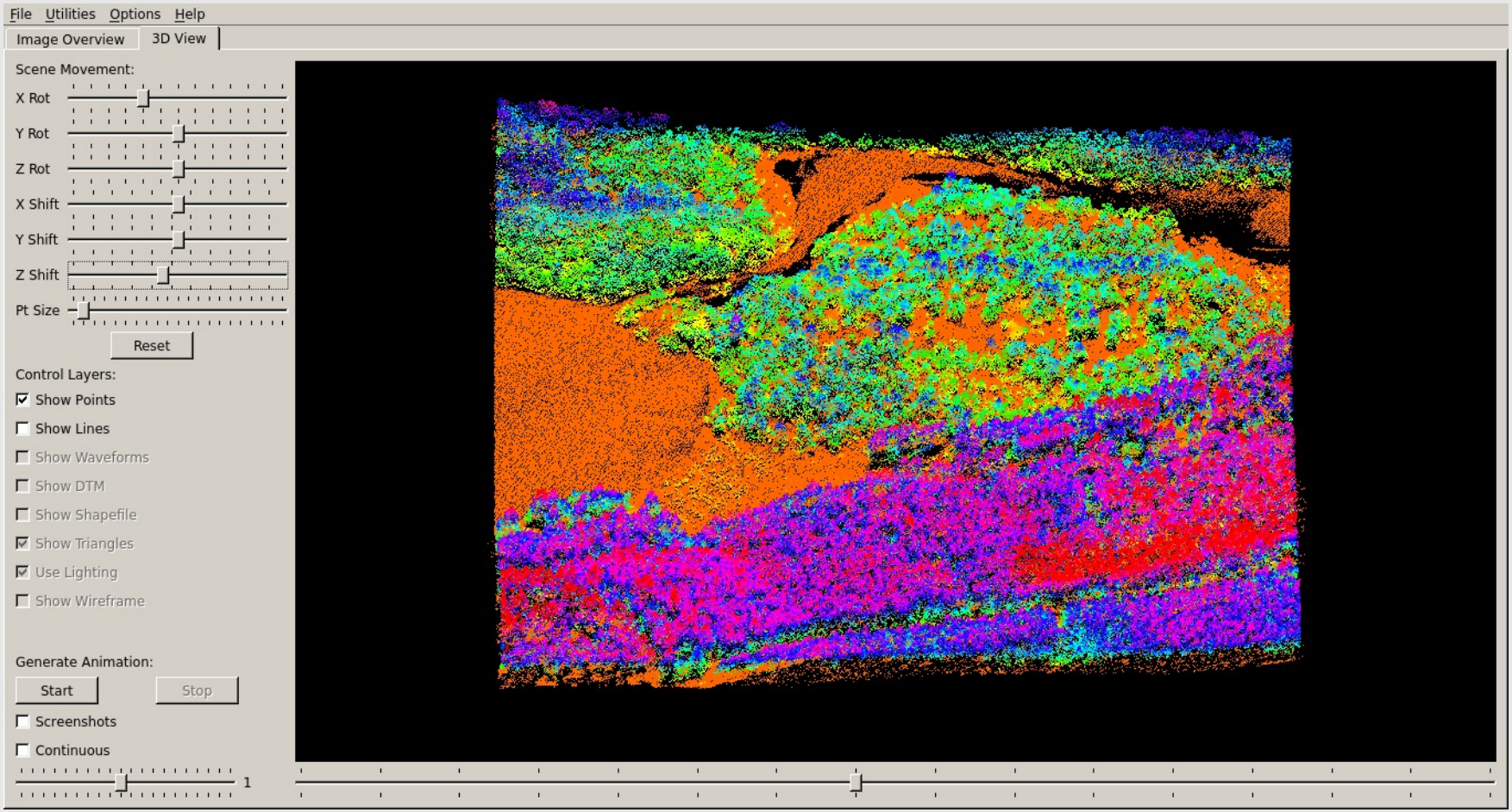
Varios impulsos:

- **FIRST**
- **LAST**

Actualmente hasta **5**

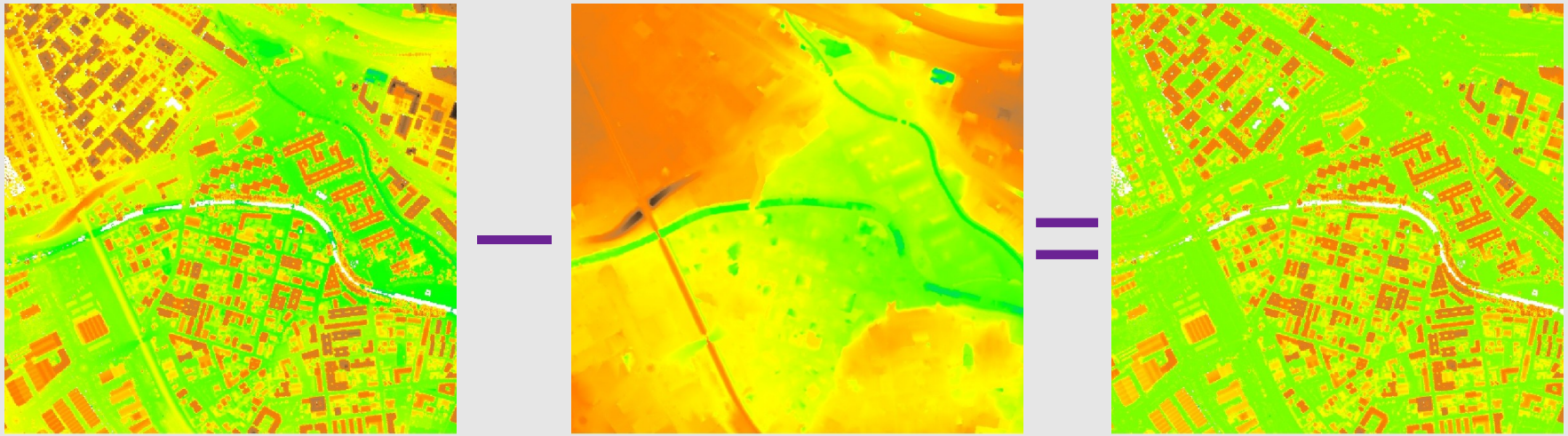
Datos LiDAR

1. **Millones** y **millones** de puntos
2. Fichero LiDAR estándar **LAS, LAZ** (comprimido)



Productos básicos

MDS – MDT = nMDS



Problema: Millones de puntos

1. Cada retorno (Tiempo, distancia, ángulo escaneo, intensidad, coordenadas...): Más de **40 bytes**
2. Ejemplo: 100 km^2 × 1 p/m^2
   - 10000×10000×4×26 bytes ≃ **10.43Gb**
3. Caso real: 20×25km^2 × 5 p/m^2 ≃ **160Gb**
4. Algoritmos especiales
   - Clasificación
   - Manipulación del volumen de información

Problema?? Millones de puntos

1. Cada retorno (Tiempo, distancia, ángulo escaneo, intensidad, coordenadas...): Más de **40 bytes**
2. Ejemplo: 100 km^2 × 1 p/m^2
   - 10000×10000×4×26 bytes ≃ **10.43Gb**
3. Caso real: 20×25km^2 × 5 p/m^2 ≃ **160Gb**
4. Algoritmos especiales
   - Clasificación
   - Manipulación del volumen de información

This sucks! GAIA ROCKS

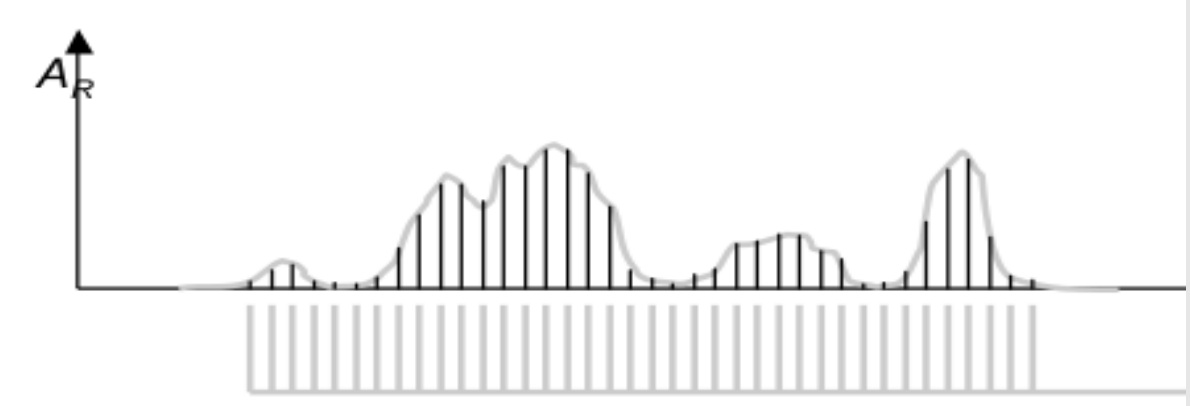
Herramientas

1. TerraScan y TerraSolid
   - Microstation (CAD)
   - Software privativo
2. LASTools
   - Librería libre (LGPLv3) para I/O formato **LAS**
   - Conjunto de herramientas: Software privativo
3. LibLAS
   - Librería libre (GPLv3) para I/O formato **LAS**
4. FUSION
   - Gratuito pero cerrado

