International Journal of Geo-Information

MDPI

Article

Optimizing Wireless Sensor Network Installations by Visibility Analysis on 3D Point Clouds

Teresa Gracchi [1,2,*], Giovanni Gigli [1], François Noël [2], Michel Jaboyedoff [2], Claudia Madiai [3] and Nicola Casagli [1]

1 Department of Earth Sciences, University of Florence, Via Giorgio La Pira 4, 50121 Florence, Italy; giovanni.gigli@unifi.it (G.G.); nicola.casagli@unifi.it (N.C.)
2 Risk Analysis Group, Institute of Earth Sciences, University of Lausanne, Géopolis, 1015 Lausanne, Switzerland; francois.noel@unil.ch (F.N.); michel.jaboyedoff@unil.ch (M.J.)
3 Department of Civil and Environmental Engineering, University of Florence, Via di Santa Marta 3, 50139 Florence, Italy; claudia.madiai@unifi.it
* Correspondence: teresa.gracchi@unifi.it

Received: 29 July 2019; Accepted: 10 October 2019; Published: 16 October 2019

Abstract: In this paper, a MATLAB tool for the automatic detection of the best locations to install a wireless sensor network (WSN) is presented. The implemented code works directly on high-resolution 3D point clouds and aims to help in positioning sensors that are part of a network requiring inter-visibility, namely, a clear line of sight (LOS). Indeed, with the development of LiDAR and Structure from Motion technologies, there is an opportunity to directly use 3D point cloud data to perform visibility analyses. By doing so, many disadvantages of traditional modelling and analysis methods can be bypassed. The algorithm points out the optimal deployment of devices following mainly two criteria: inter-visibility (using a modified version of the Hidden Point Removal operator) and inter-distance. Furthermore, an option to prioritize significant areas is provided. The proposed method was first validated on an artificial 3D model, and then on a landslide 3D point cloud acquired from terrestrial laser scanning for the real positioning of an ultrawide-band WSN already installed in 2016. The comparison between collected data and data acquired by the WSN installed following traditional patterns has demonstrated its ability for the optimal deployment of a WSN requiring inter-visibility.

Keywords: wireless sensor networks; landslide monitoring; visibility analysis; 3D point clouds

1. Introduction

A wide variety of techniques aimed at monitoring the surface displacements of unstable slopes exist today. This reflects the need to assess the dynamics and possible evolution of landslides threatening vulnerable communities and infrastructures, so as to define appropriate remediation and early warning strategies [1]. Without getting into the details of all the currently available monitoring techniques, one of the most notable advancements was arguably produced by the development of ground-based and satellite radar interferometry [2]. However, devices offering extremely high accuracy and spatial coverage are typically associated with significant costs and a number of logistical constraints. For this reason, it is also important to refine practices and analysis methodologies related to monitoring techniques characterized by lower costs and higher flexibility, whose employment may be much more sustainable and equally efficient in certain scenario types. Some examples of this kind were provided by Dou et al. [3], Hemalatha et al. [4], Kromer et al. [5], and Prabha et al. [6].

In this framework, wireless sensor networks (WSNs) have recently attracted particular interest. These are defined as networks of usually low-size and low-cost devices denoted as nodes that are

integrated with sensors that can gather information through wireless links [7]. Their use presents several advantages over traditional monitoring techniques, such as the capability to work in rough and hardly accessible terrain, thanks to the absence of wired structures [8]; the possibility to analyze data from a multipoint perspective [9]; the ease and speed of installation [10]; and the adaptability of sensors to be integrated with existing monitoring techniques [11,12]. To make the most out of a WSN, optimal sensor deployment (i.e., equal and thoughtful distribution) is essential, and, to this aim, sensor-deployment problems were studied in several contexts [13]. For instance, Wang et al. [14] geometrically analyzed the relationship between coverage and connectivity in a WSN considering an obstacle-free environment, and Zou and Chakrabarty [15] proposed a virtual force algorithm as a sensor-deployment strategy after an initial random placement of sensors. For some types of sensors, line of sight (LOS) cannot be neglected. Information is in fact properly transmitted if the transmitter and receiver stations are in view of each other without any obstacle. The LOS of each emitting sensor is therefore an important issue to consider when positioning two or more receivers.

Researchers are increasingly focusing on LOS-based WSNs that use the same signal both for communication and ranging purposes, allowing the relative distance between nodes to be detected and landslide deformation to be assessed [16–19]. Depending mainly on their wavelength and the characteristics of the material to cross, waves can follow several types of behavior, and problems connected to unfavorable LOS may lead to the absence of data or inaccurate measurements due to a delay in arrival time or to multiple-reflection effects [20–23]. While several existing monitoring techniques work in LOS conditions (e.g., LiDAR [24,25], GB-InSAR [26], robotic total station [27]), this issue becomes more complicated when considering WSNs, as each node has to be in LOS with the others in order to ensure complete inter-visibility for the whole network. In this context, Intrieri et al. [28] tested a WSN named Wireless sensor network for Ground Instability Monitoring (Wi-GIM), which exploits ultrawide-band signals to evaluate ranging between nodes by means of the time of arrival. It was demonstrated how non-LOS conditions (mainly due to vegetation) could cause a delay in the signal and, consequently, errors in distance evaluation, as well as data loss, hence compromising the reliability of the monitoring system. For these reasons, inter-visibility analysis is essential to correctly place and install a LOS-based WSN.

Visibility analysis is a common issue in the fields of urban design, landscape planning, and geographic information system (GIS) technologies, and most solutions are based on the use of gridded digital terrain and elevation models [29,30]. With the development of LiDAR and Structure from Motion techniques, there is now an opportunity to directly use 3D point cloud data to perform visibility analyses. Doing so, many disadvantages of traditional modelling and analysis methods can be bypassed. For example, this makes it possible to avoid the generation of a surface model that is potentially associated with a number of errors and approximations [31–34]. Furthermore, point cloud data can provide much more detailed information than traditional gridded raster-terrain data or triangulated mesh-terrain data, especially on steep, hummocky, vegetated, and/or highly irregular terrain.

In this paper, an algorithm able to find optimal locations to install sensors, part of a LOS-based WSN, is presented. This was attained by following three criteria: (i) inter-visibility between sensors was guaranteed by exploiting a modified version of the Hidden Point Removal operator by Katz et al. [33]; (ii) equal distribution of devices was ensured; and (iii) different weights were preferentially assigned to critical portions of the investigated landslide (e.g., areas more at risk, areas showing higher displacement rates, and easily accessible or stable areas). To check the reliability of the proposed algorithm, it was applied to two different datasets. The first set corresponded to a complex artificial scene representing a room with scattered vertical obstacles in the form of beams and sections of walls designed in AutoCAD 3D (Autodesk, v. 20.0) and transformed into a 3D point cloud with CloudCompare (v. 2.10). This was a challenging scene because of the difficulty of finding lines of sight in crowded obstacle scenes. In addition, obstacles with simple shapes made the scene easy to visually interpret, making it a good first example to understanding the functioning of the code. The second dataset corresponded to a real site, namely, the Roncovetro landslide, located in the Emilia

Romagna Region, Central Italy. At the same site, a first monitoring campaign with the system was carried out in 2016 by Intrieri et al. [28]. It was then possible to compare the data acquired in the two campaigns and evaluate the performance of the proposed method in a real serviceable case. The final aim of this work was to quantify the improvement that the introduced method produces in a wireless sensor network, requiring free LOS.

2. Proposed Method

This paper proposes an algorithm called Wireless Sensor network Installation Optimizer (WiSIO) to geometrically find the best locations to install a WSN requiring visibility, i.e., the i^{th} Point (P_i). To pursue this goal, WiSIO was implemented in MATLAB (R2018a) as simple, fast, and easy code. It can be applied to several natural scenarios, such as vertical cliffs, gentle and steep slopes, and terraces. Its employment can also be extended to buildings, small objects, and, more generally, to every artificial structure without geometrical or size limits.

The central point of the algorithm is visibility analysis. Similar analyses are the common function of most geographic information system software, but, in all cases, they are based on the digital elevation model. This kind of analysis uses the elevation of each cell of the model to detect if they are visible or not from a given point of view. Line of sight from the point of view to the target cells is substantially planned, and when cells of higher elevation intersect them, the view is considered obstructed. It goes without saying that the method works well when dealing with straight lines, as is the case of buildings and roads, and this is why these viewshed analyses are usually made on the digital elevation model and not on the digital surface model. Dealing with vegetation, in fact, constrains these methods to significant geometrical approximations, even when models have good resolution. With the aim of keeping all original information, the input data of the proposed algorithm are high-resolution 3D point clouds. Working directly with point clouds, as in the case of WiSIO, aims to be able to work with harsh environments, including not only vegetation but also steep, rough, and overhanging terrain. The algorithm works without reconstructing any kind of surface, avoiding errors and approximations usually caused by traditional modelling. The heterogeneity and the resolution of 3D point clouds are consequently main limitations of the developed code: The lower the resolution of the input data is, the less reliable the output is. In this regard, input data should be a cloud obtained by merging data acquired from different positions to avoid shadow areas and to have regular distribution of points in space.

