

MODELING OF ROOFS FROM POINT CLOUDS USING GENETIC ALGORITHMS

Natália Sabariego¹ - ORCID: 0000-0002-1118-2829

Jorge Antonio Silva Centeno² - ORCID: 0000-0002-2669-7147

¹Universidade Federal do Paraná, Programa de Pós-Graduação em Ciências Geodésicas, Curitiba, Paraná, Brasil.

E-mail: sabariego.na@gmail.com

²Universidade Federal do Paraná, Departamento de Geomática, Curitiba, Paraná, Brasil.

E-mail: centeno@ufpr.br

Received in 04th November 2019

Accepted in 29th January 2020

Abstract:

Building roof extraction has been studied for more than thirty years and it generates models that provide important information for many applications, especially urban planning. The present work aimed to model roofs only from point clouds using genetic algorithms (GAs) to develop a more automatized and efficient method. For this, firstly, an algorithm for edge detection was developed. Experiments were performed with simulated and real point clouds, obtained by LIDAR. In the experiments with simulated point clouds, three types of point clouds with different complexities were created, and the effects of noise and scan line spacing on the results were evaluated. For the experiments with real point clouds, five roofs were chosen as examples, each with a different characteristic. GAs were used to select, among the points identified during edge detection, the so-called 'significant points', those which are essential to the accurate reconstruction of the roof model. These points were then used to generate the models, which were assessed qualitatively and quantitatively. Such evaluations showed that the use of GAs proved to be efficient for the modeling of roofs, as the model geometry was satisfactory, the error was within an acceptable range, and the computational effort was clearly reduced.

Keywords: Roof modeling; Genetic algorithms; Point clouds; Light Detection And Ranging - LIDAR.

How to cite this article: SABARIEGO, N.; CENTENO, J. A. S. Modeling of roofs from point clouds using genetic algorithms. *Bulletin of Geodetic Sciences*. 26(1): e2020004, 2020



This content is licensed under a Creative Commons Attribution 4.0 International License.

1. Introduction

The extraction of building roofs has been studied for more than thirty years and it generates models that provide important information for the registration and leasing of radio/phone antennas, urban planning and solar energy studies, for example. To build an accurate 3D model, every edge and plane of roofs must be identified. This task can be performed using remote sensing. Until the mid-1990s, this data came only from high-resolution aerial images. However, in late 1990s, a new data source emerged: LIDAR (Light Detection and Ranging) which has significantly contributed with new data to the field (Awrangjeb, Zhang and Fraser, 2013). 3D modeling LIDAR proved to be more effective than aerial imaging in some applications. When it comes to detecting roof planes and their orientation, the point clouds obtained with LIDAR present better results due to the availability of a large number of points on the surface. Classic examples were described by Maas and Vosselman (1999), Rottensteiner and Briese (2003), Brenner (2005) or Rottensteiner et al (2005).

The most commonly applied principles in building extraction and modeling using LIDAR data assume that the roof is composed by planar surfaces and are based on the detection of planar regions, as in Ercolin Filho, Centeno and Mitishita (2016), or edges (Xiao et al 2015). These approaches are based on the estimation of the planes that compose the roof to model the intersections that define its shape. A major drawback is the need to manually enter the number of roof planes before selecting significant points of each plane to estimate an equation that describes this roof.

To overcome this limitation, a different approach was proposed in this project: genetic algorithms (GAs) were used as an optimization tool to automatically find significant points in a particular roof's point cloud, so that there would be no need to indicate the number of planes to be modeled beforehand.

2. Literature review

Urban scenes are very complex and different objects can be found within a small region. That is the reason why, automated urban object extraction from remotely sensed data is challenging and the results are not always reliable. Therefore, most research on the area focuses on the extraction of a single object type – like roofs – for example (Niemeyer, Rottensteiner and Soergel, 2014).

