1. Ground filtering

1.1 General

To generate the DTM rasters, we must first filter out the non-ground points (points that are not part of the bare-earth surface of the earth) from the point cloud. The AHN offers a well-classified point cloud that can be used to easily separate ground points from the point cloud and use them as an input for our interpolation algorithm. A list depicting the first ten classifications maintained by the AHN can be seen in Table 1.

Code	Meaning
0	never classified
1	unclassified
2	ground
3	low vegetation
4	medium vegetation
5	high vegetation
6	building
7	low point (noise)
8	*reserved*
9	water

Table 1. The first 10 classification code numbers of AHN3 raw point clouds. More codes exist, but they are not listed here. (sources needed terrain book)

However, we decided not to trust the AHN classification and conduct a series of tests to assess the quality of the classification and to determine if we are going to use it or not. Luckily, other ground filtering tools/algorithms that could possibly improve the classification and deliver better results are available namely, PDAL library that offers algorithms like the Progressive Morphological Filter (PMF) and the Simple Morphological Filter (SMRF), Cloudcompare that offers cloth simulation filter (CSF) and LAStools that have the lasgound tool (and lasground_new) used for ground filtering. To decide whether we are should rely on the classification of the AHN, we compared the results from using the available ground filtering algorithms we mention above on 5 data samples we clipped from the AHN point cloud with the AHN classification. For our testing samples, we tried to capture different sceneries that we will come across when processing the whole Netherlands and our code will have to deal with. The five locations that we chose are the city centre of Amsterdam (city centre), two less densely populated areas in the cities of Delft (city) and Groningen (city outskirts), a part of the National Park Veluwezoom (forests and heathland) and a part of the National Park De Biesbosch (willow forests, wet grasslands and fields of reed).
The assessment of the quality of the AHN classification we conducted consists of the following steps:

- Filtering out points that belong to the following class before applying some of the algorithms: “buildings”, “low points” and “water”. This led to better results.
- Applying the available ground filtering algorithms on the AHN raw point cloud samples.
- Examining the results and comparing them with AHN ground points.
- Writing a statistical and visual comparison report of the results.

The results of the assessment indicate that all these algorithms we tested, except for CSF, can provide us with a good result. However, they have flaws in some of the 5 test data samples (figures #). For instance, the lasground tool delivers good results but with wrongly classified points. It performs poorly in urban areas like Amsterdam which gives to the AHN classification

an edge. On the other hand, PDAL algorithms give the best results that can rival the AHN classification only when we remove points classified as water from the whole process.
For PDAL we made use of the so-called Pipelines which define the processing of data within PDAL. (source: https://pdal.io/index.html) Using a pipeline, we specified what algorithms to apply to the data. More specifically, besides the two ground filtering we tested we also made use of the following filters:

- filters.range: to exclude some points that belong to certain classes from the whole process and achieve better results. It is also used again at the very end to store only the points classified as ground in the result.
- filters.assign: to reclassify all points to the same class.
- filters.elm: to mark low points as noise with the use of the Extended Local Minimum (ELM) filter.
- filters.outlier: to remove outliers.

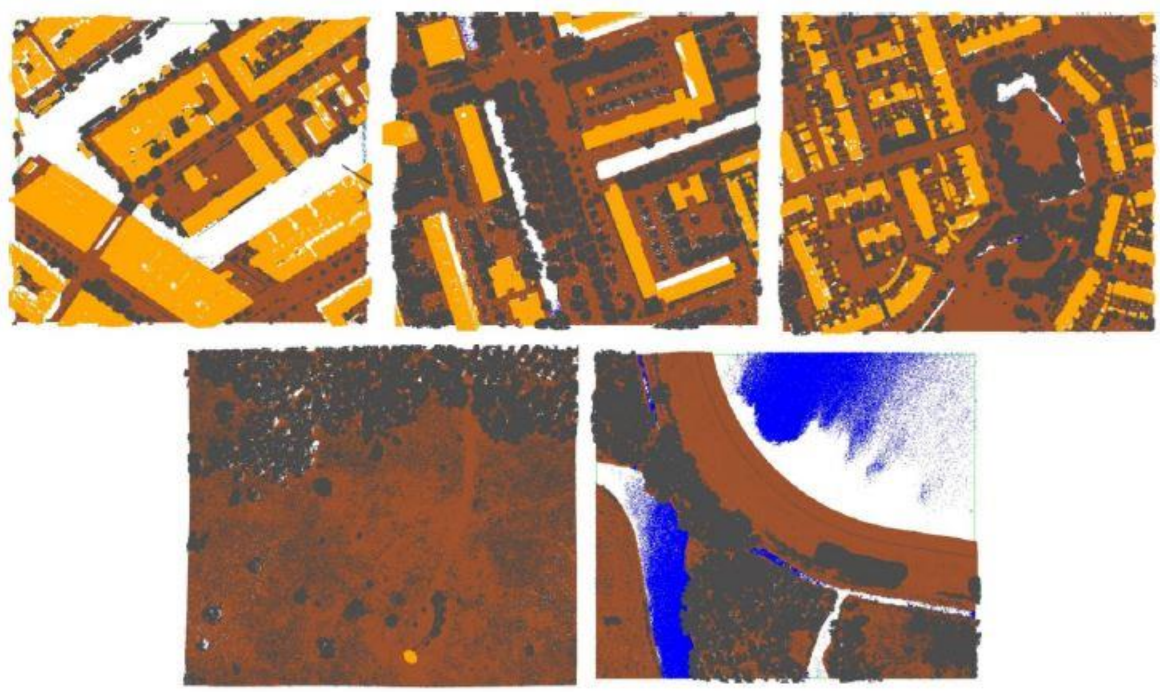


Figure #: Data samples: top-left Amsterdam, top-middle Delft, top-right Groningen, bottom-left National Park Veluwezoom and bottom-right National Park De Biesbosch

1.2 Ground filtering algorithms

Below we provide a brief description of how the ground filtering algorithms we used work.