To appreciate the importance of working with 3D point clouds instead of the digital surface model for viewshed detection, a comparison between the two processes made on the same dataset was carried out (Figure 1). In the one case, the Viewshed 3D analyst tool of ArcGIS was applied on a digital surface model with a resolution of 1 cm; in the other, WiSIO was run on the 3D point cloud (1 cm resolution), obtained by photogrammetry and the source of the abovementioned surface model. Figure 1 shows the results in detail, allowing for an easy comparison. It is evident that working with points instead of surfaces produces more reliable results.

Given a landslide's 3D point cloud, only the points respecting the following criteria were considered as the best locations (Figure 2):

1. Inter-visibility—each selected point is in LOS with the others;
2. Inter-distance—nodes are equally distributed in space;
3. When the previous conditions are satisfied, it is preferable if Pi is positioned into preselected areas, hence Priority Areas (PAs).

Once these principles are met, a WSN is in its best working condition, which translates into an accurate landslide deformation map and an efficient real-time monitoring system to achieve the following:

1. Have numerous and reliable data, since complete inter-visibility is guaranteed for the whole network;
2. Obtain a reliable and complete deformation map;

3. Constrain the deployment of the WSN in areas preselected by the user.

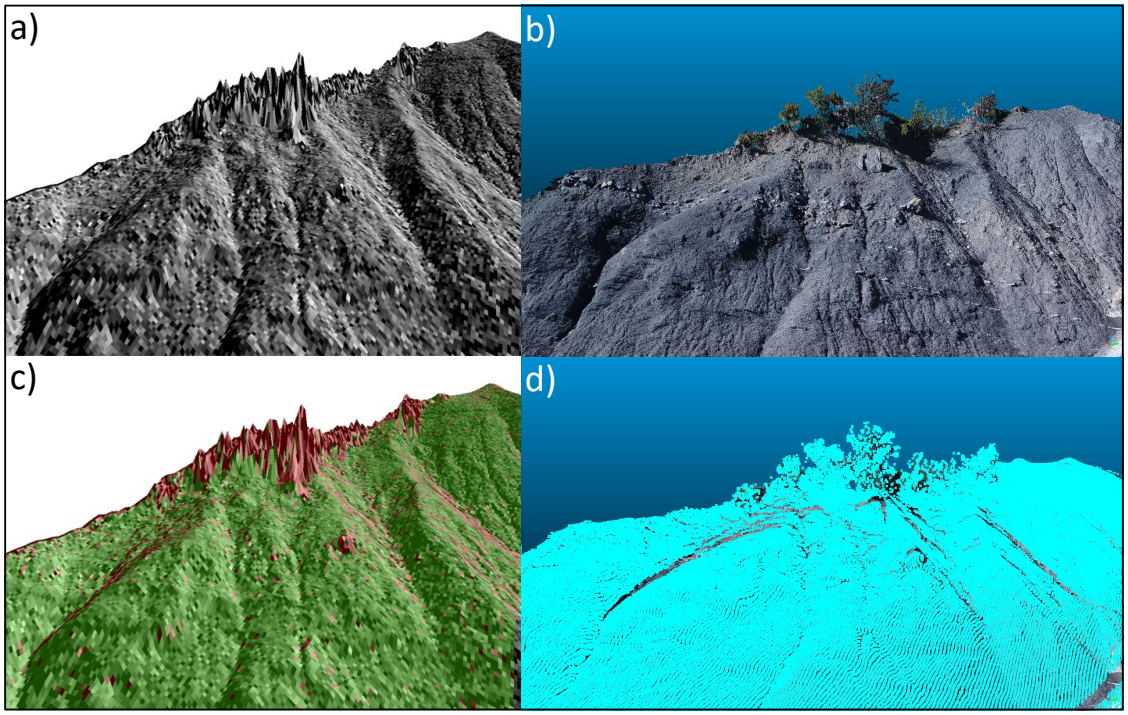

Figure 1. Comparison between viewshed analysis on a digital surface model and Wireless Sensor network Installation Optimizer (WiSIO) on a 3D point cloud using a particular of a landslide's 3D model and point cloud. For better comparison, both analyses were carried out, considering the same point of view. (**a**) Hillshade of digital surface model with 1 cm resolution. (**b**) Three-dimensional point cloud with 1 cm resolution. (**c**) Application of ArcGIS 3DAnalyst Viewshed applied on digital surface model. Visible and nonvisible areas are marked, respectively, in green and red. (**d**) Application of visibility analysis carried out in WiSIO on the 3D point cloud. Visible points are highlighted in cyan.

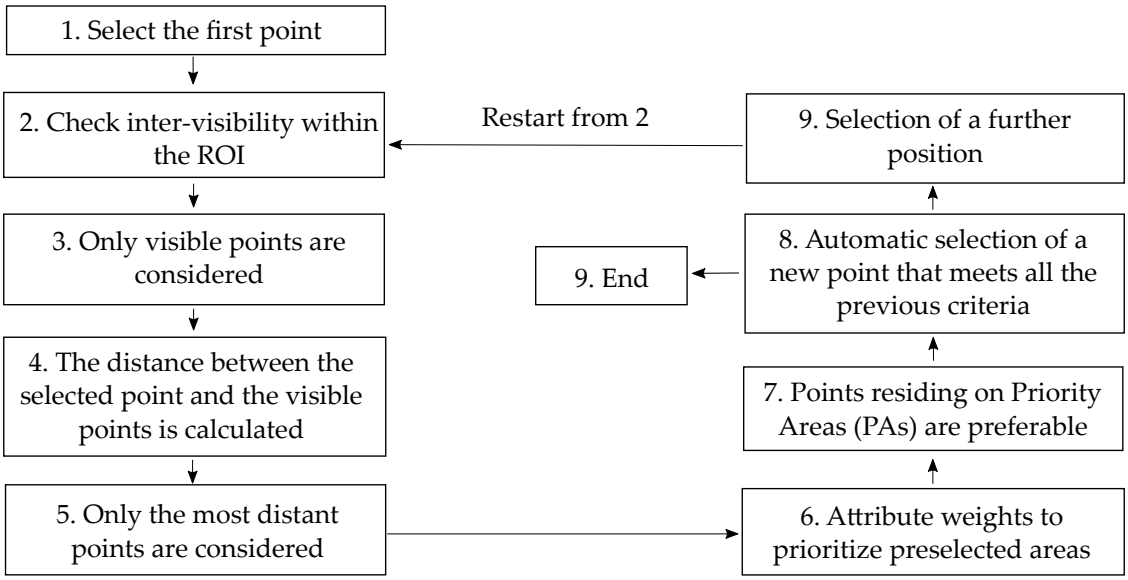

Figure 2. WiSIO operating scheme. Selection of the first point occurs in a semiautomatic way. Depending on the quantity of devices the user wants to install, once a new P_i is pointed out (Box 8), the code ends or restarts from Box 2.

To avoid the involvement of useless data and the consequent increase of elaboration time, the algorithm automatically calculates a region of interest (ROI) for each node. Only points residing inside this region are then considered for subsequent elaborations. The ROI extension depends on the landslide geometry and on the monitoring system used, but it generally coincides with the maximum distance at which two sensors can communicate properly (D_{max}). Given C, the point of view of X, Y, and Z coordinates, the region is defined by a cube whose farthest points are ($X - D_{max}$; $X + D_{max}$; $Y + D_{max}$; $Y - D_{max}$; $Z + D_{max}$; $Z - D_{max}$). The best positions are thus selected automatically, except for the first one, which is chosen in a semiautomatic way. The user manually selects by clicking on the point cloud with pointer P_1. The algorithm automatically calculates a second ROI (hence ROI_{P1}), smaller than the previous one, inside which the first point is chosen. The choice is made by means of an iterative procedure, during which n P_1 within ROI_{P1} (i.e., the region of interest around P_1) are selected randomly. For each of them, visibility analysis is computed. As a result, the point from which more points are visible is selected as the first point. This helps in positioning P_1 close to the desired area while having a high visibility. The choice of P_1 with low visibility could indeed compromise the results.

2.1. Visibility Analysis

To face the problem of inter-visibility between nodes, the Hidden Point Removal operator proposed by Katz et al. (2007) was applied. The operator is a simple algorithm to determine visibility in point clouds without performing surface reconstruction. It first transforms the points of the cloud by means of spherical flipping inversion, and then computes the convex hull of the set containing the viewpoint and the transformed points [35]. A point is then marked visible from a given point of view if its inverted point lies on the convex hull (Figure 3) [33]. The operator is easy and fast, and it gives back points that are actually visible. Straightforward solutions are in fact bound to fail, and calculating the LOS between two point is not helpful since, except for extreme cases, a point is always visible, even with high-resolution point clouds. The method has found numerous applications in several domains, is supported by theoretical guarantees, and was explained in depth by Katz et al. [33–36].