HERRAMIENTAS SPDLIB

SPDLib

1. Librerías para I/O datos LiDAR
2. Conjunto de herramientas para la **manipulación**, **procesamiento** y **análisis** LiDAR:
   - **ALS** (Aerotransportado)
   - **TLS** (Terrestre)
   - ***Full waveform***



SPDLib

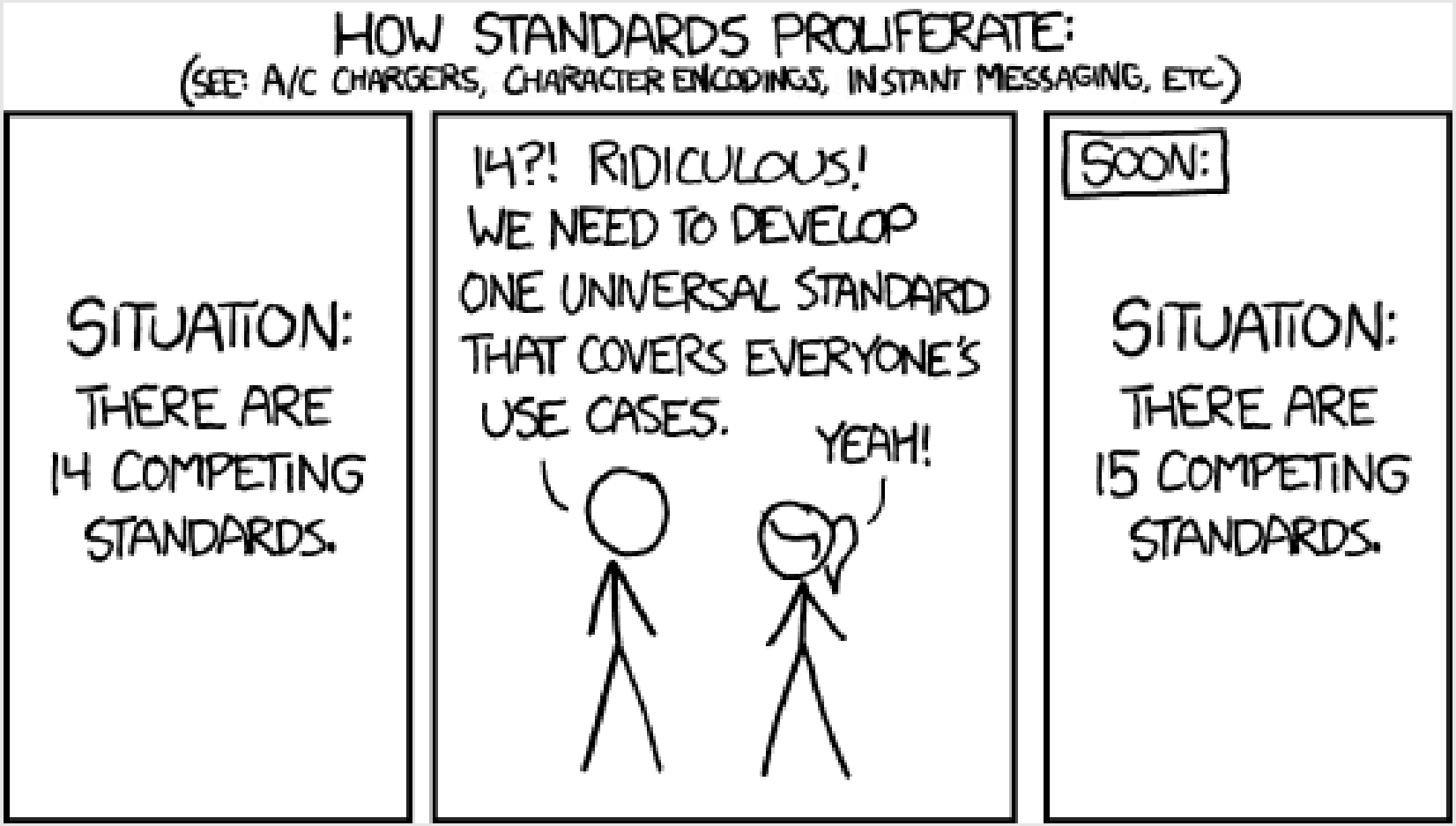
1. Libres: PGLv3
2. C++
3. Bindings para Python
4. Multiplataforma
5. http://www.spdlib.org/

Por qué SPDlib?

Function	FUSION	LASTools	SPDLib	Terrascan
License	BSD	LGPL / Commercial	GPL3 / MIT-X	Commercial
Source Code	Closed	Mixed Open / Close	Open	Close
Import LAS/ASCII	Yes	Yes	Yes	Yes
Raster formats	ASCII / Binary	ASCII / GTIFF	GDAL	ASCII / Proprietary
Canopy Height Metrics	Yes	Yes	Yes	No
Point Cloud Z Metrics	No	No	Yes	No
Intensity Metrics	No	No	Yes	No
Metrics Calculator	No	No	Yes	No
Ground Classification	Yes	Yes	Yes	Yes
Merge Point Clouds	Yes	Yes	Yes	Yes
Interpolate Surfaces	Yes	Yes	Yes	Yes
Assign RGB Values	No	Yes	Yes	Yes
Tile Point Cloud	No	Yes	Yes	Yes
Mosaic Rasters	Yes	No	Yes	Yes
Data Indexing	No	Yes	Yes	Yes
Waveform Processing	No	No	Yes	No

Inconvenientes

Nuevo formato **SPD** (**S**orted **P**ulse **D**ata)



SPDLib Tools

- Management tools:
  - **spdcopy**: Makes a copy of an indexed SPD file
  - **spddeftiles**: Tools for defining a set of tiles ✓
  - **spdextract**: Extract returns and pulses which meet a set of criteria
  - **spdmerge**: Merge compatable files into a single non-indexed SPD file ✓
  - **spdproj**: Print and convert projection strings
  - **spdtiling**: Tools for tiling a set of SPD files using predefined tile areas ✓
  - **spdthin**: Thin a point cloud to a defined bin spacing
  - **spdtranslate**: Convert between file formats ✓
  - **spdsubset**: Subset point cloud data ✓

- Info tools:
  - **spdinfo**: Print header info for an SPD File ✓
  - **spdlastest**: Print data pulses from a LAS file - for debugging
  - **spdstats**: Provides statistics the point and pulse density of an SPD file
  - **spdversion**: Prints version information

- Interpolation methods:
  - **spdwarp**: Apply a nonlinear warp to the SPD file defined by a set of GCPs
  - **spdinterp**: Interpolate a raster elevation surface ✓

SPDLib Tools

- Terrain filters:
  - **spdmccgrd**: Classifies the ground returns using the multiscale curvature algorithm ✓
  - **spdpffgrd**: Classifies the ground returns using a parameter-free filtering algorithm
  - **spdpmfgrd**: Classifies the ground returns using the progressive morphology algorithm ✓
  - **spdpolygrd**: Classify ground returns using a surface fitting algorithm
- Forest tools:
  - **spdmetrics**: Calculate metrics ✓
- Heights:
  - **spdefheight**: Alter the elevation of the pulses ✓
  - **spdprofile**: Generate vertical profiles
- Miscellaneous
  - **spdclearclas**: Clear the classification of an SPD file
  - **spdmaskgen**: Generate a binary mask for the an input SPD file
  - **spdoverlap**: Calculate the overlap between UPD and SPD files
  - **spdtileimg**: Tools for mosaicing raster results following tiling

SPDLib Tools

1. Ejemplo:

```
# Transformar de LAS a SPD
$ > spdtranslate -i fichero_LAS.las -o fichero_SPD.spd \
    --if LAS --of SPD -x FIRST_RETURN
# Transformar de LAZ a SPD
$ > spdtranslate -i fichero_LAZ.laz -o fichero_SPD.spd \
    --if LAS --of SPD -x FIRST_RETURN
# Consultar metadatos
$ > spdinfo fichero_SPD.spd
```

PLUGIN EN QGIS

Plugin SDPlib en QGIS

1. **Qué**? Interfaz gráfica para las herramientas SPDlib
2. **Dónde** se integra? *Processing* ToolBox
3. **Cómo**? ***Python 2.7***
4. **Por qué** QGIS? Modularidad, capacidad, prestaciones, es **libre** y... es **FÁCIL**

SDPlib en QGIS: Cómo?