Extraction approaches can be divided into three categories, based on their data sources: 2D information based (images, including aerial ones), 3D information based (airborne laser scanning - ALS) and fused 2D-3D information based. The first category is not considered sufficiently automated or reliable because image artifacts can lead to significant errors. That is why 2D and 3D information are commonly fused, because this complementary information can be used to improve the accuracy of the extraction process (Du et al., 2017).

For example, large-scale maps can be helpful in the detection and modeling of roofs, because they provide the buildings' locations in the point cloud (sparing the effort of detecting them) and their contours. An example of this approach can be found in Awrangjeb, Zhang, and Fraser (2013) who proposed a method for automatic 3D reconstruction of roofs by integration between LIDAR data and multispectral orthoimages. Their work consisted of segmenting points above the ground using an image line guided segmentation technique to extract roof planes, which proved to be very efficient for removing vegetation.

Further examples of integration of LIDAR data and spectral data are found in Cheng et al. (2008), Yong Li et al. (2013), Potje et al. (2014), Xu, Morgenroth and Manley (2015); Adeleke and Smit (2016), Zhang and Lin (2017) and Martín-Jiménez et al. (2020). These methods make full use of the complementary advantages of LIDAR data and optical images.

In some cases, however, fusing 2D and 3D data may increase the risk of error propagation in the extraction

results, especially due to misalignment and spatial resolution differences. That is why some approaches rely just on 3D information for roof extraction. These methods consist of segmenting the point cloud to detect parts of the roof elements, which are then combined to rebuild the entire roof. The advantage of this type of approach for roofs is that it is not restricted to pre-defined forms, but it demands high computational effort.

Kim and Shan (2011) presented an approach to roof modeling from LIDAR data, where segmentation was performed by minimizing an energy function formulated as a multiphase level set. The ridges and edges were delineated by the union of the zero level contours of the level adjustment functions. To rebuild a 3D roof model, the points of the roof structure were determined by the intersection of adjacent segments or line segments of the building boundaries and then connected based on their inferred topological relationships from the segmentation result.

Kabolizade, Ebadi, and Mohammadzadeh (2012) presented a reconstruction method based on GAs by optimizing the height and slopes of gable roofs. The proposed algorithm followed three steps: (1) detection, where points above the ground (the potential buildings) were identified and aggregated to form building blobs; (2) extraction, where a generalization function simplified the outline of the building and produced a meaningful shape; (3) reconstruction, where a GA-based model was reconstructed by looking for the planes that are closest to the point cloud through a fitness function.

A more recent example is described by Rychard and Borkowski (2016), who developed an automated method that allows the recognition and semantic interpretation of topological building structures. These authors proposed an unambiguous decomposition of complex objects into simpler, predefined parametric structures, and allowed a reconstruction of a topological unit without independent overlapping elements. The first step was roof plane extraction (by the segmentation of building point clouds), followed by topology identification and recognition of predefined structures.

Bearing all this in mind, the present work can be included in the group of algorithms that use only LIDAR data for roof modeling (i.e. with no spectral information). The proposed approach follows the line of thought proposed by Kim and Shan (2011), where the basic elements for modeling are roof edges. The main difference is that in their work there is no need to verify the approximation of regions of the point cloud by planes. As in Kabolizade, Ebadi, and Mohammadzadeh (2012), the solution is sought through GAs, but unlike what the aforementioned authors propose, the objective function is not associated with the best roof planes, but to the most significant edges. The proposed algorithm has two steps: (1) potential edges are detected analyzing the local variation of the neighborhood around each point; (2) the most representative edge points are identified applying GA.

2.1 Genetic Algorithms (GA)

GAs are sets of rules and logical procedures that simulate biological evolution processes (using metaphors of natural selection, e.g. genetic recombination and survival of the fittest) in the computational universe to solve a problem. For this, the possible solutions are treated as individuals of a population that will evolve within an iterative process, looking for the optimal (or suboptimal) solution by combining previous solutions (individuals). Basically, a GA is a directed search controlled by a fitness function, which determines the adequacy of a solution based on its fitness score at a given moment of the evolutionary process. So, once the adequacy of each solution is known, the best individuals are then selected for reproduction (Kramer, 2017).