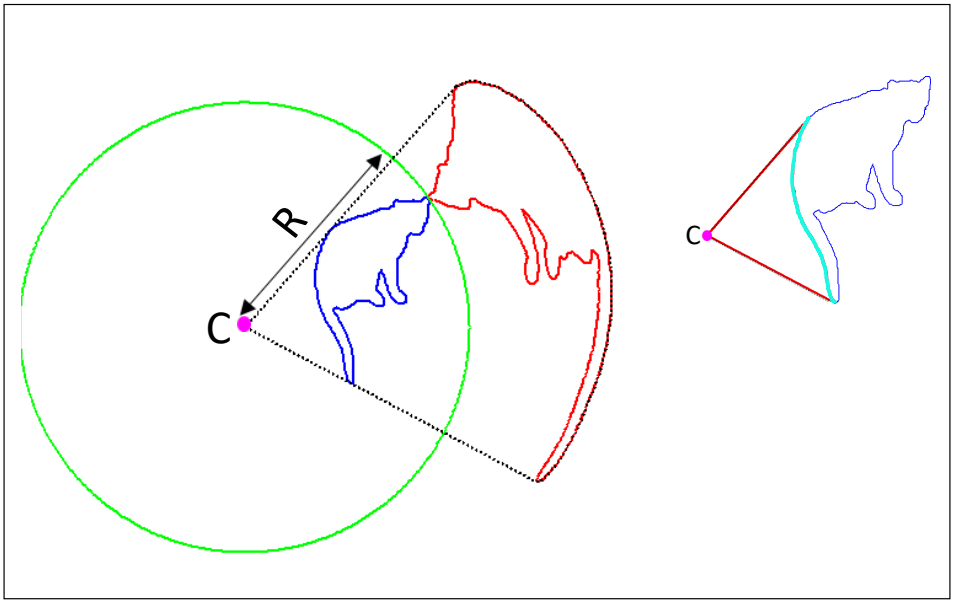

Figure 3. Hidden Point Removal operator. Blue, investigated point cloud. Green, sphere with radius R used to invert points by spherical flipping (left side). Points residing on convex hull of inverted points (cyan at top-right) are marked as visible from C (point of view; modified after Katz et al. [33]).

R, the radius of the sphere for spherical flipping, is the only parameter to set. As it increases, more points pass the threshold of the convex hull and are hence market visible [33]. Consequently, a large

radius is recommended for dense point clouds and the other way around. It is evaluated by an iterative procedure that considers two points of view at opposite sides, with respect to the object center. R is then defined as the value that maximizes the total number of disjoint visible points [35].

The Hidden Point Removal operator was developed for little objects like small sculptures. In the proposed method, dealing with huge point clouds, R is not constant but changes by increasing the distance from C (point of view) to the target, and it is not only a function of point density (homogeneous in ideal cases). Indeed, it also depends on the amount of points residing within a cuboid centered on C and having a half diagonal (D/2) equal to the distance from C to target. As shown in Figure 4d, given a point cloud (P), the number of involved points obviously grows with distance. The higher D/2 is, the higher the number of points is. In this sense, to correctly apply the operator, the distance has the same role as density: For points closer to C, a small R is recommended, while for farther points, a larger R is more suitable, as demonstrated in Figure 4, where the same R is not optimal for both closer and the farthest points. Too-high R values produce false positives (i.e., nonvisible points marked visible, Figure 4b), and too-low values originate false negatives (i.e., visible points marked as nonvisible, Figure 4a). Choosing a medium value for R is not a good solution since it involves a high number of false positives for short distances and false negatives for longer distances. To solve this problem, the point cloud was split in s cubic sectors centered in C, and R was evaluated for each of them. As a result, optimal R increases by moving away from C (Figure 4d). If the point cloud is divided in s sectors, visibility analysis is computed s time. For each of them, all points between C and sector s (included) were considered, but only the ones that are within sector s are marked as visible or nonvisible. Obviously, the more the sectors in which P is split there are, the more accurate the visibility-analysis results are and the higher the running time is.

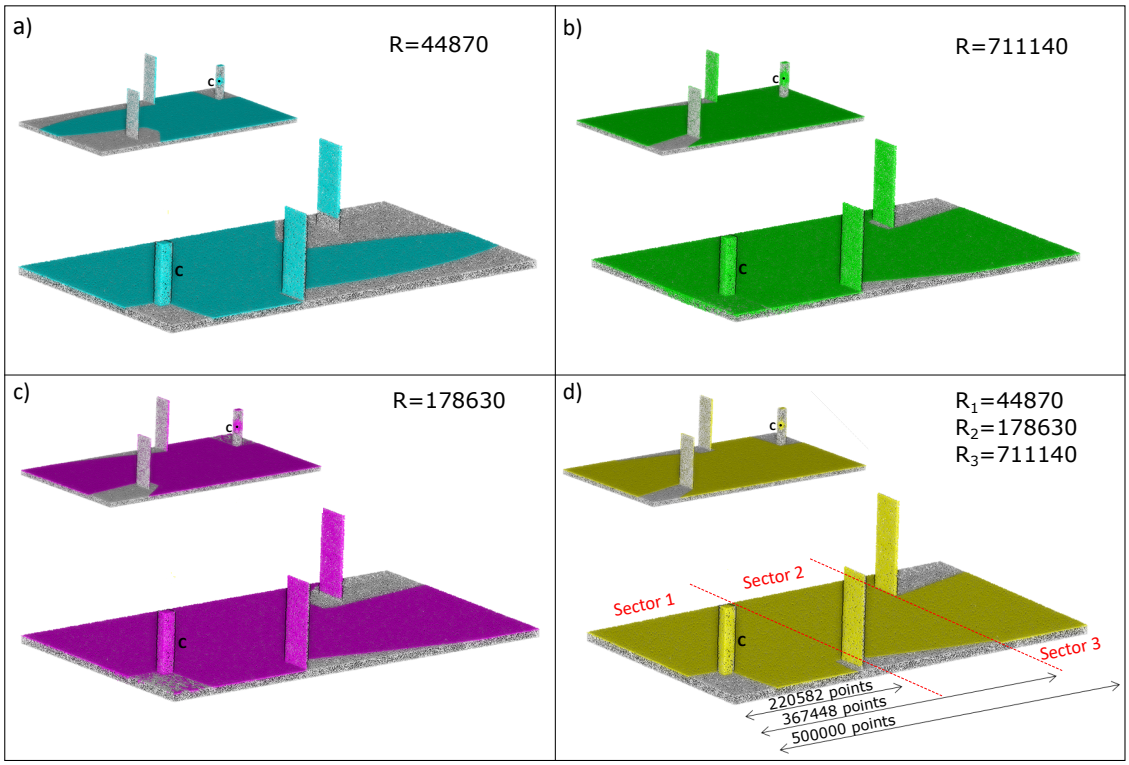

Figure 4. Test to evaluate optimal R. (**a**–**c**) cyan, green, magenta, and yellow, points marked visible from C changing the radius. (**d**) Use of different Rs for each sector.

To reach the goal of the present work, the Hidden Point Removal operator was applied to a landslide 3D point cloud. Since monitoring sensors (regardless of the type) generally lies on the ground, it is necessary to split the original point cloud in terrain and vegetation point clouds, to avoid the placing of sensors on points representing vegetation. Indeed, in the landscape field, a major

obstacle in terms of visibility is vegetation, and monitoring devices are usually installed on the terrain. Therefore, the operator is applied to the whole point cloud, but only terrain points are considered as points where the network could be installed. For distance calculation and computation concerning preselected areas, the point cloud terrain is also the required datum. Vegetation is only involved in visibility analysis. As a result, providing as inputs the point cloud vegetation (PCV), the point cloud terrain (PCT), and the point-of-view coordinates (C), the algorithm returns visible PCT points from C. Furthermore, in some cases, sensors are not directly touching the ground, but they are fixed on supports (e.g., poles and tripods). The proposed algorithm gives the possibility to consider the height of the support, placing points where sensors would effectively be placed; that is fundamental for visibility analysis.

Given *h*, the support's height, visibility analysis was computed once h was added to the P_i Z component. If installation on *n* nodes was required, the Hidden Point Removal operator was applied for each of them for all sectors. Since complete network inter-visibility is required, one can note that node locations are selected between points that are visible from P_1 to $P_{(i-1)}$, e.g., the choice of P_4 is between points that are visible from P_1, P_2, and P_3.

It is worth noting that obtained visible points lie on the ground, while, in the field, sensors must be in LOS with other sensors, raised in the turn of *h*. The assumption was that a point raised by *h* was visible from a given point of view if the same point lying on the ground was visible. Problems come in harsh situations where there is a lot of dense vegetation, such as tall grass, and a minimal amount of terrain points is visible from C. In this case, the algorithm could underestimate the visible points from C, and it could be necessary to ask the algorithm which terrain points would be visible if they were raised by *h*. To solve this problem, it should be necessary to raise each PCT point (point cloud terrain), one by one, verifying if they lie on the point cloud convex hull. This means that, if the PCT is made by one million points, for each point of view, the operator should be applied one million times: Each time a different PCT point should be raised by *h*, visibility should be computed and the same point should be lowered by *h*. This version could be the best solution in areas with low visibility, as well as in areas with not many suitable points to install a device, but it obviously takes too much elaboration time, especially when dealing with big clouds. For this reason, considering the size of the employed point cloud, the authors decided to test the algorithm rising only P_i by *h*.