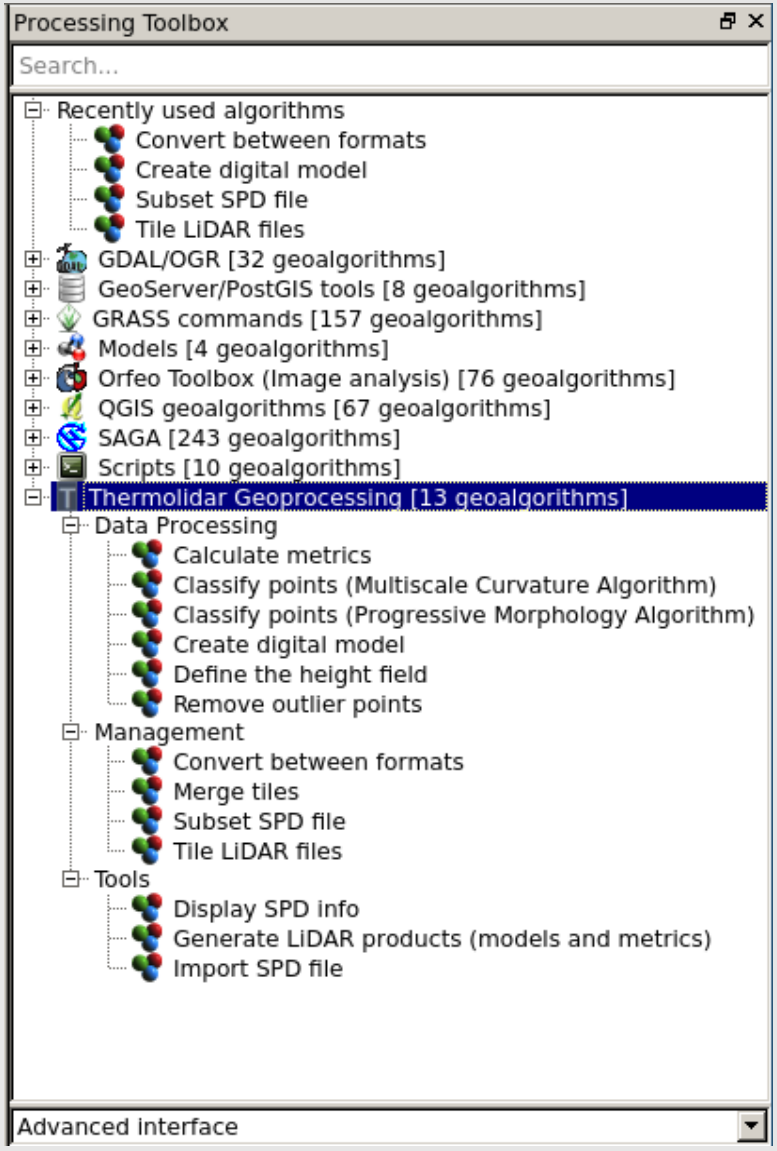
Librerías

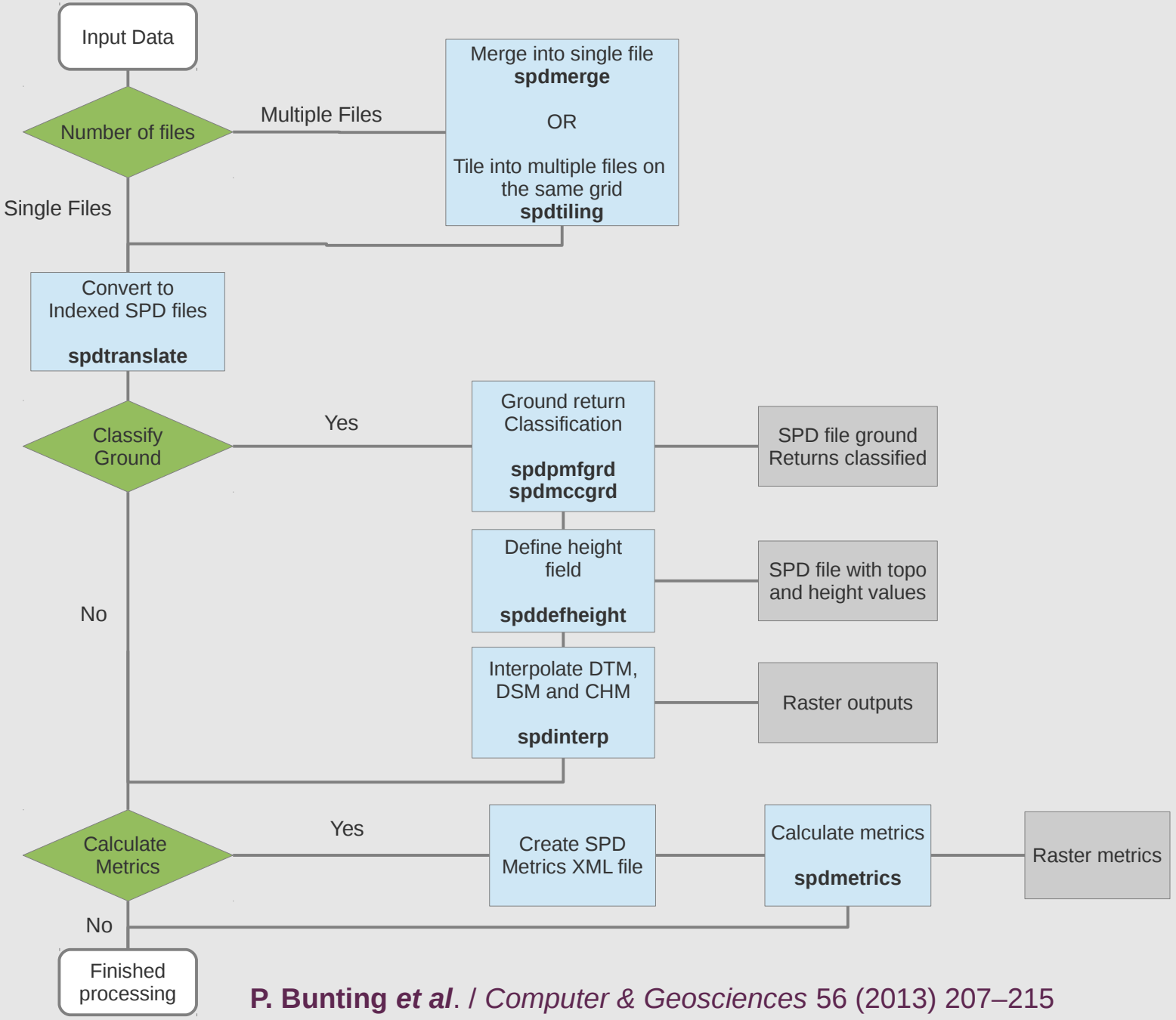
```
from processing.thermolidar.process.lidar.SPDToolsUtils import SPDToolsUtils
from processing.thermolidar.process.lidar.SPDToolsAlgorithm import SPDToolsAlgorithm
from processing.parameters.ParameterRaster import ParameterRaster
from processing.outputs.OutputHTML import OutputHTML
```

Parámetros

```
self.addParameter(ParameterFile(spdinfo.INPUT, "Input layer"))
self.addOutput(OutputHTML(spdinfo.OUTPUT, "Output file"))
```