<u>CloudCompare's Cloth Simulation Filter (CSF):</u>

According to the CSF algorithm, it is assumed that if we let a piece of cloth fall upon an upside-down terrain then the cloth will take the shape of the DTM and the shape of the DSM in the opposite case. Thus, with this algorithm, we extract the ground points from a point cloud by turning it upside down first and then dropping the cloth over it. Finally, the shape of the cloth can be determined and used as a base to classify the point cloud points into ground and non-ground points through an analysis made on the intersections between the points of the cloth and the point cloud (Zhang et al., 2016).

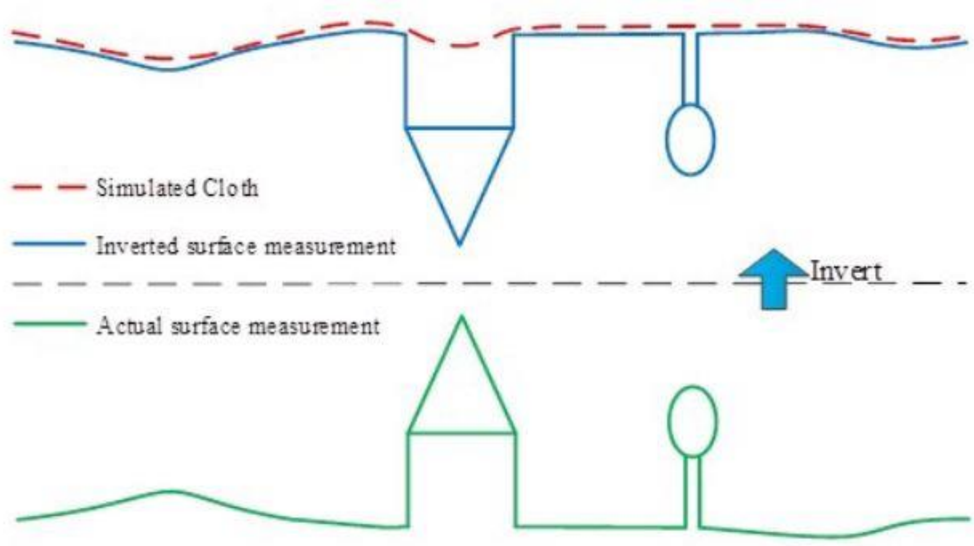


Figure #. Overview of the CSF algorithm (Zhang et al., 2016).

Zhang W, Qi J, Wan P, Wang H, Xie D, Wang X, Yan G. An Easy-to-Use Airborne LiDAR Data Filtering Method Based on Cloth Simulation. Remote Sensing. 2016; 8(6):501.

<u>PDAL's Simple Morphological Filter (SMRF):</u>

The workflow of the SMRF algorithm can be divided into four parts. Firstly, the minimum surface is generated by splitting the extent of the point cloud into cells and finding the lowest points within these cells. Then in the second part of the process, the cells that belong to bare earth are identified. This part is subdivided into four steps. The first step is to create a copy of the minimum surface. The second step is to create a vector of window sizes based on the supplied maximum which increases from one by one pixel to the ceiling of the maximum value divided by the cell size. The third step is to calculate an elevation threshold for every window size in the vector, create a new surface by applying a morphological opening on the minimum surface, add to the set of flagged ground cells any cell for which the difference between the minimum surface and the new surface we created is greater than the threshold we calculated and set the minimum surface equal to the new surface. The fourth and final step is to locate low outliers in the minimum surface. Moving on, the third part of the process is concerned with the creation of a provisional DEM by retaining the cells from the minimum surface that were identified as bare earth. Lastly, in the final part of the process, the points in the input point cloud are classified into ground points based on a required vertical distance parameter and an optional scaling parameter (Pingel, 2013).

Pingel, Thomas J., Keith C. Clarke, and William A. McBride. “An Improved Simple Morphological Filter for the Terrain Classification of Airborne LIDAR Data.” ISPRS Journal of Photogrammetry and Remote Sensing 77 (2013): 21–30.

PDAL’s Progressive Morphological Filter (PMF):

In a brief synopsis, the Progressive Morphological Filter is an iterative process that creates an initial filtered surface by applying an opening operation with a window of a specified length on the point cloud. Non-ground features that are smaller than the length of the window are removed and the in the next iteration the length of the window is increased. To overcome the problem that the filtering process tends to incorrectly remove measurements at the top of high-relief terrain elevation difference thresholds based on elevation variations of the terrain, buildings and trees are used.
The framework of the PMF algorithm can be divided into four steps. The first step is to superimpose a grid over the point cloud, select the minimum elevation in each cell of the grid and generate the minimum surface. For steps two and three an iterative process begins where we apply the progressive morphological filter to the grid surface. Initially, we have an input filtering window and in each iteration the filter is applied on the resulting surface of the previous iteration. The filtering window is increased using elevation difference thresholds for every iteration. Finally, the DTMs are generated by removing the non-ground points (Zhang et al., 2003).