2.2. Distribution

For the geometrical deployment of the sensors, with the aim to have equal distribution in space, the Euclidean distance with already selected ones for each P_i was computed. Only the farthest points were then considered to avoid two or more close monitoring devices or areas with no sensors at all. The procedure was carried out by associating weights to node inter-distances.

Given $X_{(i-1)}$, $Y_{(i-1)}$, and $Z_{(i-1)}$, the coordinates of $P_{(i-1)}$, and X_i, Y_i, and Z_i, the coordinates of P_i,

$$Distance = \sqrt{\left(X_{(i-1)} - X_i\right)^2 + \left(Y_{(i-1)} - Y_i\right)^2 + \left(Z_{(i-1)} - Z_i\right)^2} \quad (1)$$

P_i is selected among the farthest points of the PCT from the already chosen points. The remoteness is evaluated by calculating the product of all the distance obtained with Equation (1), between $P_1, \ldots .$, P_{i-1} and all the points of the PCT.

2.3. Priority Areas (PAs)

A step that allows the user to prioritize sensors' deployment in selected portions of the point cloud was added to the algorithm. It aims to give different levels of relevance to distinct areas of a landslide, avoiding the absence of monitoring devices in significant areas. The boundary of these sections (priority areas, PAs) is made manually by the user, and a file with the XYZ coordinates of points residing inside these areas must be uploaded as an input.

In detail, the m farthest points obtained from the distance evaluation (Equation (1)) are pointed out. If between them there are some points residing in PAs, then P_i is selected among those ones. If there is no intersection between the farthest points of the PCT (point cloud terrain) and PAs, then P_i is the farthest point of the PCT from $P_1, \ldots, P_{i-1}$. It goes without saying that the choice of m depends on how important the inter-distance is when prioritizing these areas. This makes sure that the user's experience and knowledge play an important role in the procedure, despite the automatic nature of the tool. Prioritized areas can be areas closer to element at risk, region that have higher displacement values, the most accessible zones, the most stable points, or any areas where the user would like to place devices.

3. Results

To check the reliability of the proposed algorithm, and with the aim to make the function of the code easy to understand step by step, a 3D model was produced and subsequently transformed into a 3D point cloud. The model was built to have low visibility and to highlight the WiSIO ability in positioning a network, ensuring complete inter-visibility. Furthermore, it allows users to easily perceive visibility from a given point of view (Figure 5). The model was designed on AutoCAD 3D (Autodesk, v. 20.0) and transformed into a 3D point cloud (made by approximately two million points) with CloudCompare (v. 2.10). First, validation was done by selecting generic point of view C, evaluating visible points from C, and exploiting the camera/eye center visualization function of the CloudCompare software (Figure 5). As shown in Figure 5d, all points detected as visible from WiSIO were indeed visible from C (in cyan), since there was no overlap of points. On the other hand, a small number of false negatives were reckoned up due to the precautionary R chosen for the test. The problem of false negatives can easily be solved by increasing the number of sectors. The higher the number of sectors is, the lower the number of false negatives and the higher the running time, as shown in Table 1, where data of three different approaches are reported. Note that Test 1 is the approach whose results are shown in Figure 5.

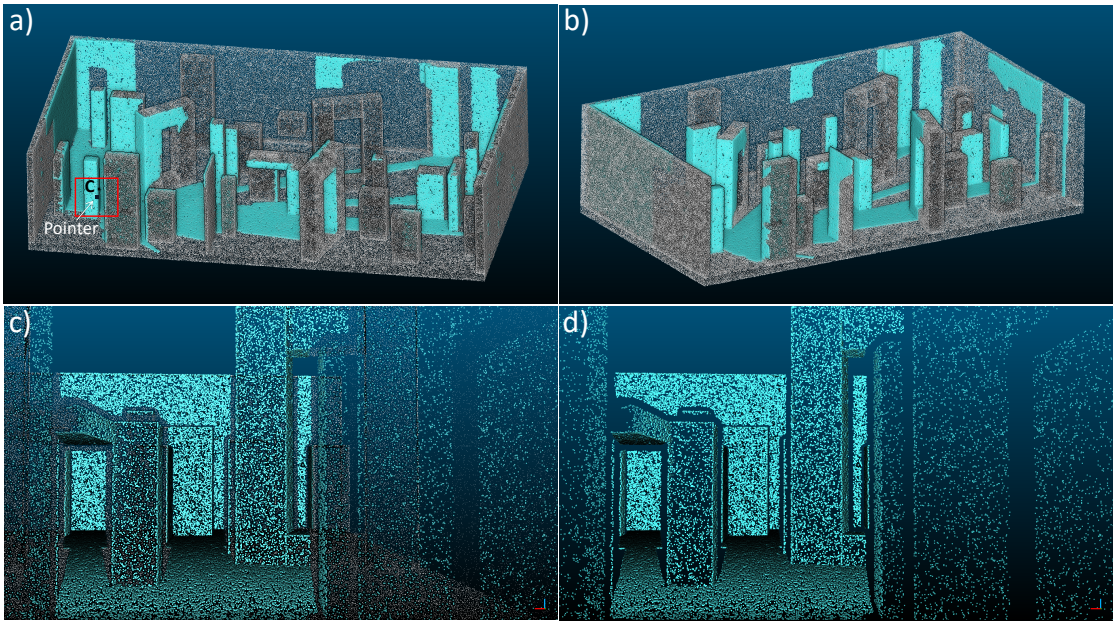

Figure 5. Three-dimensional model developed to simulate a harsh environment in terms of visibility. (**a**) Visible points from C highlighted in cyan. Dark-blue dot, point selected with pointer as first choice; red box, small region of interest (ROI_c) inside which C was semi-automatically detected. (**b**) Visible points from C highlighted in cyan from different angulation; (**c**) view from C of points marked visible (in cyan) and whole point cloud (grey points); and (**d**) view from C of only points marked visible.

Table 1. Number of false negatives obtained with various sectors. Test 1 results are shown in Figure 5.

	Radius, R (m)						False Negatives	Running Time
Test 1	Sector 1 44,568	Sector 2 354,019	Sector 3 1,119,508	Sector 4 1,774,301	Sector 5 4,456,843		10,504	(min) 0.350
Test 2	Sector 1 44,568	Sector 2 354,019	Sector 3 706,362	Sector 4 2,812,078			25,120	(min) 0.317
Test 3	Sector 1 44,568	Sector 2 223,371	Sector 3 706,362	Sector 4 2,233,713	Sector 5 4,456,843	Sector 6 5,610,833	4131	(min) 0.393

Only one generic point cloud P, made by 1,972,852 points, was used, since no vegetation was present in the model. Figure 6 illustrates the whole elaboration made by WiSIO to point out seven node positions, carried out in 3.33 minutes. Walls were removed to better display all the involved P_i. To detect P_1, the iterative procedure was repeated three times. The automatically evaluated ROI around P_1 (ROI_{P1}) is depicted as a black box in Figure 6a. Moreover, some objects were randomly selected in the model and uploaded as PAs (in red in Figure 6e,f).

Five sectors were used whose respective radius are those reported in Table 1, Test 1. The test was also repeated, changing the quantity of points to evaluate the amount of time required by varying the size of the point cloud (Table 2). It is worth noting that P_3, P_4, and P_6, were outside priority areas since no farther points overlapped with them (Figure 6e). All selected points were in LOS with respect to the others and equally distributed in space, and four of them resided in areas marked as Pas, despite the designed harsh environment (Figure 6f).