GAs are part of the so-called “evolutionary computation” and, according to Beasley (1993), they may not find the global optimum solution to a problem, but they are generally good at finding acceptably good solutions. They are robust, generic and versatile tools, useful for many applications, such as development of mathematical strategies and formulas; object reconstruction; analysis of economic models; simulation of ecosystems and of organic molecules’ properties, among others (Kramer, 2017).

John Holland, who first proposed and developed GAs in the 1960s/1970s based on the theory of Darwinism to explain the evolution of species, divided the execution of the algorithm into seven steps: initialization, fitness function, selection, crossing, mutation, update, and finalization (Mitchel, 1995). These stages can be summarized as follows: (1) an initial population, usually formed by random individuals, is determined; (2) the whole population is evaluated according to a criterion determined by the fitness function, that measures the quality of the individual; (3) the best-evaluated individuals are chosen through the selection operator as the basis for a new set of possible solutions, also called “new generation”, which in turn is obtained by applying (4) crossover and (5) mutation operations on the selected individuals by mixing their characteristics. These steps undergo (6) repetition until an acceptable solution is found or if it is no longer possible for the algorithm to improve the solution (7). The process of population generation through GAs to arrive at the most adequate solution using the fitness of a chromosome/individual can be called convergence (Cox, 2005).

3. Methods

The problem to be solved was the modeling of flat face roofs. The flowchart of the process is shown in Figure 1, and is further explained in the subsequent subsections.

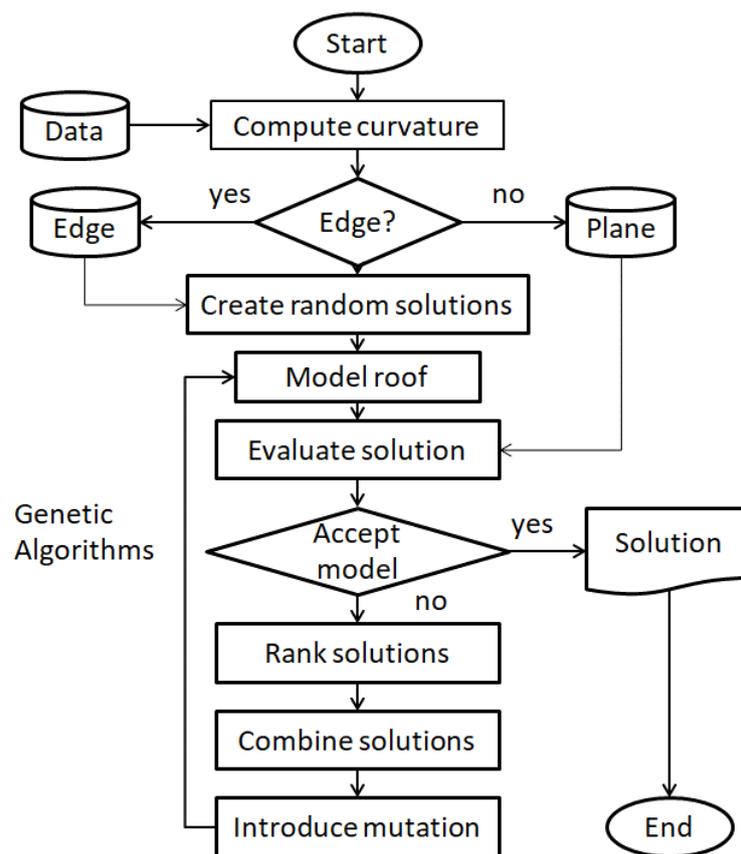


Figure 1: Flowchart of the method.