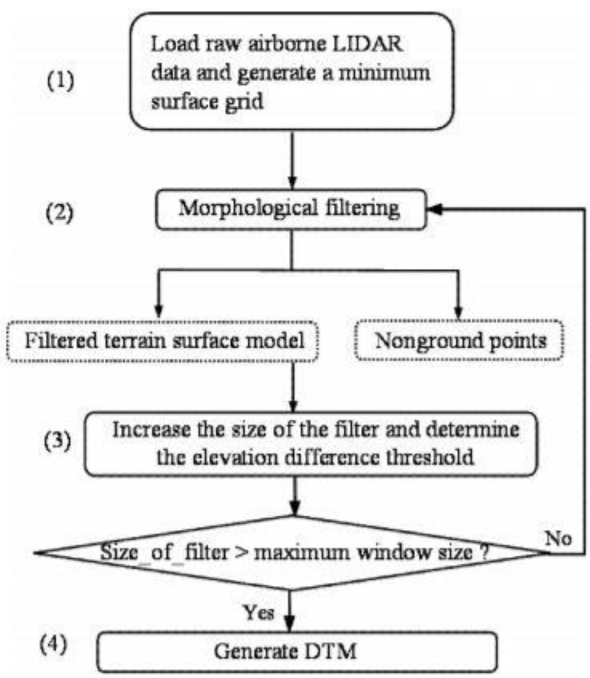


Figure #. Framework of the progressive morphological filter (Zhang et al., 2003).

Zhang, Keqi, et al. “A progressive morphological filter for removing nonground measurements from airborne LIDAR data.” Geoscience and Remote Sensing, IEEE Transactions on 41.4 (2003): 872-882.

1.3 Ground filtering results

The results of our assessment made us realize that we are left with two options:

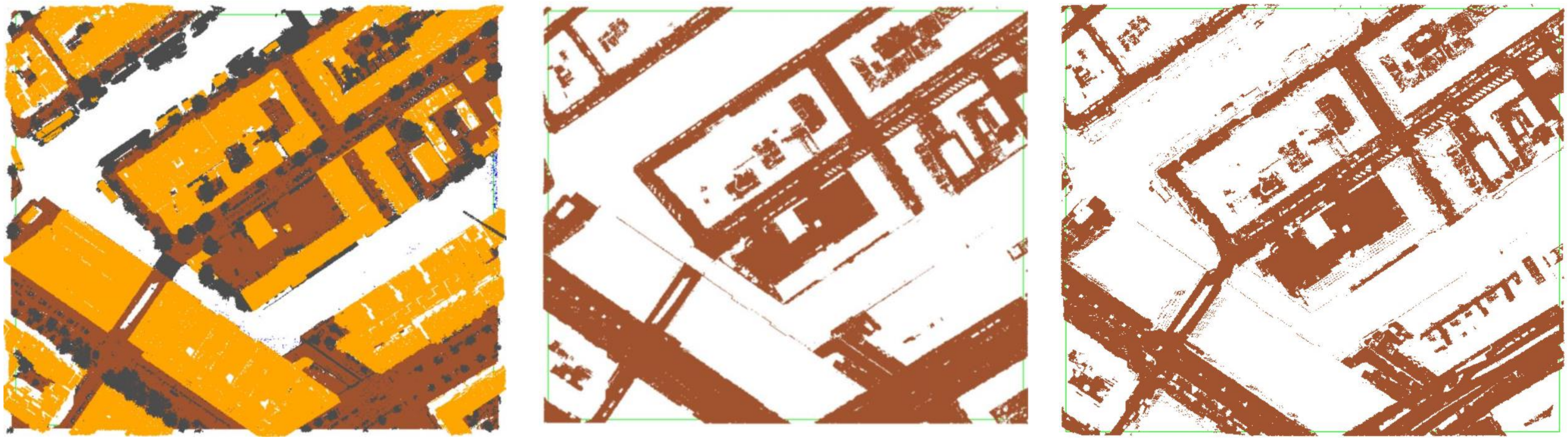
1. either we use the AHN classification which in general is reliable if we first apply some improvements like filtering out outliers and noise using PDAL.
2. or we apply a ground filtering algorithm using PDAL and make some use of the AHN classification along the way to remove, for example, water points from the whole process before applying the ground filtering algorithm.

After conducting our tests with the different available ground filtering algorithms, we decided that our best option is the first. The results show that the difference is not that significant to implement another ground filtering algorithm. The AHN classification has a really good quality and can be trusted. Thus, the final pipeline of our ground filtering process using PDAL is following one as shown in figure #.