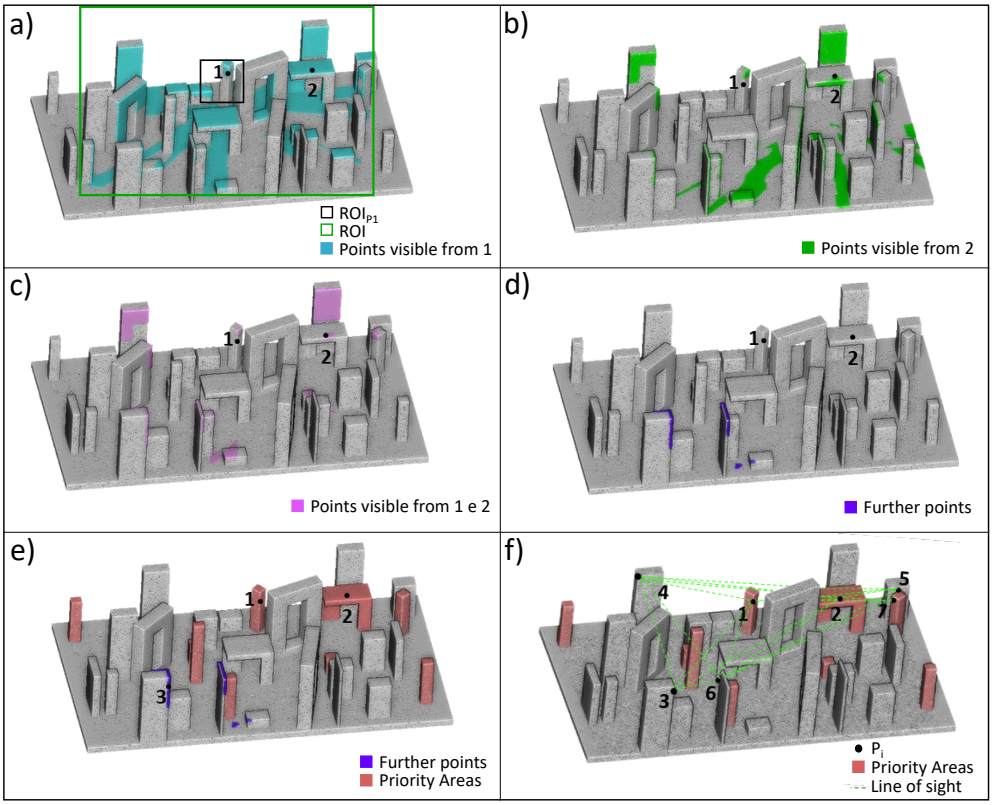

Figure 6. Results indicated by WiSIO for the installation of seven nodes using the developed 3D model. (**a**–**e**) All steps computed by WiSIO for automatic detection of P_3 and (**f**) final result with the positioning of the seven points.

Table 2. Different running times for different point numbers.

Point Numbers	Running Time
427,920	0.85 min
1,972,852	3.33 min
2,000,000	3.61 min

Field Application

The method was applied in the field for the installation of the Wi-GIM system, a Wireless Sensor Network measuring node inter-distance by means of ultrawide-band technology. For more details about the system, please refer to Mucchi et al. [18] and Intrieri et al. [28]. The devices worked properly in LOS conditions; thus, aimed deployment of the sensors could widely improve their efficiency. For better evaluation of the improvement, Wi-GIM was installed in the Roncovetro landslide, as previously done by Intrieri et al. [28]. During this first monitoring period, system accuracy and precision were analyzed. While precision ranged from 2 to 5 cm, the medium recorded error of the distance was about 75 cm, and it was strictly connected to both the presence of obstacles and the low-cost nature of the system [28]. However, this large error did not compromise the effectiveness of the system since it focused on relative distances and not absolute ones. The slide is a 2 km long mudflow located in the Emilia Romagna Region, Central Italy (Figure 7). The site was particularly suitable for the application of WiSIO due to its dimensions, the amount of vegetation, and the presence of geomorphological highs and lows that make it harsh even in terms of visibility.

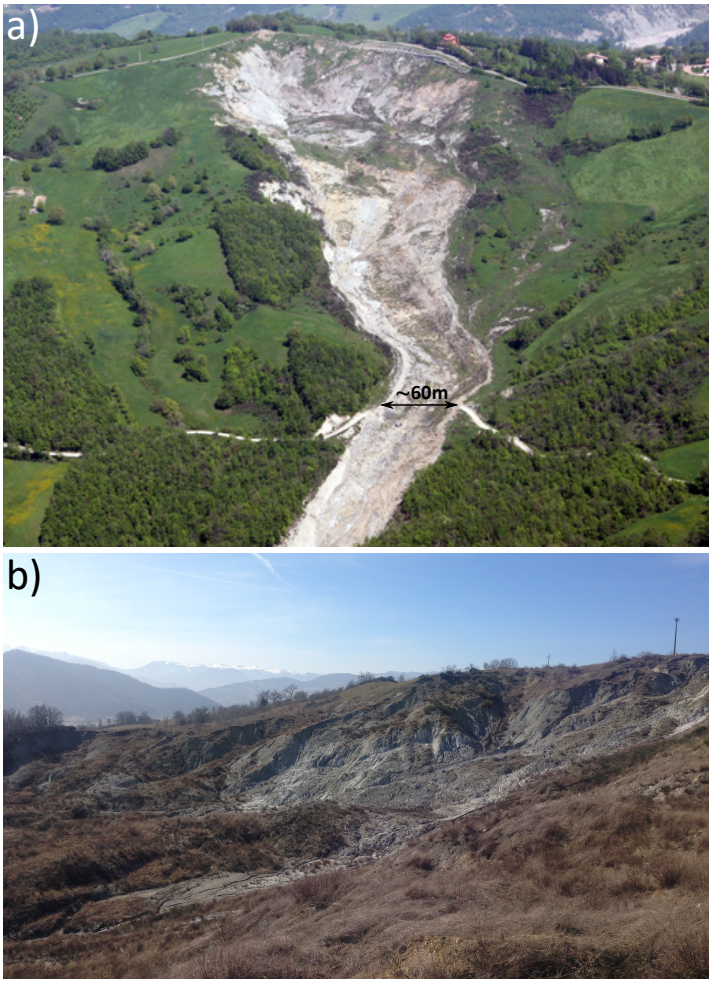

Figure 7. Roncovetro landslide. (**a**) Aerial image of mudflow and (**b**) detail of landslide's crown.

The 3D point cloud was obtained with a terrestrial laser-scanning survey in which seven different scans were acquired from different positions in order to limit the shadow areas (Figure 8).

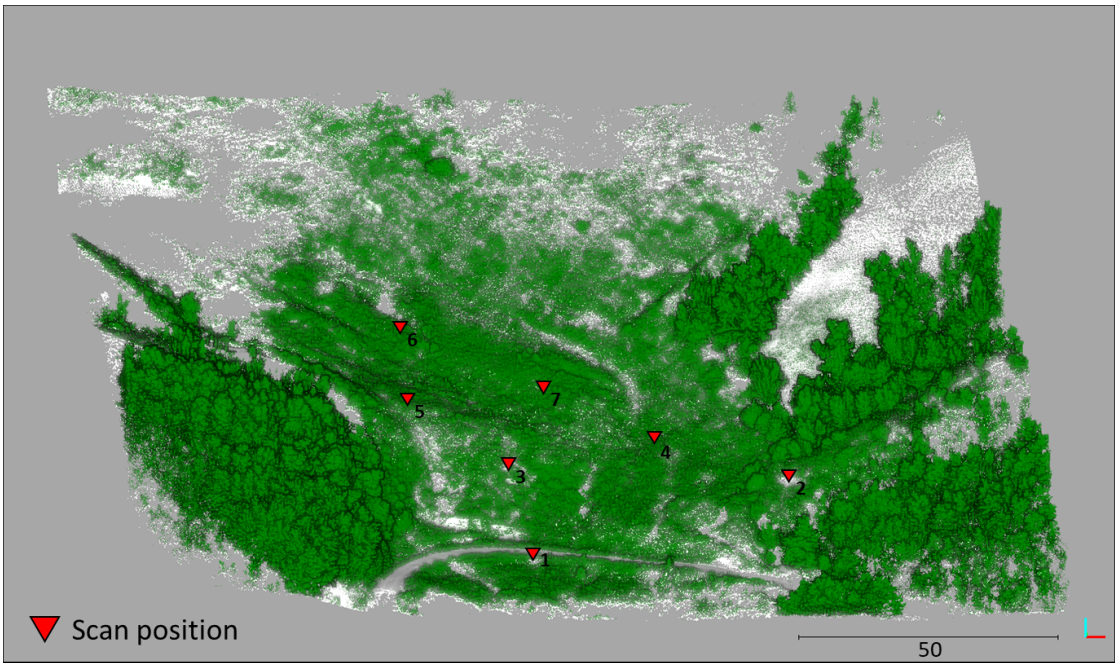

Figure 8. Three-dimensional point cloud of Roncovetro landslide. Red triangles show the position of the seven scans that were carried out with terrestrial laser scanner RIEGL VZ-1000. Green and grey colors, vegetation and terrain, respectively, classified by applying a vegetation filter.

After being aligned, a vegetation filter was applied to split the point cloud into a PCV and a PCT. As in 2016, sensors were installed on 1.5 m high iron poles, as consequently considered in the parameter settings. The ROI was fixed at 100 m, i.e., the maximum distance at which Wi-GIM sensors properly communicate [28].

Due to the low visibility of the site, the authors wanted to exclude the positioning of one or more sensors in areas with scarce visibility, since it would have compromised the best positioning of the subsequent sensors. WiSIO places devices in LOS to each other but without considering if the surrounding areas are highly visible. In stress conditions like the Roncovetro landslide, due to the presence of vegetation, as well as terrain roughness and slopes, it may happen that a sensor is correctly positioned in LOS with the others and is following the distance criteria, but in an area from which few points are visible, sacrificing the positioning of succeeding devices. To prevent this, the geomorphologically highest locations were selected as PAs. To automatically select these areas, simple computation was performed. Cells were extracted by moving a sampling cube of fixed dimension (5 m in the presented work) on the point cloud terrain. For each of them, the medium elevation value (Z_{med}) and the medium of the 10 highest elevation values (Z_{max}) were calculated. PAs were thus defined as points residing in cells whose Z_{max}–Z_{med} passed the threshold of 2 m (Figure 9). Note that the choice of calculating Z_{med} as mentioned above, instead of directly using the maximal-elevation value of each cell, was done to avoid outlier issues. All input parameters used for the Roncovetro application are listed in Table 3.