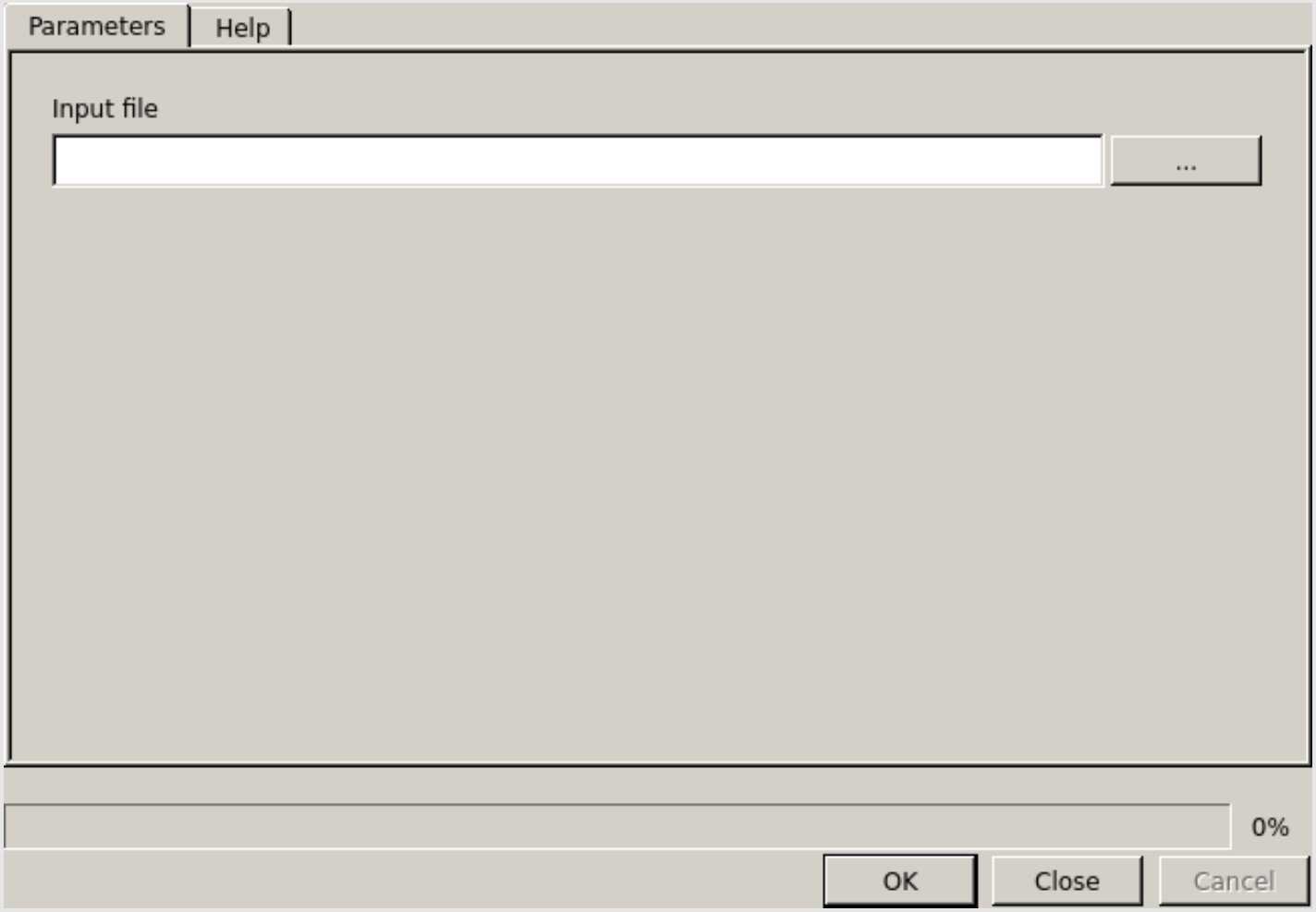
Processing Toolbox



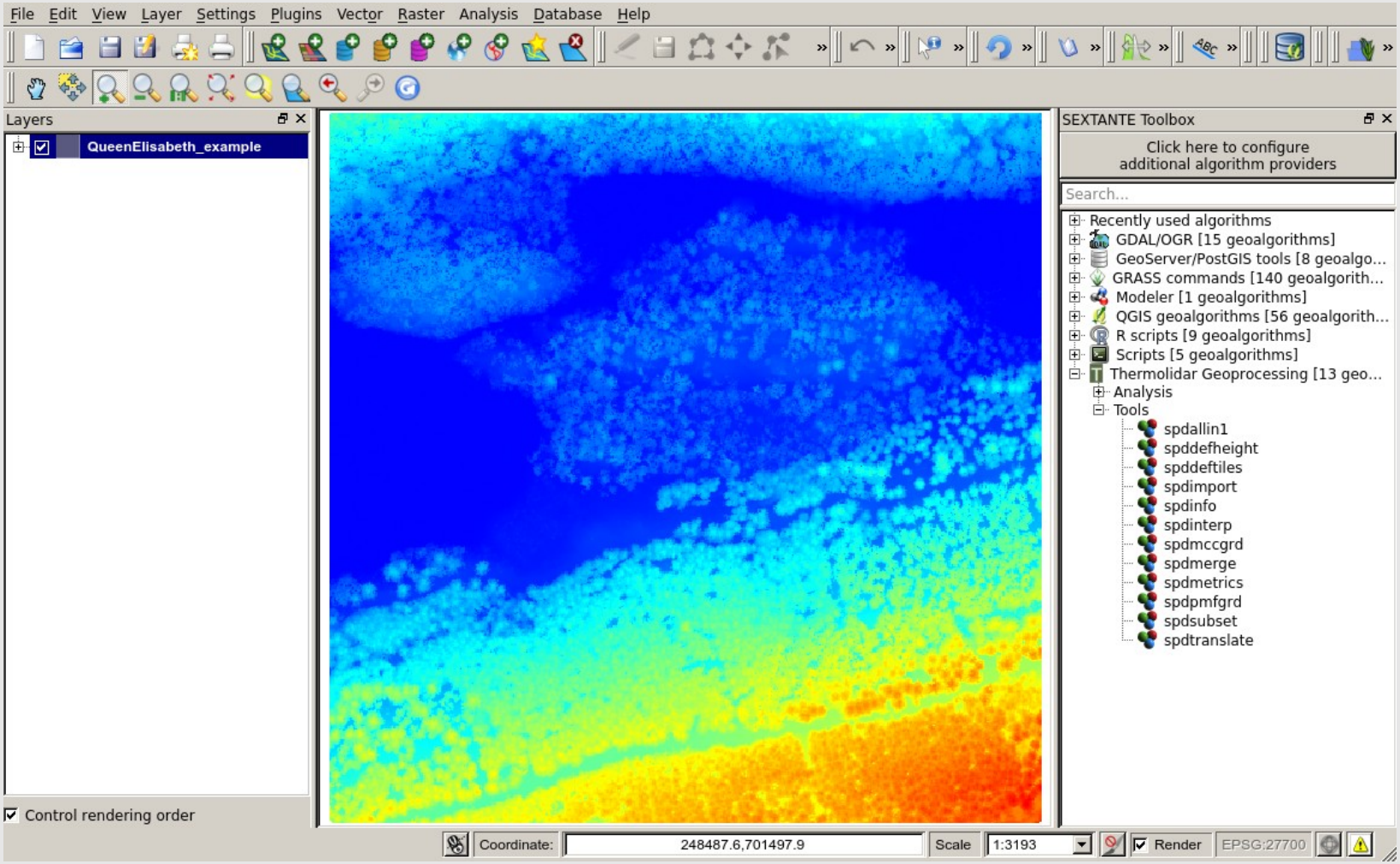


P. Bunting *et al*. / *Computer & Geosciences* 56 (2013) 207–215

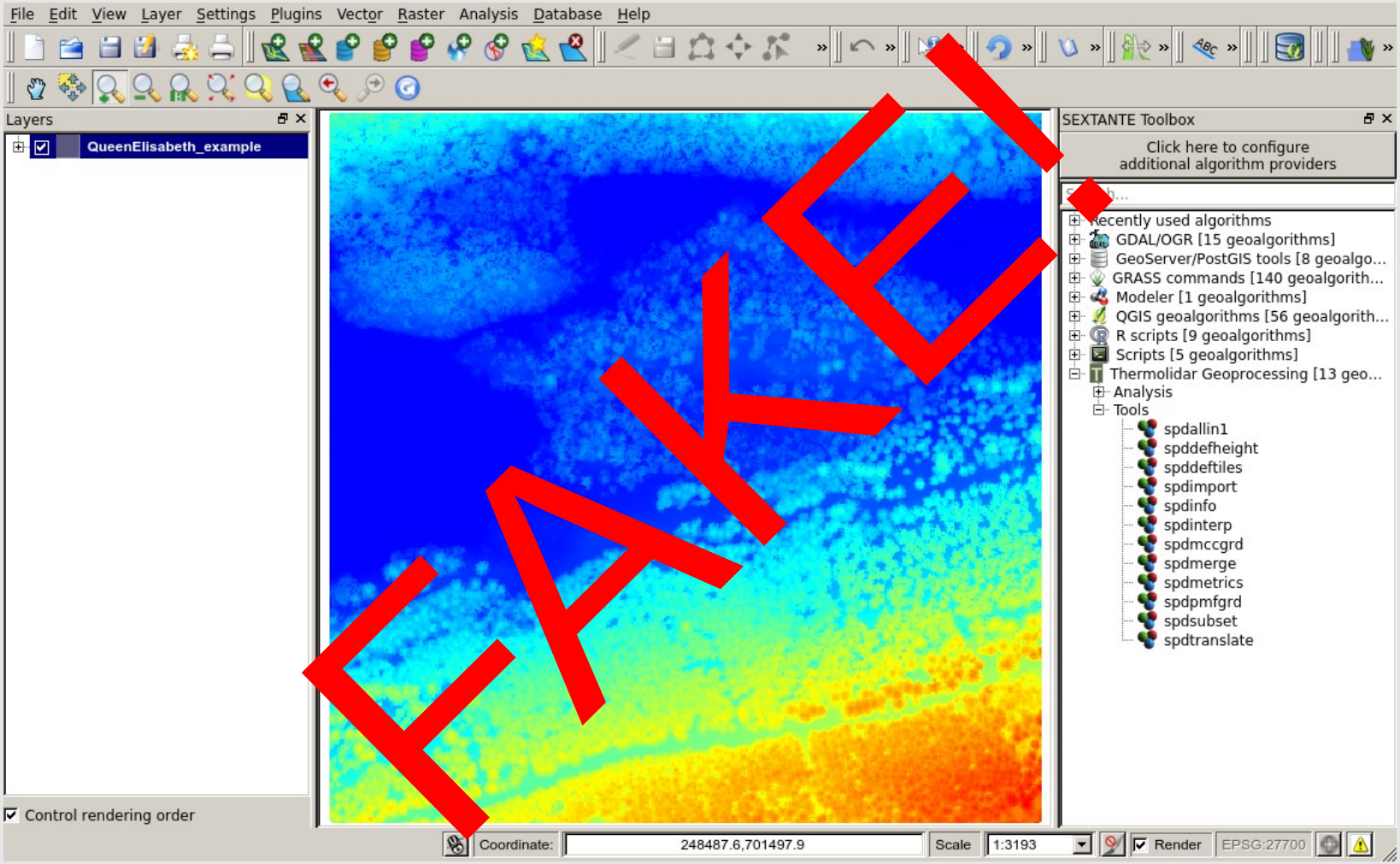
Importar datos



Importar datos

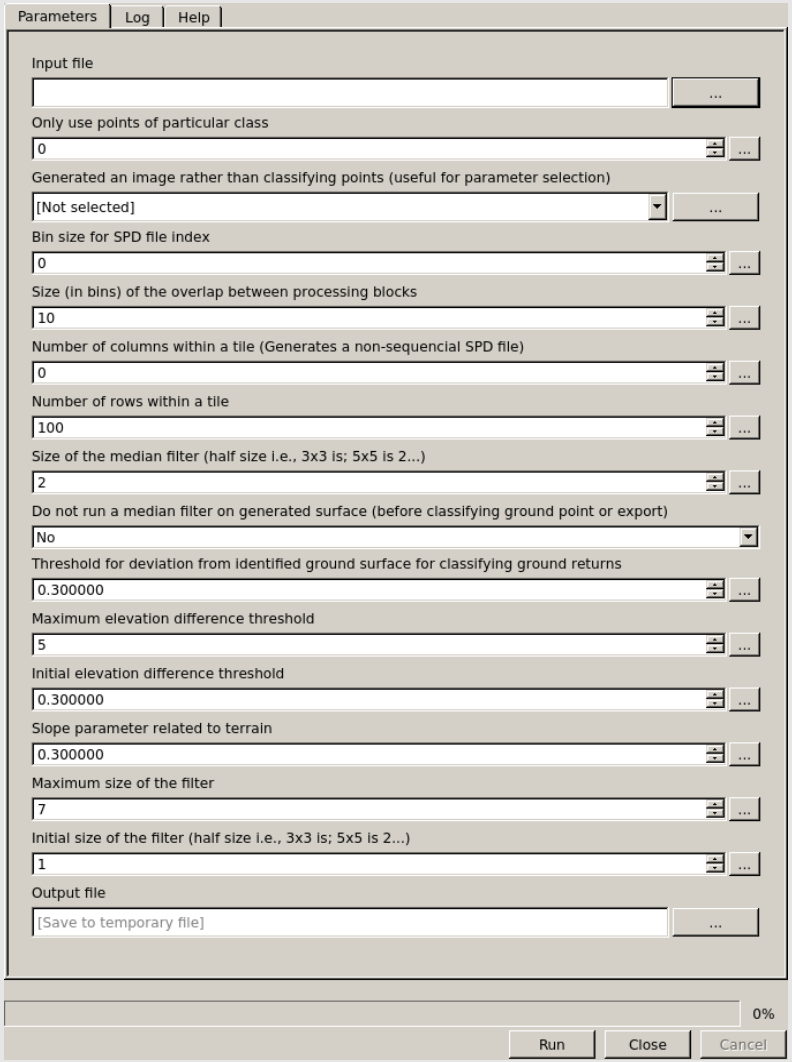


Importar datos