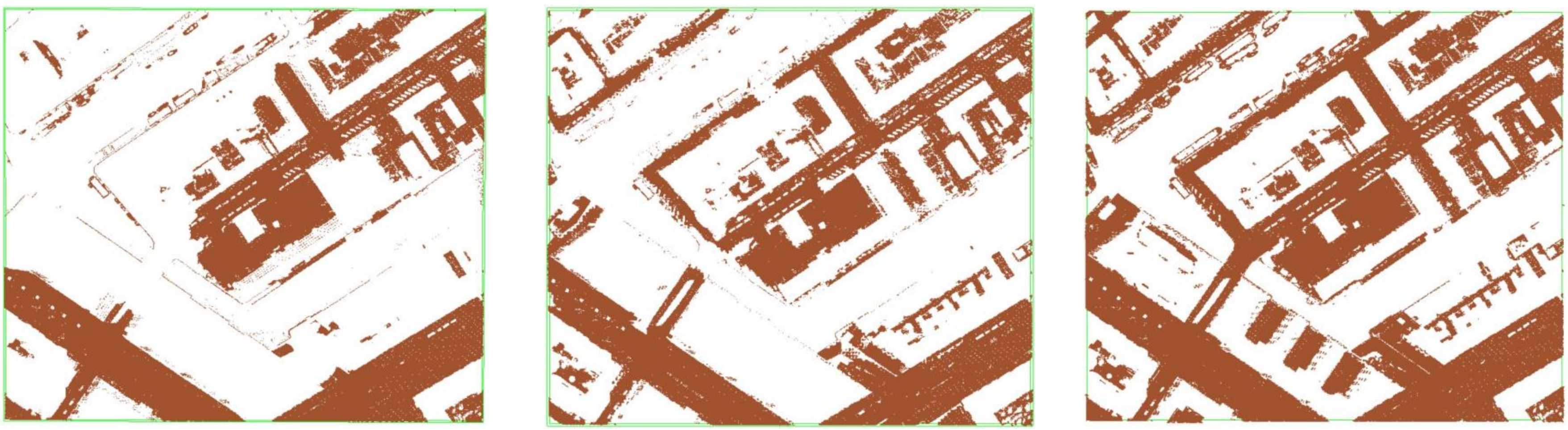
```
{
        "type":"filters.elm"
    },
    {
        "type":"filters.outlier"
    },
    {
        "type":"filters.range",
        "limits":"Classification[2:2]"
    },
```

Figure #. Final pipeline for ground filtering using PDAL. We keep the points classified as ground by the AHN classification and remove potential outliers and noise points.

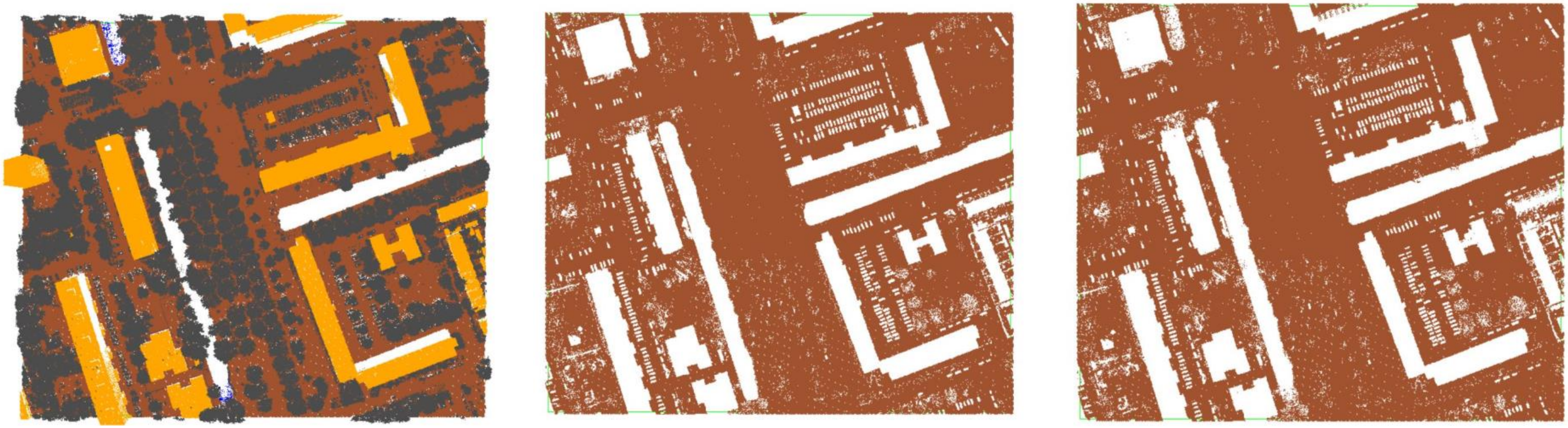
The two options do not differ a lot. Both approaches delivered quite similar results for most of the data samples we have. It is worth noting that the data sample of Amsterdam was the most problematic since it has more objects and street furniture are present as well as ships and uniquely shaped buildings. The results of the ground filtering algorithms for this sample have the most flaws. Therefore, to justify our final decision, provide a visual comparison of all the results, a table of comparison and we further discuss in more detail the flaws found in the results of the ground filtering algorithms for the data sample of Amsterdam.