Table 3. Summary of input parameters used for Roncovetro application.

h [1]	ROI	ROI_{P1}	*m* [2]	P1 Iterations
1.5 m	100 m	5 m	20%	5
R Sector 1	**R Sector 2**	**R Sector 3**	**R Sector 4**	**R Sector 5**
3593 m	18,010 m	71,700 m	180,104 m	285,446 m

Note: [1] Iron-pole height. [2] Value expressed as percentage of points visible from Pi.

The selection of P_1 strongly influenced the whole deployment. As an example, Figure 9 shows three different configurations obtained by changing original position ROI_{P1}, which framed an area of 5 m around P_1. In Cases 1 and 3, it was situated in central and easy-to-reach positions, while in Case 2, C was situated a bit farther upstream. The most satisfying configurations were the ones in Cases 2 and 3, due to their distribution in the landslide's body and to the presence of one or more stable points located outside the landslide. Since it was the intention of the authors to install the system in the same area where it was in 2016 (see Figure 10c,d), aiming at a complete comparison of the collected data, the new installation of the Wi-GIM system was done in February 2019, following Configuration 3 (Figure 9c).

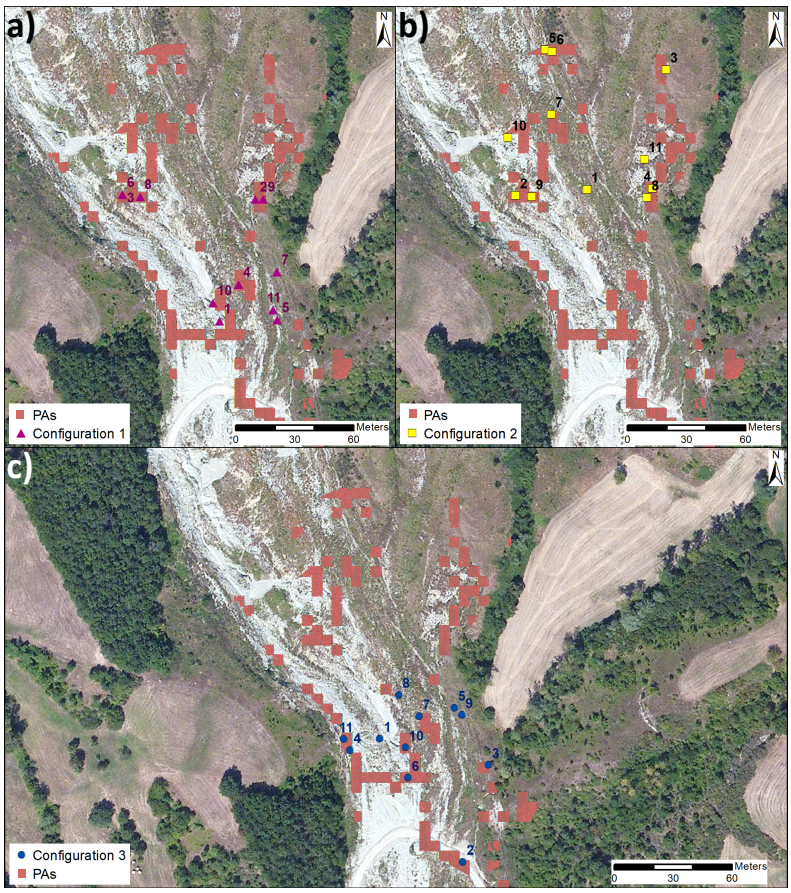

Figure 9. Different configurations obtained by changing the position of P_1. Red dots represent points residing in cubes whose Z_{max}–Z_{med} value passed the 2 m threshold, i.e., priority areas. (**a**) Configuration 1; (**b**) Configuration 2; (**c**) and Configuration 3, the one used for the new installation of the Wireless sensor network for Ground Instability Monitoring (Wi-GIM) system.

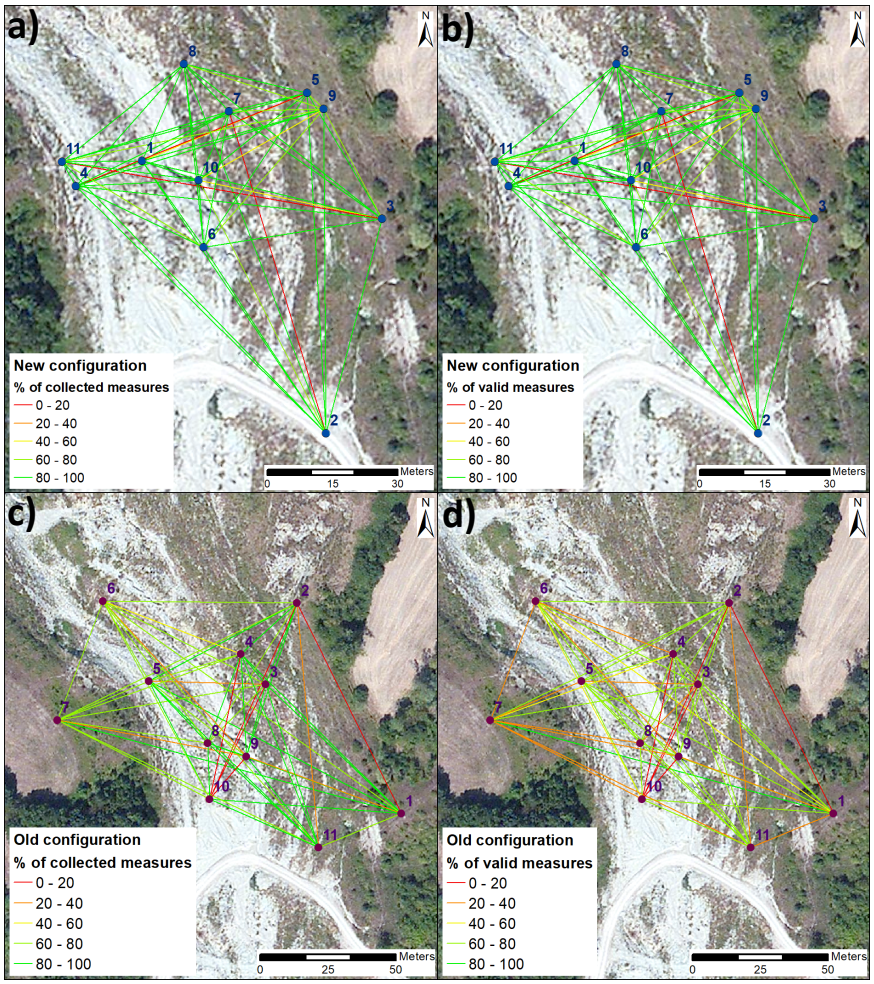

Figure 10. Links. Percentage of (**a**) collected measures in new configuration; (**b**) valid measures in new configuration; (**c**) collected measures in old configuration; and (**d**) valid measures in old configuration.

4. Discussion

As shown in Figure 9c, seven out of eleven devices lay in PAs. A non-PA point was, however, P_1, whose position was manually picked out. The remaining three were outside them, as a result of the relationship between distance and *m* value (set equal to 20% of points visible from P_i). All sensors were effectively in LOS, as demonstrated by the analysis carried out with the camera/eye center-visualization function in CloudCompare. Despite this, the acquired data showed a lack of three couples, i.e., 3–11, 2–7, and 4–5. This leak could be due to changes in vegetation or to problems related to the decawave. Concerning vegetation, please note that, due to logistic problems, three months passed from scan survey to installation. This period may have led to some changes on the field due to animal transitions or branches falling. Furthermore, some problems were already encountered in the past because of the low-cost nature of the decawave [28].

To better understand the effectiveness of WiSIO, a comparison between data from the 2016 installation and actual ones was pointed out, considering the amount of acquired data as their worth. The percentage of collected and valid data in the old, as in the new, configuration is shown in Figure 10, where valid data means the data with no outliers or zero values. In Figure 10a,b, the percentages of collected and valid measures, respectively, are reported in the form of colored links for each couple of nodes. As evident, values were between 90% and 100% for all couples, except for the four abovementioned pairs and for couples 1–5, 3–10, and 9–10. In these three last cases, both collected and valid data were between 45% and 50% (Figure 10a,b). The same procedure was done for data obtained with the old configuration, i.e., the 2016 one (Figure 10c,d). In this case, the percentage of collected data

was less than 60% for 10 couples, and the percentage of valid measures did not reach 60% for 22 couples. That means that not only were fewer data acquired in the old configuration due to non-LOS problems, but data quality was most importantly worse, as well. Indeed, in the old configuration, data acquisition did not guarantee validity, as it is for the new configuration, where there was no discrepancy between the percentage of collected and valid data. Considering couples whose percentage of valid data was at least 60% as suitable, 50 out of 56 couples were suitable in the new configuration (i.e., 89.3%), instead of 34 out of 56 in the old one (i.e., 60.7%); thanks to WiSIO, the amount of valid data increased by 28.6%. An example showing this improvement is reported in Figure 11. Couples 1–11 from the old configuration and 2–3 from the new one were considered due to similar distance, position, and visibility conditions. They both collected at least 60% of the measures (even though it is strongly visible from Figure 11 that there was loss of data in couple 1–11) but yielded different results in terms of validity. Indeed, couple 1–11 presented several problems with outliers and multipath effects, issues which were eased in couple 2–3 (Figure 11).