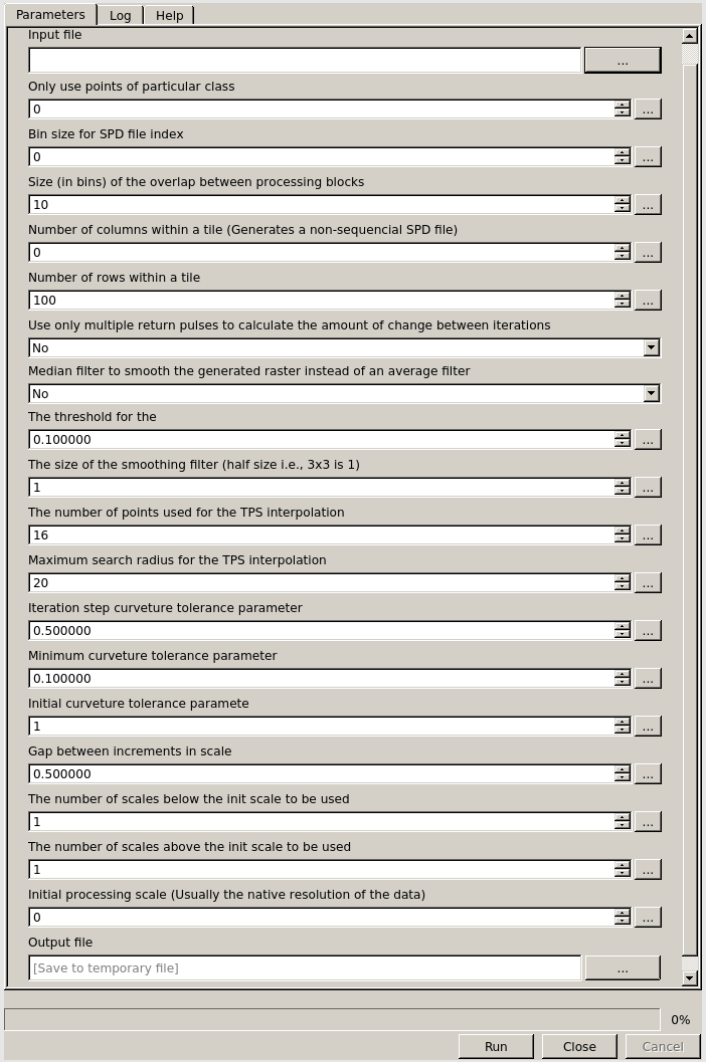
Clasificar puntos: Algoritmo morfológico progresivo (PMF)

1. **Input**
2. **Output**
3. **Bin size**:
   - Indexado de puntos
4. **Class**:
   - Clases particulares

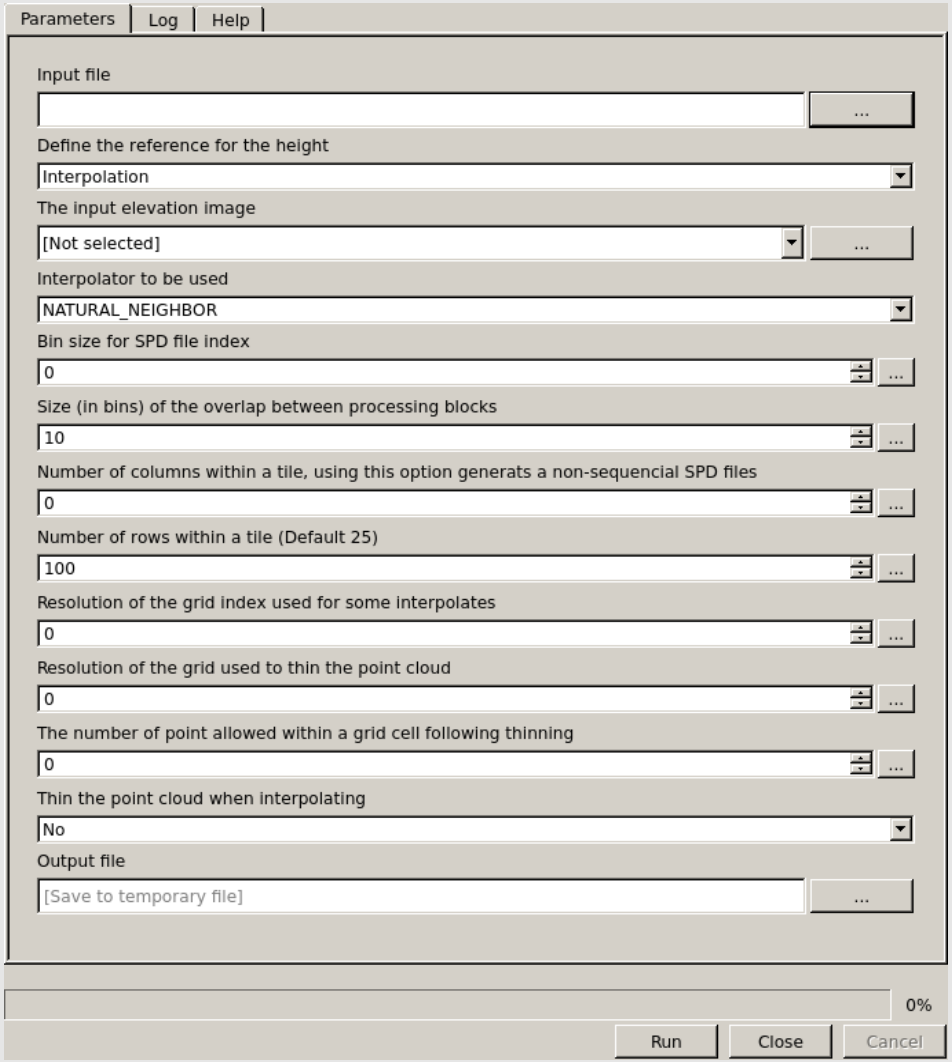


Clasificar puntos: Algoritmo curvatura multiescala (MCC)