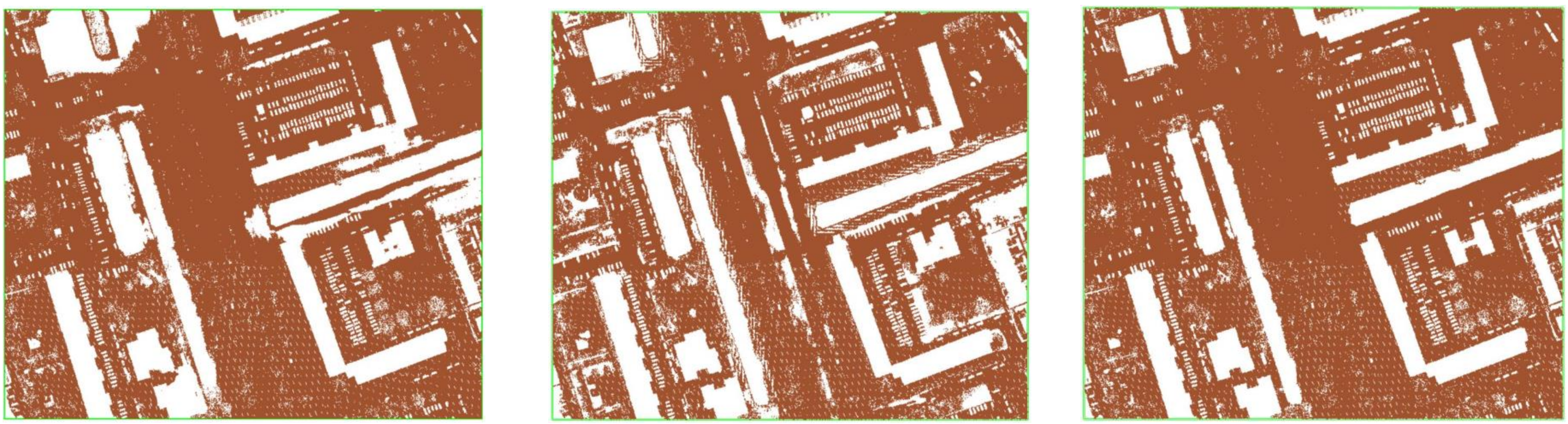
Amsterdam raw point cloud (left), AHN3 classification (center), lasground tool (right)



CSF algorithm classification (left), PDAL’s PMF (center), PDAL’s SMRF (right)



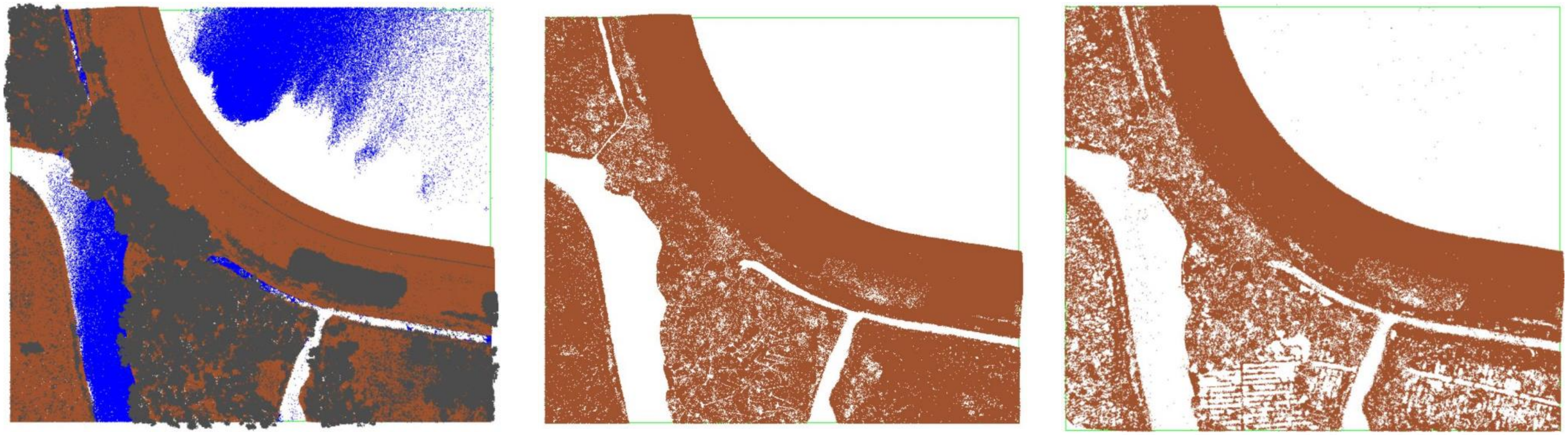
Delft raw point cloud (left), AHN3 classification (center), lasground tool (right)



CSF algorithm classification (left), PDAL’s PMF (center), PDAL’s SMRF (right)