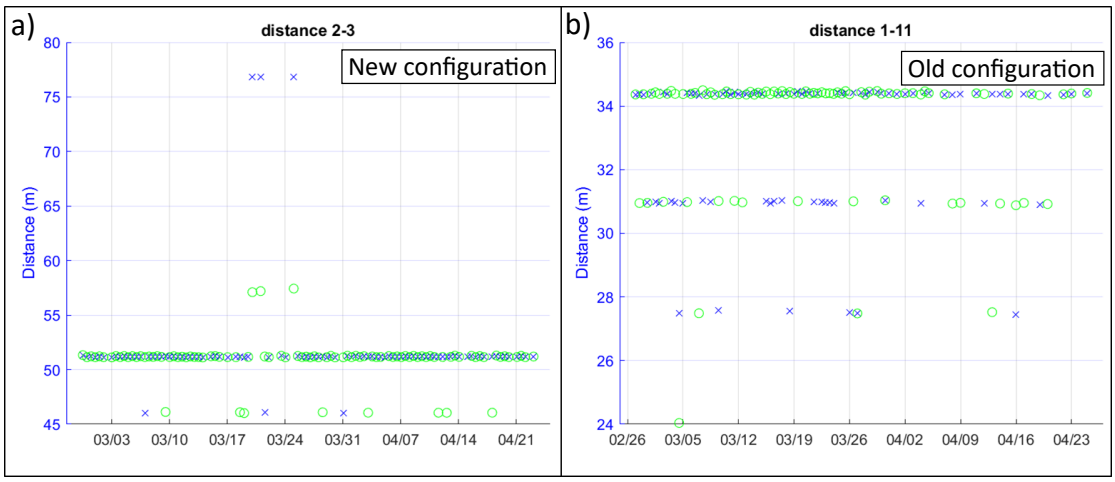

Figure 11. Comparison between data from new and old configurations. (**a**) Couple of nodes 2–3 and data from new configuration. (**b**) Couple of nodes 1–11 from old configuration. Green circles and blue crosses, distance from one node to the other and vice versa. Where measurement points were extremely dense, blue crosses were overlaid by green circles.

Contrary to precision, whose values remained in the range of 2 to 5 cm once outliers were removed [28], WiSIO also had an effect on data accuracy, affected by the presence of obstacles. Comparing the offset between real distances and Wi-GIM-measured distances, it was clear how the placement of sensors in LOS contributes to increasing accuracy (Figure 12). The histogram shows the offset between real and measured distances ("Error" in the graph) for all node couples in the two configurations. It is clearly visible that, in the old configuration, the error was higher in almost all couples due to the presence of obstacles, whereas it is lower and much more distributed in the new configuration.

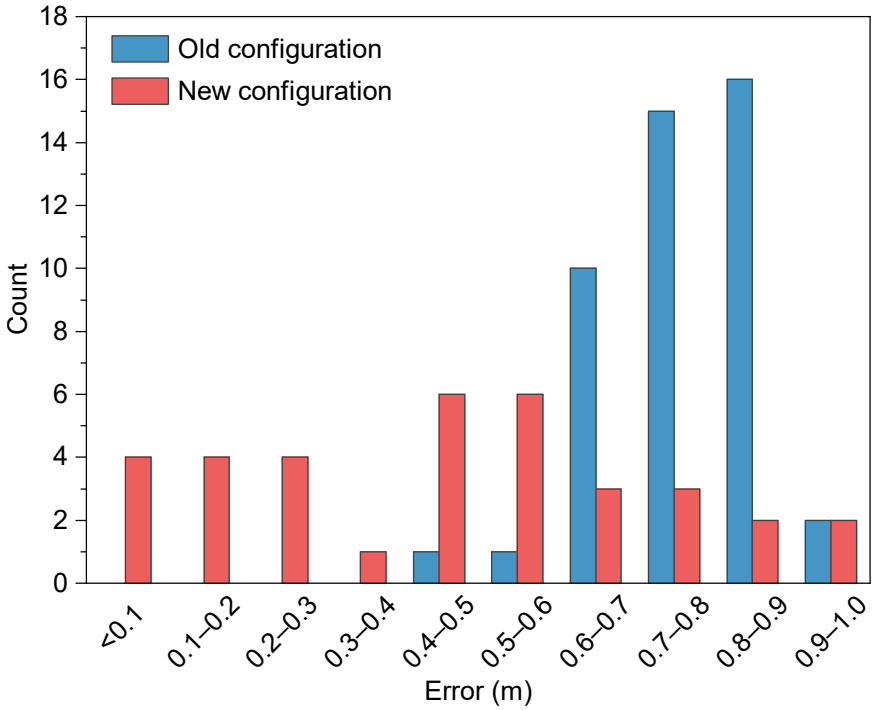

Figure 12. Histogram of errors in the new and old configurations. X-axis, error expressed as offset between real distances and ones evaluated by Wi-GIM. Y-axis, count shows number of couples. For old configurations, real distances were evaluated by means of a robotic total station, while, in the new one, they were measured on the georeferenced digital elevation model.

5. Conclusions

With the aim of finding the best position to place WSN devices requiring free LOS, a MATLAB tool called WiSIO was developed. The algorithm finds the best device deployment by following three criteria: inter-visibility by means of the Hidden Point Removal modified operator, equal distribution, and positioning in preselected priority areas. This works directly with 3D point clouds without performing any surface reconstructions, leading to skipping the process of generating surface models avoiding errors and approximations.

A 3D model of a harsh environment with low visibility was developed to validate WiSIO. Results highlighted good correspondence with reality, showing just a few false negatives (i.e., visible points marked as nonvisible). Precision could, in any case, be improved by increasing the number of sectors and evaluating the radius for each of them. The sectors there are, the fewer the false negatives or/and false positives, and the higher the running time. In this framework, the automatic configuration of a ROI, allowing to process only points inside it, helps in saving time, avoiding the use of useless data. Its size is a parameter under the control of the user, able to govern the area to place devices without cutting or modifying the original point cloud. Moreover, the choice of m (the number of points to consider checking if some of them lie in PAs) allows the user to evaluate the importance of PAs with respect to the equal distribution of the devices in the field. This, together with the detection of P_1 within ROI_{P1} and the selection of PAs, ensures that users apply their knowledge in the procedure, despite the automatic approach of the code.

WiSIO was applied to the real case of the Roncovetro landslide, Central Italy. There, an ultrawide-band WSN, i.e., the Wi-GIM system, was installed, as previously done in 2016. The comparison of data acquired by the system positioned with and without the help of the proposed algorithm allowed for a complete analysis of the efficiency of the method. Both the quality and quantity of the acquired data were analyzed through a comparison with the 2016 data. According to the results, almost all devices were in LOS with each other, evenly distributed in space, and seven of eleven devices lay

in Priority Areas. As a matter of fact, 87.5% of measurements were collected by the master node, and, between them, 89.3% were good enough to be considered reliable data with respect to 2016, when the acquired measurements were 82% and only 60.7% were valid. The presented results are very promising, showing how a simple elaboration can be essential to have more and more reliable data.

The continuous growth of new technologies for the development of increasingly high-resolution 3D point clouds (i.e., drone laser scanners and high-resolution cameras) could still improve the reliability of the presented approach that is highly dependent on the data source.

Author Contributions: Conceptualization, T.G., G.G., and F.N.; methodology, T.G. and F.N.; resources, N.C.; supervision, C.M., G.G. and M.J.; validation, T.G. and G.G.; writing—original draft, T.G.; writing—review and editing, M.J.

Funding: This research received no external funding.