1. **Input**
2. **Output**
3. **Bin size**:
   - Indexado de puntos
4. **Class**:
   - Clases particulares



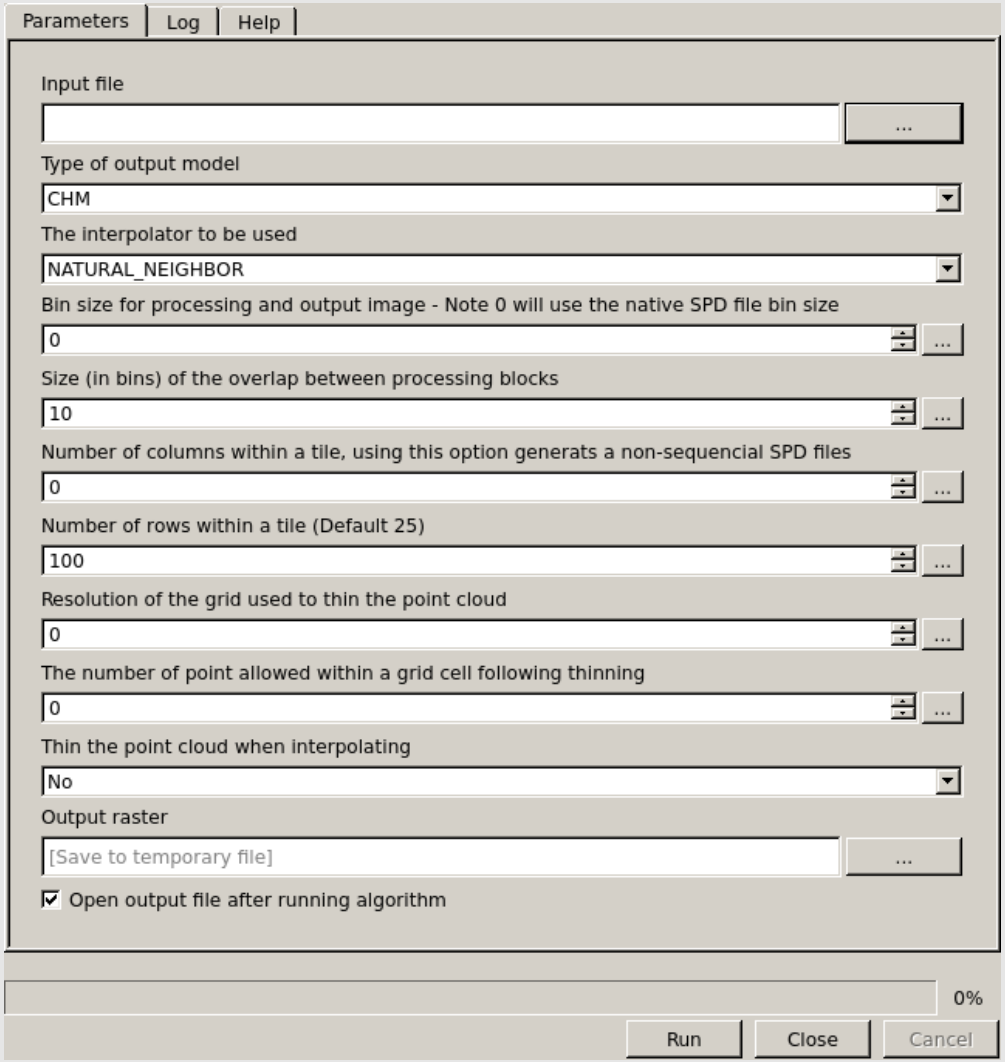
Definir alturas (respecto al terreno)



Definir alturas (respecto al terreno)

1. **Input**
2. **Output**
3. Superficie de referencia:
   - Interpolación
   - Imagen
4. **Bin size**: Indexado