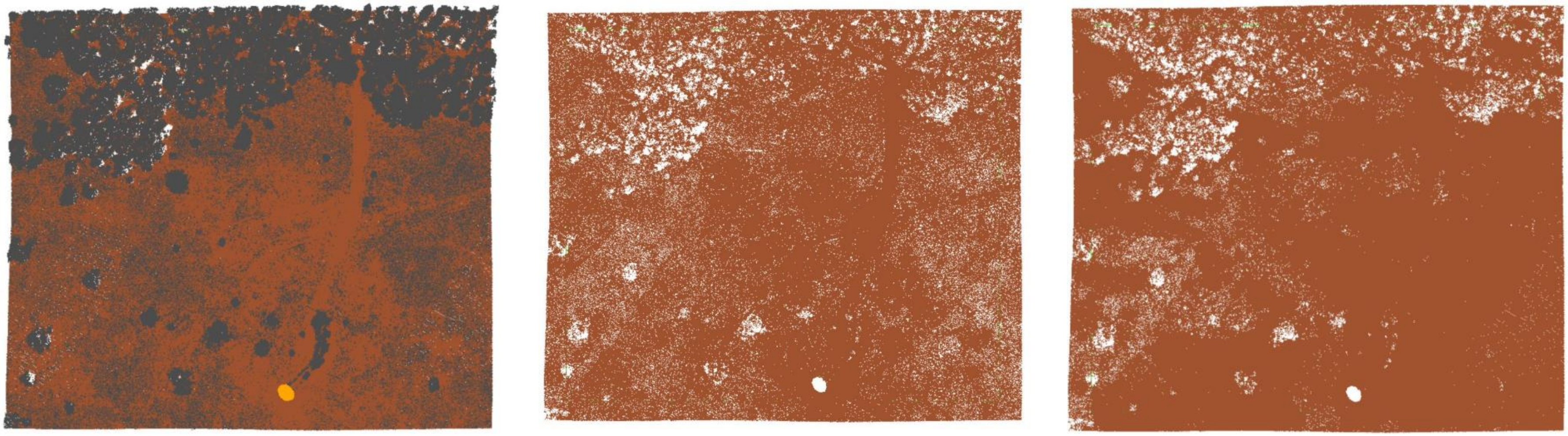
Groningen raw point cloud (left), AHN3 classification (center), lasground tool (right)

CSF algorithm classification (left), PDAL's PMF (center), PDAL's SMRF (right)

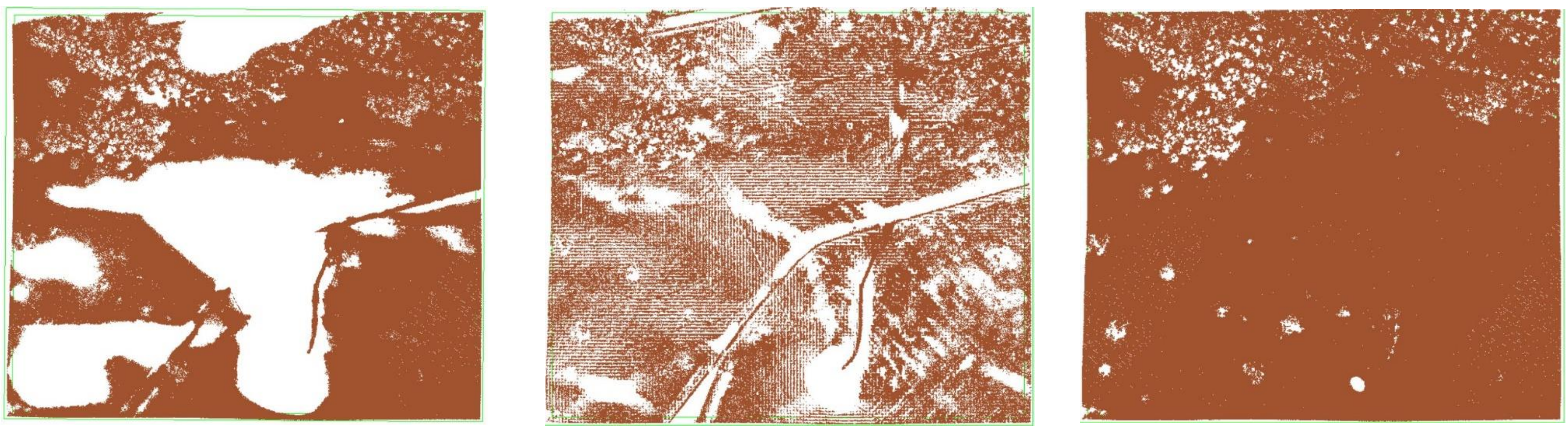


National Park De Biesbosch raw point cloud (left), AHN3 classification (center), lasground tool (right)

CSF algorithm classification (left), PDAL’s PMF (center), PDAL’s SMRF (right)



National Park Veluwezoom raw point cloud (left), AHN3 classification (center), lasground tool (right)



CSF algorithm classification (left), PDAL's PMF (center), PDAL's SMRF (right)

Location	Ground filtering algorithm	No. of ground points	Comments
Amsterdam	AHN3 classification	385651	--
	CloudCompare: Csf algorithm	303201	Less points are classified as ground, big holes are visible, the algorithm couldn't identify many ground points.
	LAStools: lasground	401356	More points are classified as ground but points that belong to boats and bridges are wrongly classified as ground.
	PDAL: filters.smrf	453767	Same as lasground tool but points from the roofs of some buildings are wrongly classified as ground as well (worse result).
	PDAL: filters.pmf	349188	Same artefacts as the smrf filter but with less points identified as ground.
Delft	AHN3 classification	614333	--
	CloudCompare: Csf algorithm	593126	The result looks the same but less points are identified as ground.
	LAStools: lasground	617281	The result looks the same but more points are identified as ground.
	PDAL: filters.smrf	628786	The result looks the same but more points are identified as ground.
	PDAL: filters.pmf	533611	Similar result as the smrf filter but with less points identified as ground.
Groningen	AHN3 classification	795751	--
	CloudCompare: Csf algorithm	713450	The result looks the same but less points are identified as ground. Some points in some canals are wrongly classified as ground.
	LAStools: lasground	838133	The result looks the same but more points are identified as ground. Some points in some canals are wrongly classified as ground.