Acknowledgments: The authors would like to thank Elena Benedetta Masi and Mattia Ceccatelli of the University of Florence for their fieldwork support. We would also like to express our gratefulness to the anonymous reviewers for their valuable advice.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Eberhardt, E.; Watson, A.D.; Simon, S. Improving the interpretation of slope monitoring and early warning data through better understanding of complex deep-seated landslide failure mechanisms. In Proceedings of the 10th International Symposium on Landslides and Engineered Slopes, Xi'an, China, 30 June–4 July 2008; pp. 39–51.
2. Antonello, G.; Casagli, N.; Farina, P.; Leva, D.; Nico, G.; Sieber, A.J.; Tarchi, D. Ground-based SAR interferometry for monitoring mass movements. *Landslides* **2004**, *1*, 21–28. [CrossRef]
3. Dou, J.; Chang, K.T.; Chen, S.; Yunus, A.; Liu, J.K.; Xia, H.; Zhu, Z. Automatic case-based reasoning approach for landslide detection: Integration of object-oriented image analysis and a genetic algorithm. *Remote Sens.* **2015**, *7*, 4318–4342. [CrossRef]
4. Hemalatha, T.; Ramesh, M.V.; Rangan, V.P. Effective and Accelerated Forewarning of Landslides Using Wireless Sensor Networks and Machine Learning. *IEEE Sens. J.* **2019**, *14*. [CrossRef]
5. Kromer, R.; Walton, G.; Gray, B.; Lato, M. Development and Optimization of an Automated Fixed-Location Time Lapse Photogrammetric Rock Slope Monitoring System. *Remote Sens.* **2019**, *11*, 1890. [CrossRef]
6. Prabha, R.; Ramesh, M.V.; Rangan, V.P.; Ushakumari, P.V.; Hemalatha, T. Energy efficient data acquisition techniques using context aware sensing for landslide monitoring systems. *IEEE Sens. J.* **2017**, *17*, 6006–6018. [CrossRef]
7. Estrin, D.; Culler, D.; Pister, K.; Sukhatme, G. Connecting the Physical World with Pervasive Networks. *IEEE Pervasive Comput.* **2002**, *1*, 59–69. [CrossRef]
8. Zanella, A.; Zorzi, M. Reti di sensori: Dalla teoria alla pratica. *Notiziario Tecnico Telecom Italia* **2006**, *1*, 47–59.
9. Rosi, A.; Berti, M.; Bicocchi, N.; Castelli, G.; Corsini, A.; Mamei, M.; Zambonelli, F. Landslide monitoring with sensor networks: Experiences and lessons learnt from a real-world deployment. *Int. J. Sens. Netw.* **2011**, *10*, 111–122. [CrossRef]
10. Kamal, A.R.M.; Hamid, M.A. Reliable data approximation in wireless sensor network. *Ad Hoc Netw.* **2013**, *11*, 2470–2483. [CrossRef]
11. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless sensor networks: A survey. *Comput. Netw.* **2002**, *38*, 393–422. [CrossRef]
12. Arampatzis, T.; Lygeros, J.; Manesis, S. A survey of applications of wireless sensors and wireless sensor networks. Intelligent Control. In Proceedings of the 2005 IEEE International Symposium on Mediterrean Conference on Control and Automation, Limassol, Cyprus, 27–29 June 2005; pp. 719–724.
13. Tan, G.; Jarvis, S.A.; Kermarrec, A.M. Connectivity-guaranteed and obstacle-adaptive deployment schemes for mobile sensor networks. *IEEE Trans. Mob. Comput.* **2009**, *8*, 836–848.

14. Wang, X.; Xing, G.; Zhang, Y.; Lu, C.; Pless, R.; Gill, C. Integrated coverage and connectivity configuration in wireless sensor networks. In Proceedings of the 2003 1st International Conference on Embedded Networked Sensor Systems, Los Angeles, CA, USA, 5–7 November 2003; ACM: New York, NY, USA, 2003; pp. 28–39.
15. Zou, Y.; Chakrabarty, K. Sensor deployment and target localization based on virtual forces. In Proceedings of the INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications, San Francisco, CA, USA, 30 March–3 April 2003; pp. 1293–1303.
16. Le Breton, M.; Baillet, L.; Larose, E.; Rey, E.; Benech, P.; Jongmans, D.; Guyoton, F. Outdoor UHF RFID: Phase Stabilization for Real-World Applications. *IEEE J. Radio Freq. Identif.* **2017**, *1*, 279–290. [CrossRef]
17. Mandrone, G.; Vacha, D. Real-time landslide monitoring system using UWB radio transceiver as a sensor. In Proceedings of the EGU General Assembly Conference Abstracts 2018, Vienna, Austria, 8–13 April 2018; Volume 20, p. 4941.
18. Mucchi, L.; Jayousi, S.; Martinelli, A.; Caputo, S.; Intrieri, E.; Gigli, G.; Gracchi, T.; Mugnai, F.; Favalli, M.; Fornaciai, A.; et al. A Flexible Wireless Sensor Network Based on Ultra-Wide Band Technology for Ground Instability Monitoring. *Sensors* **2018**, *18*, 2948.
19. Yavari, M.; Nickerson, B.G. *Ultra-Wideband Wireless Positioning Systems*; Technical Report TR14-230; Department Faculty Computer Science University New Brunswick: Fredericton, NB, Canada, 2014.
20. Alwan, N.A.; Mahmood, A.S. Distributed gradient descent localization in wireless sensor networks. *Arab. J. Sci. Eng.* **2015**, *40*, 893–899. [CrossRef]
21. Dardari, D.; Conti, A.; Ferner, U.; Giorgetti, A.; Win, M.Z. Ranging with ultrawide bandwidth signals in multipath environments. *Proc. IEEE* **2009**, *97*, 404–426. [CrossRef]
22. Schroder-Preikschat, W.; Kapitza, R.; Kleinoder, J.; Felser, M.; Karmeier, K.; Labella, T.H.; Dressler, F. Robust and efficient software management in sensor networks. In Proceedings of the 2017 2nd International Conference IEEE Communication Systems Software and Middleware, Bangalore, India, 7–12 January 2007; pp. 1–6.
23. Wang, X.; Wang, Z.; O'Dea, B. A TOA-based location algorithm reducing the errors due to non-line-of-sight (NLOS) propagation. *IEEE Trans. Veh. Technol.* **2003**, *52*, 112–116. [CrossRef]
24. Jaboyedoff, M.; Oppikofer, T.; Abellán, A.; Derron, M.H.; Loye, A.; Metzger, R.; Pedrazzini, A. Use of LIDAR in landslide investigations: A review. *Nat. Hazards* **2012**, *61*, 5–28. [CrossRef]
25. Lato, M.J.; Diederichs, M.S.; Hutchinson, D.J.; Harrap, R. Evaluating roadside rockmasses for rockfall hazards using LiDAR data: Optimizing data collection and processing protocols. *Nat. Hazards* **2012**, *60*, 831–864. [CrossRef]
26. Carlà, T.; Farina, P.; Intrieri, E.; Botsialas, K.; Casagli, N. On the monitoring and early-warning of brittle slope failures in hard rock masses: Examples from an open-pit mine. *Eng. Geol.* **2017**, *228*, 71–81. [CrossRef]
27. Atzeni, C.; Barla, M.; Pieraccini, M.; Antolini, F. Early warning monitoring of natural and engineered slopes with ground-based synthetic-aperture radar. *Rock Mech. Rock Eng.* **2015**, *48*, 235–246. [CrossRef]
28. Intrieri, E.; Gigli, G.; Gracchi, T.; Nocentini, M.; Lombardi, L.; Mugnai, F.; Frodella, W.; Bertolini, G.; Carnevale, E.; Favalli, M.; et al. Application of an ultra-wide band sensor-free wireless network for ground monitoring. *Eng. Geol.* **2018**, *238*, 1–14. [CrossRef]
29. Fisher, P.F. First experiments in viewshed uncertainty: The accuracy of the viewshed area. *Photogramm. Eng. Remote Sens.* **1991**, *57*, 1321–1327.
30. Murgoitio, J.J.; Shrestha, R.; Glenn, N.F.; Spaete, L.P. Improved visibility calculations with tree trunk obstruction modeling from aerial LiDAR. *Int. J. Geogr. Inf. Sci.* **2013**, *27*, 1865–1883. [CrossRef]
31. Dodd, H.M. The Validity of Using a Geographic Information System's Viewshed Function as a Predictor for the Reception of Line-of-Sight Radio Waves. Ph.D. Thesis, Virginia Tech, Blacksburg, VA, USA, 2001.
32. Guth, P.L. Incorporating vegetation in viewshed and line-of-sight algorithms. In Proceedings of the ASPRS/MAPPS 2009 Conference, San Antonio, TX, USA, 16–19 November 2009.
33. Katz, S.; Tal, A.; Basri, R. Direct visibility of point sets. *ACM Trans. Graph.* **2007**, *26*, 24. [CrossRef]
34. Zhang, G.; Van Oosterom, P.; Verbree, E. Point Cloud Based Visibility Analysis: First experimental results. In Proceedings of the Societal Geo-Innovation: Short Papers, Posters and Poster Abstracts of the 20th AGILE Conference on Geographic Information Science, Wageningen, The Netherlands, 9–12 May 2017.

35. Machado e Silva, R.; Esperança, C.; Marroquim, R.; Oliveira, A.A. Image space rendering of point clouds using the HPR operator. *Comput. Graph. Forum* **2014**, *33*, 178–189. [CrossRef]
36. Katz, S.; Tal, A. On the visibility of point clouds. In Proceedings of the IEEE International Conference on Computer Vision 2015, Las Condes, Chile, 11–18 December 2015; pp. 1350–1358.