3.1 Analysis of curvature

In the first step, an edge detection algorithm based on the concept described in Weinmann, Jutzi, and Mallet (2013) was implemented to detect the points that best represented the edges. Weinmann, Jutzi, and Mallet (2013) proposed to analyze the distribution of the points within a neighborhood around each point (a local analysis) by using the variance-covariance matrix of the three-dimensional coordinates of the points. The eigenvalues (λ_1 , λ_2 , and λ_3) and eigenvectors of this matrix were used to compute spatial descriptors, which in turn were used to describe the shape of the region.

Based on the spatial information of all points within a given neighborhood U , invariant moments representing geometric properties can be computed for each 3D point (Maas and Vosselman, 1999), as well as the respective 3D covariance matrix (S) (Jutzi and Gross, 2009). The eigenvalues λ_1 , λ_2 and λ_3 of the covariance matrix S , where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$, can be directly used to describe the local 3D structure, or also other measures based on these eigenvalues can be derived, which encompass special geometric properties (West et al., 2004; Toshev, Taskar and Daniilidis, 2010; Mallet et al., 2011).

The spatial distribution of the points within a small region provide information about the local structure of the surface and can be measured using parameters such as linearity (L_λ) and planarity (P_λ), which provide information about the presence of a linear 1D structure, a flat 2D structure or a volumetric 3D structure. Change of curvature is represented by C_λ . The mathematical definitions of the aforementioned measurements are provided bellow and they show a characteristic behavior depending on the local 3D structure, according to Weinmann, Jutzi and Mallet (2013). The linearity, planarity and curvature are described in Equations 1, 2 and 3.

$$L_\lambda = \frac{\lambda_1 - \lambda_2}{\lambda_1} \quad (1)$$

$$P_\lambda = \frac{\lambda_2 - \lambda_3}{\lambda_1} \quad (2)$$

$$C_\lambda = \frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3} \quad (3)$$

3.1.1 Roof model and candidate points

A roof can be well described by its significant edges. As an example, Figure 2a shows a high-density point cloud corresponding to a gable roof. It is noted that only six points, located in the roof edges, are crucial for the definition of the geometry, and these were called 'significant points' (Figure 2b). The rest of points could be considered redundant, mostly coplanar and occurring in flat regions, as observed in Figure 2a. The same applies to roofs with four or more sloping sides.

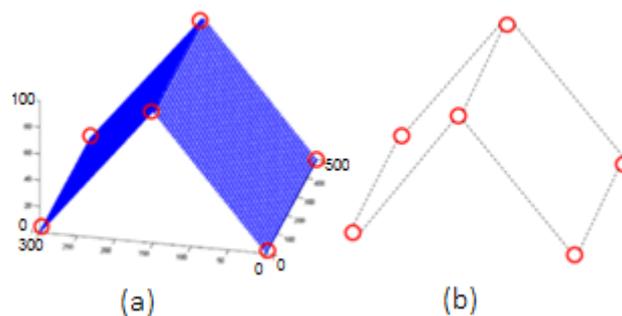


Figure 2: Example of roof modeling: (a) point cloud of a roof (blue) and its significant points (red); (b) model that was obtained from these significant points.

Among the desired points there are points along the borders of the roof that need to be identified. The input point cloud is the result of a previous segmentation that removed the ground points. Therefore it contains only points on the roof. For the detection of the borders additional artificial points were added (with $Z = 0$) to represent the ground around the point cloud. For this purpose, the point cloud was projected on a regular grid with resolution equal to the theoretical distance between points. Grid elements with at least one point were attributed the value 1. This allowed detecting the grid cells that are null and have at least one neighbor with positive value. The central coordinates (XY) of such cells were computed and added to the point cloud with a null height. Then, the Delaunay triangulation was computed and the original points that belong to triangles with at least one point with null height were identified and classified as border. The result of this step is the extraction of a subset of points (E) located at borders from the point cloud (PC). This new points were added to the original point cloud and used to detect edges with the curvature analysis.