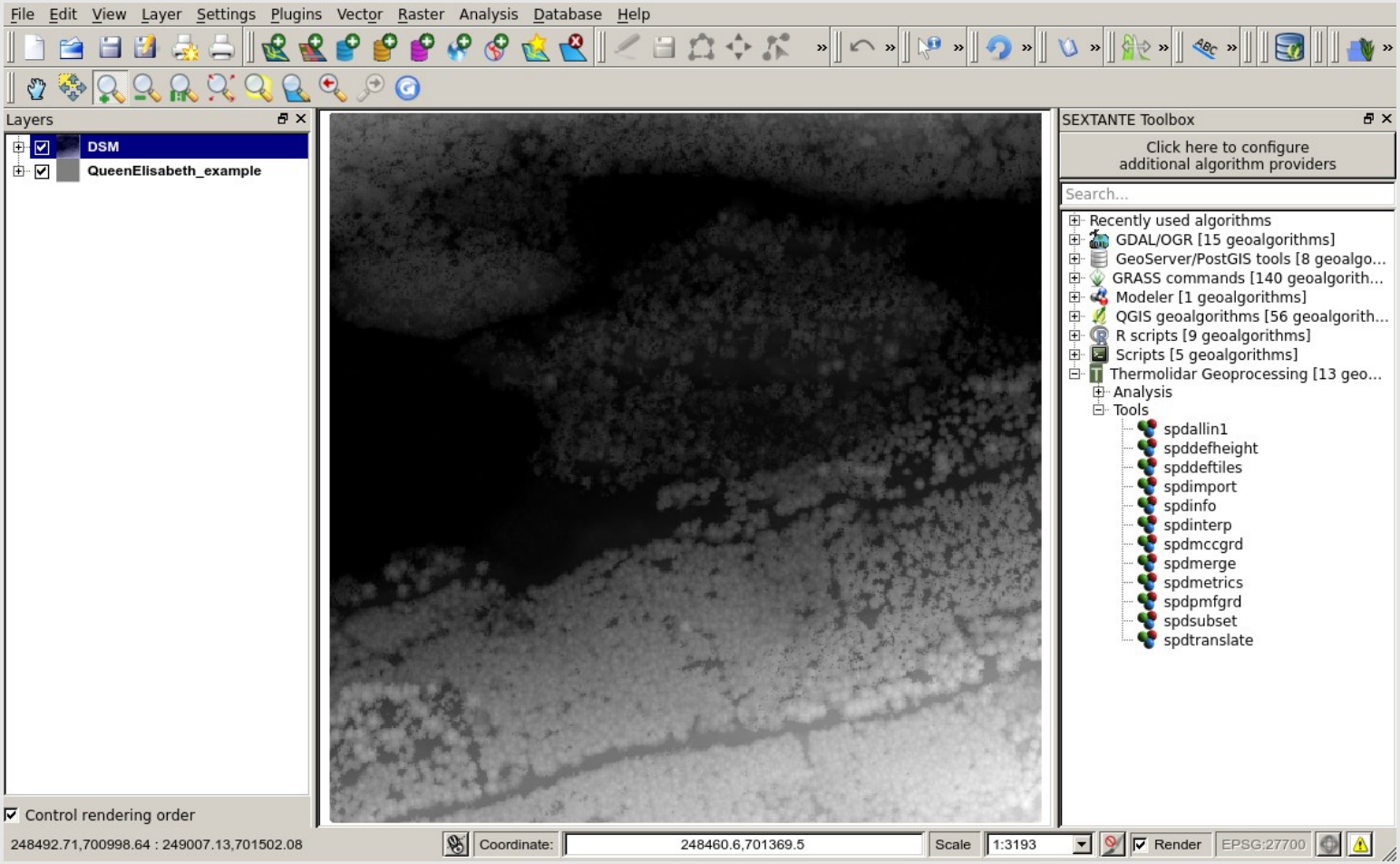
Crear modelos digitales



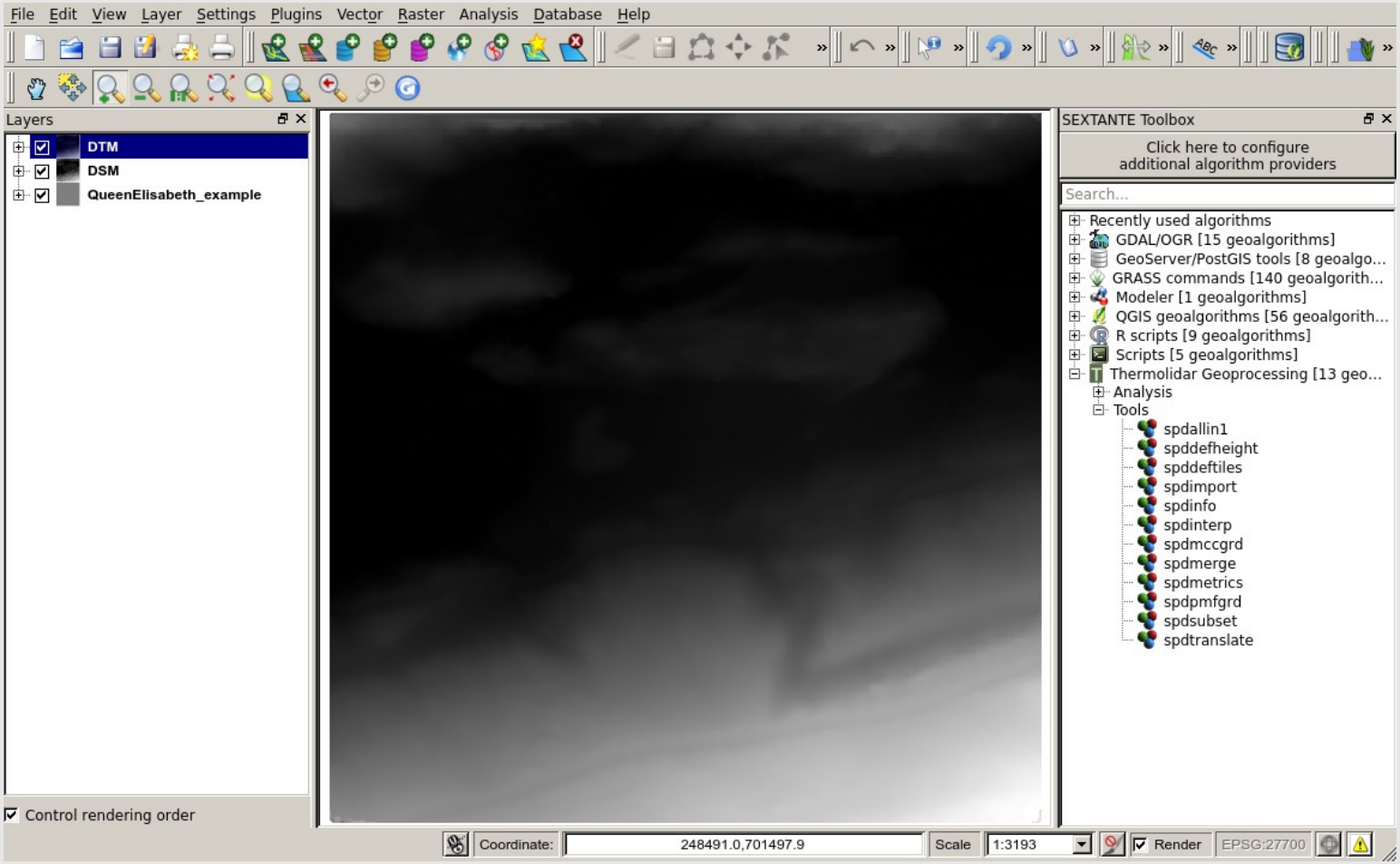
Crear modelos digitales

1. **Input**
2. **Output**
3. ***Bin size***: Resolución
4. Tipo de **modelo**
    - MDT
    - MDS
    - CHM (Modelo de copas = **C**anopy **H**eight **M**odel)
5. **Interpolador**
    - NATURAL_NEIGHBOR: Vecino natural
    - NEAREST_NEIGHBOR: Vecino más próximo
    - TIN_PLATE: Linear