	PDAL: filters.smrf	854810	The result looks the same but more points are identified as ground. Some points in some canals are wrongly classified as ground.
	PDAL: filters.pmf	699698	Less point identified as ground. There are large holes.
National Park Veluwezoom	AHN3 classification	817329	--
	CloudCompare: Csf algorithm	646117	Big holes are visible. Less points are classified as ground. The algorithm could not identify a lot of ground points.
	LAStools: lasground	945196	The result looks the same. More points have been classified as ground.
	PDAL: filters.smrf	1000126	The result looks the same. More points have been classified as ground.
	PDAL: filters.pmf	473013	Large holes, a lot of points missing.
National Park De Biesbosch	AHN3 classification	567210	--
	CloudCompare: Csf algorithm	367299	Big holes are visible. Less points are classified as ground. The algorithm could not identify a lot of ground points.
	LAStools: lasground	540870	A lot more points have been classified as ground. Water and trees points have been wrongly classified as ground. The result is better is they are removed but it has less ground points and many holes.
	PDAL: filters.smrf	627770	A lot more points have been classified as ground. Water points have been wrongly classified as ground. After removing them the result looks better but some points, probably tree points, are wrongly classified as ground and they are visible (inside the lake).
	PDAL: filters.pmf	243671	Large holes, a lot of points missing.

It is clear that the best results were given by PDAL's SMRF algorithm. Four out of the five cases had really good results compared to the AHN classification. A lot more points had been identified as ground for them but some small artefacts are visible even thought we used a filter for outliers after the ground filtering algorithm. The sample of Amsterdam had the worst result just like in all other cases. In The following figures we present some the artefacts we could not overcome without ignoring a lot of the points classes. Firstly, almost all water points in all the cases were classified as ground (Figure #). This challenge could be easily overcome by removing them from the whole process.

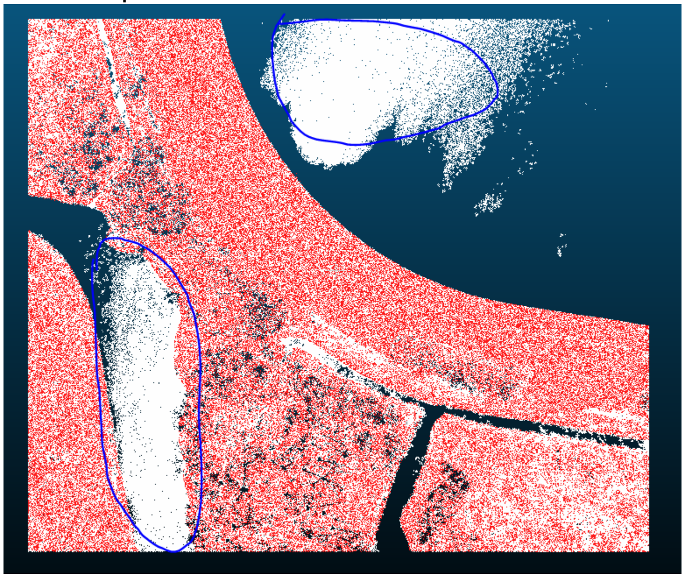


Figure #: White: PDAL's SMRF algorithm result. Water bodies are classified as ground, Red: AHN ground points.

The most difficult challenge that we are faced with is ships and boats in the canals of Amsterdam (Figure #). Removing them without ignoring some classes of points from the whole process was impassible. Their points belong in two different classes, unclassified and buildings, so removing those two classes is not the best approach as we would lose a lot of potential ground points. It is something that we could not avoid no matter what parameters we used and thus, it influenced a lot our final decision on the approach for the ground filtering part of our project.

One more challenge was that buildings points were classified as ground. Changing the parameters could lower the amount of building points that were misclassified as ground (Figure #). However, to completely overcome this challenge we had to remove all buildings points from the process which we considered not a good approach.

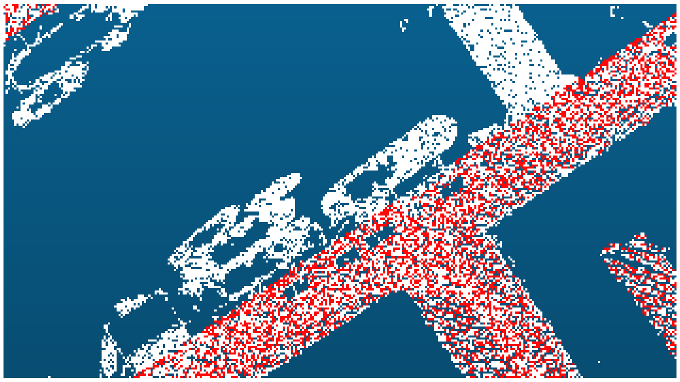


Figure #: White: PDAL's SMRF algorithm result. Boats classified as ground, Red: AHN ground points.