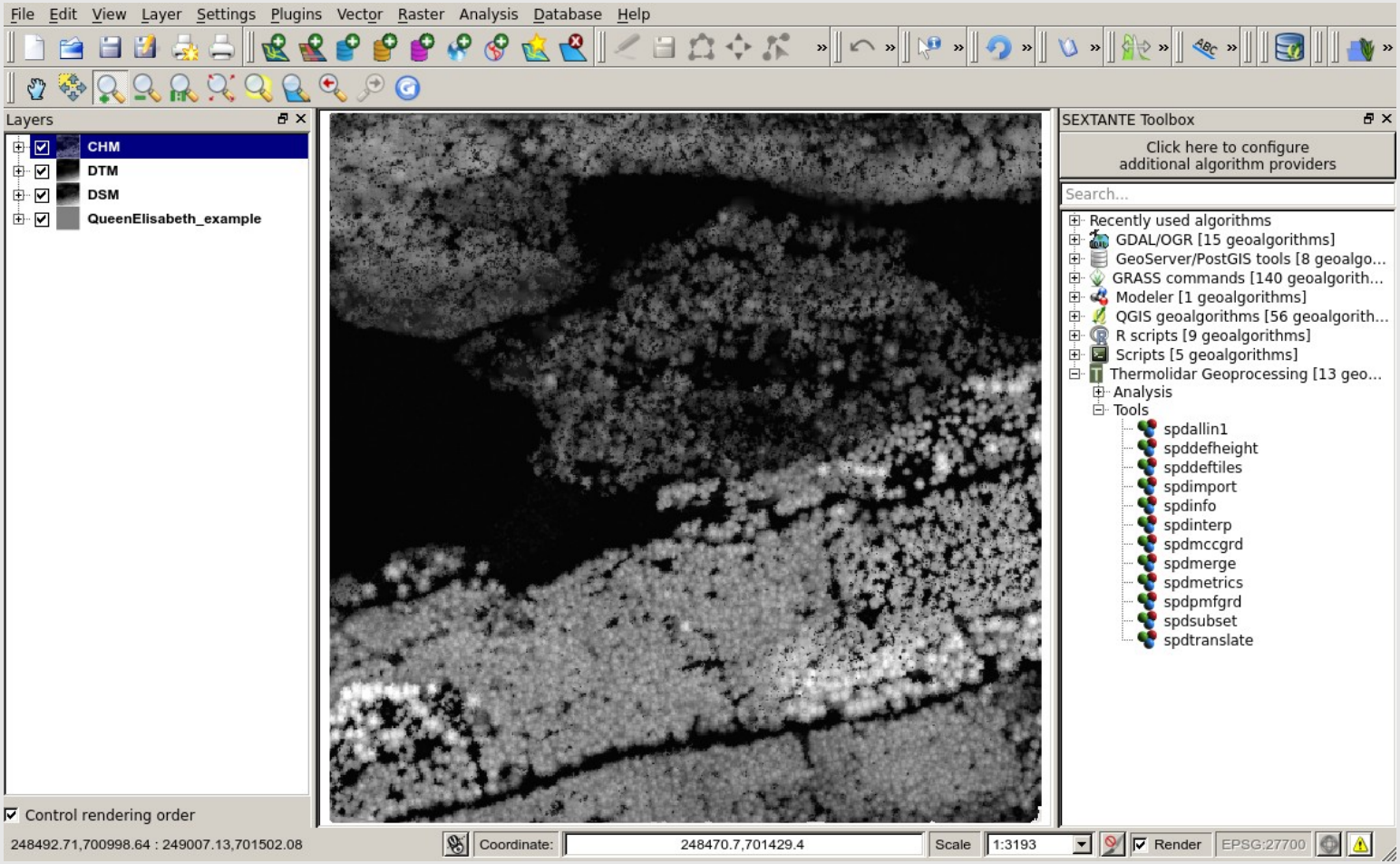
Visualizar MDS



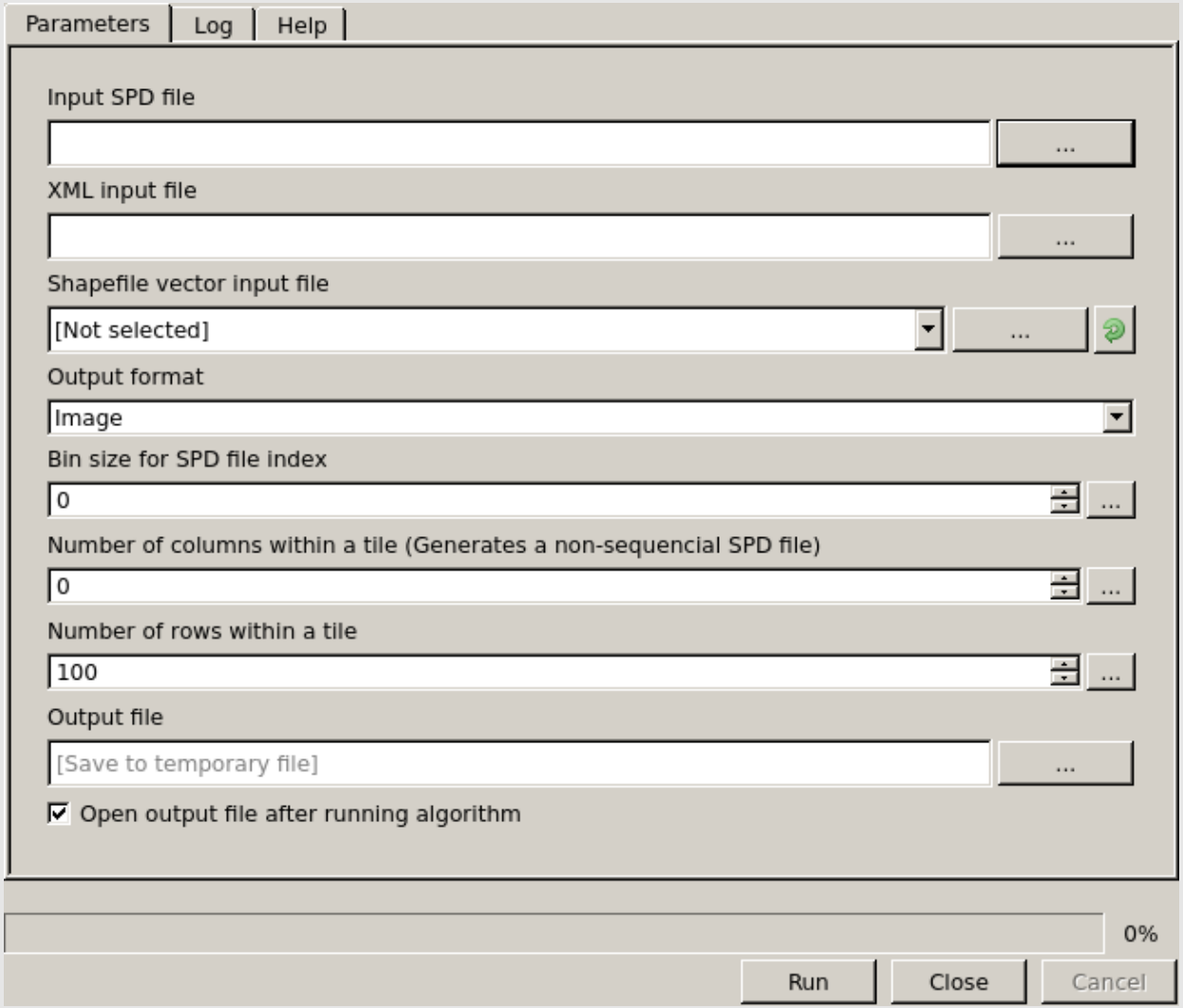
Visualizar MDT



Visualizar CHM



Calcular métricas



Calcular métricas

1. Utiliza XML:
   - **Metrics**: Nombre de la métrica
   - **Field**: El nombre asignado al atributo shp
   - **Options**: Opciones relacionadas con la métrica
     - *return*
     - *class*
     - *threshold*

Calcular métricas

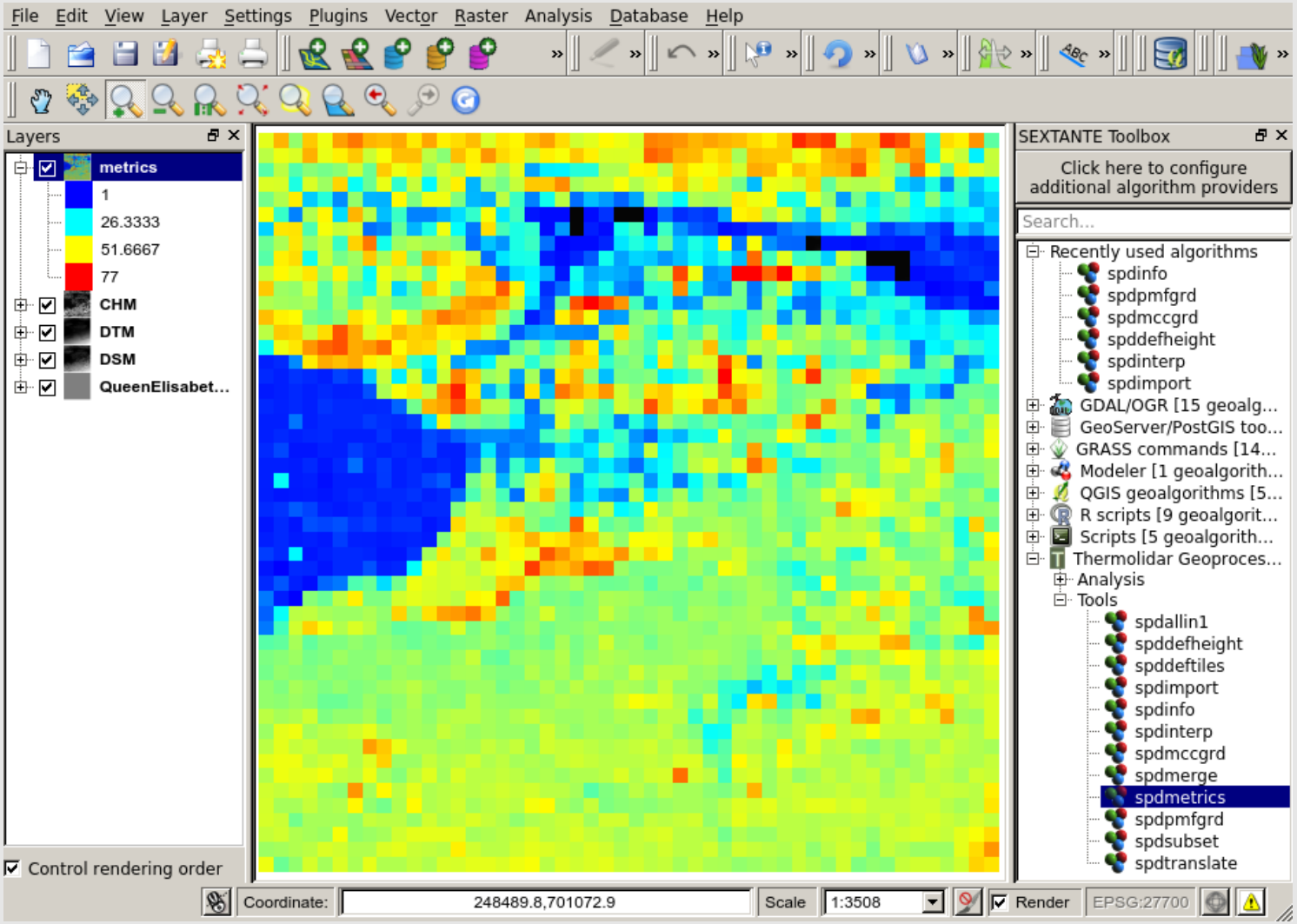
```
<?xml version="1.0" encoding="UTF-8" ?>
<!--
    Description:
        XML File for execution within SPDLib
        This file contains a template for the
        metrics XML interface.

    Created by Roberto Antolin on Thu Jan 30 16:33:36 2014.
    Copyright (c) 2014 Roberto Antolin.
-->

<spdlib:metrics xmlns:spdlib="http://www.spdlib.org/xml/">
    <spdlib:metric metric="percentileheight" field="95thPerH" percentile="95" return="All" class="NotGrd" lowthreshold="0.1" />
    <spdlib:metric metric="meanheight" field="MeanH" return="All" class="NotGrd" lowthreshold="2" />
    <spdlib:metric metric="numpulses" field="numPulses" minNumReturns="0" />
    <spdlib:metric metric="divide" field="groundCover" >
        <spdlib:metric metric="numreturnsheight" field="Out_Name" return="All" class="Grd" upthreshold="40" />
        <spdlib:metric metric="numreturnsheight" field="Out_Name" return="All" class="All" upthreshold="40" />
    </spdlib:metric>
    <spdlib:metric metric="divide" field="CanopyCover" >
        <spdlib:metric metric="numreturnsheight" field="Out_Name" return="All" class="NotGrd" upthreshold="40" />
        <spdlib:metric metric="numreturnsheight" field="Out_Name" return="All" class="All" upthreshold="40" />
    </spdlib:metric>
</spdlib:metrics>
```

1. **metrics**: *percentilheight*
2. **field**: *95thPerH*
3. **options**:
   - **Percentile**: *95%*
   - **return**: *All*
   - **class** *NotGrd*
   - **lowthreshold**: *0.1*

Calcular métricas: Percentil 95%



CONCLUSIONES

Conclusiones

1. **LiDAR** requiere de mucha capacidad y algoritmos especiales por su volumen de datos
2. **No** existen herramientas totalmente libres
3. **SPDlib** es una herramientas libre para la manipulación de datos LiDAR, con capacidad para
   - **Procesar**: teselar, clasificar y determinar alturas
   - **Generar modelos**: MDT, MDS, Nmds
   - **Generar métricas**
4. SPDlib se ha integrado en **QGIS** como puglin basado en módulos

Procesado de datos LiDAR en QGIS con SPDlib

GRACIAS!

E-mail: Roberto.antolin@forestry.gsi.gov.uk
Twitter: @Tolanss