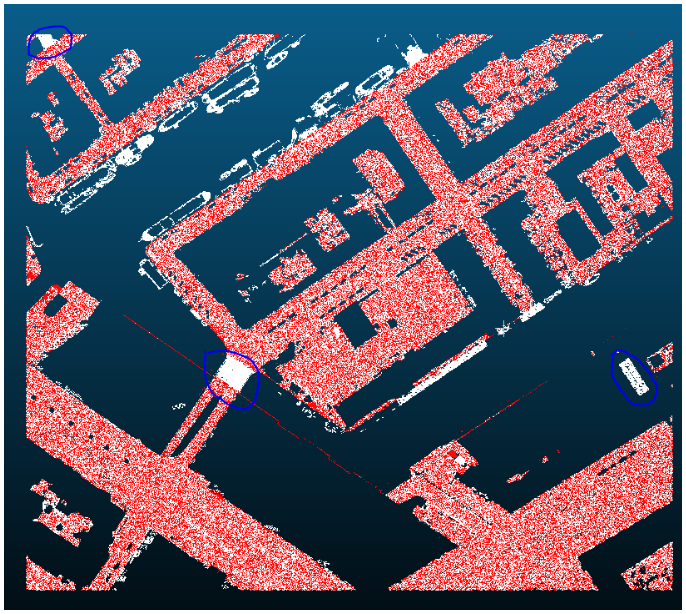


Figure #: White: PDAL's SMRF algorithm result. Red: AHN ground points. Overall result showing how an improvement was made (buildings artefacts removed) after many tries. However, ships, boats, buildings and bridges are still present.

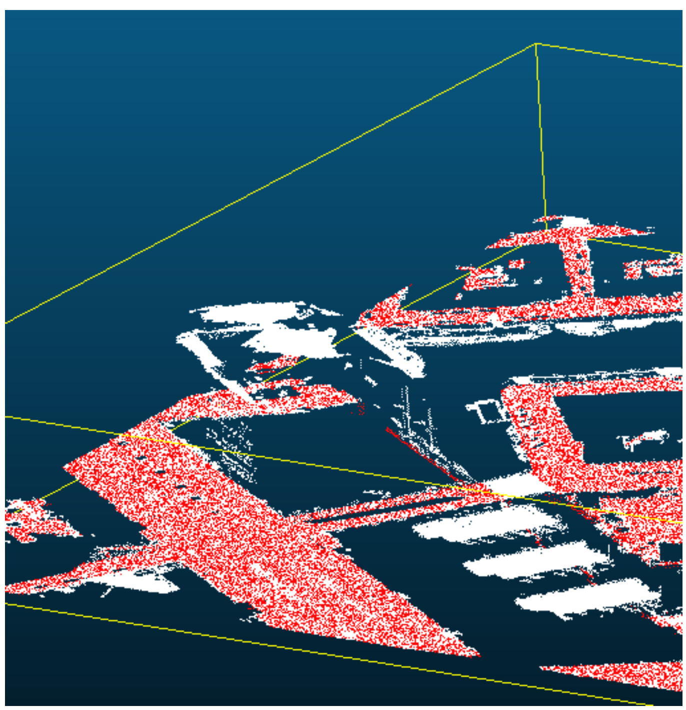


Figure #: White: PDAL's SMRF algorithm result. Buildings classified as ground, Red: AHN ground points.

Another challenge we were faced with is if we were going to consider bridges as ground points and include them in the interpolation algorithm for the generation of the DTM. The AHN does not classify them as ground. Instead it uses another code for them (26) and does not include them in the raster DTM (Figure #). Our final decision was to not consider them as ground points.

Figure #. AHN3 raw point cloud (left) and DTM raster (right). Bridges are not included in the DTM raster.

Lastly, it is also worth mentioning that in some cases we got much better results from the AHN classification like in the case of the National Park De Biesbosch (Figure #). However, using two different ground filtering algorithms to have the best results is difficult to implement and thus it was not proposed as a potential solution.

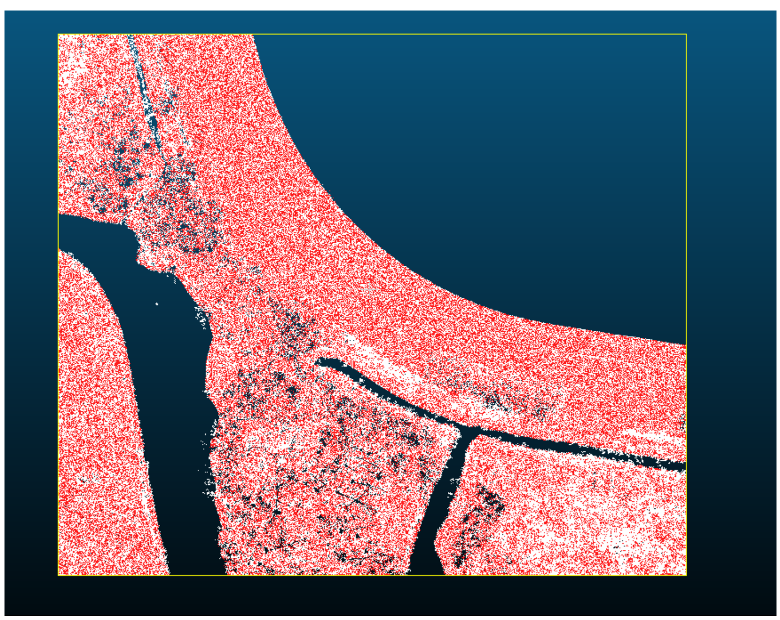


Figure 6: White: PDAL's SMRF algorithm result. Red: AHN ground points. 60,560 more points classified as ground and visually the result looks better.