In the next step, the curvature of the point cloud around each point was analyzed to detect sharp edges. A Delaunay triangulation was computed from the XY coordinates and a mesh, like the one displayed on Figure 3, produced. The tessellation allowed identifying the neighborhood around each point and with the coordinates of the point in the neighborhood dispersion of the points in this location was described by the variance-covariance matrix. The eigenvalues and eigenvectors of this matrix were obtained, and finally, a volume descriptor was computed for the region. In this case, the curvature parameter, described in Equation 3, was used. By applying a curvature threshold, the point cloud was divided into two groups: flat regions and curved regions, which can be either ridges or edges giving rise to a set of significant points.

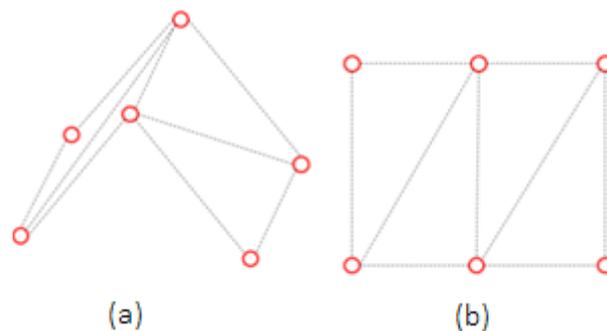


Figure 3: Representation of a roof using a triangulation: (a) in 3D; (b) top view.

This threshold strongly depends on the point cloud's texture. If the cloud contains data from two different and superposed areas from the LIDAR scan, it may show high relief even in flat regions due to registration error. That is why special attention must be paid to the registration step. Generally, this threshold value is close to the mean curvature.

As the point cloud contains only points on the roof, a border (with $z = 0$) was added to represent the ground around the object. For this, regularly spaced points were distributed around the horizontal projection of the point cloud. The result of this step is the extraction of a subset of points (E) located at edges from the point cloud (PC).

3.2 Search algorithm based on genetic approach

After edge detection, it is verified if a subset of N points of the set E (identified edges) can be used to describe the roof, that is, to describe a shape that contains all the other points of the point cloud (PC). For this purpose, a search algorithm was developed based on the principles of genetic algorithms. However, as mentioned before, the set of points located in flat areas is very large and provides redundant data. Since analyzing all points would be unnecessary and demand high computational effort, a progressive reduction of point density in the clouds was made beforehand. The reductions

were applied by selecting: 1 out of every 5 points (low-density – 1); and 1 out of every 2 points (medium-density – 2).

The low-density point cloud was adopted as initial solution. As the location of the real edges is *a priori* unknown, an initial guess was proposed. The bounding box of the cloud was computed and its dimensions were used to build a regular grid. The closest point to each node of the grid was then selected as initial guess. This includes points at the corners of the bounding box and some others inside it, which is a good choice because roofs are generally regular and rectangular. In this case, GAs work faster, but the solution is only a good approximation of the roof shape. In this stage, the objective function is linear.

In the second stage, the solution obtained with the low-density point cloud was included as a part of the initial population and the medium-density point cloud was analyzed. By doing so, GAs can optimize the previous solution and get an answer that is closer to the object's shape.

Finally, the set of all points in flat regions was analyzed, this time including the suboptimal solution from the medium-density cloud analysis. In this case, the objective function is quadratic and allows a fine adjustment.

The selected individuals were mated using a single-point crossover (Pham and Liu, 1995), the chosen crossover point being in the middle of the chromosome (between the two central genes). Then the subroutine that executes the mutation operator randomly altered one or more gene values.

3.2.1 Proposed Optimization Problem

The formulated optimization problem can be described as follows.

The total set of points U was divided into two groups of points with 3D coordinates: P (points located in edges); Q (points located in flat areas). P being the set:

$$P(i) = (x_i, y_i, z_i) \in U \quad (4)$$

The next step was to find the optimal (or sub-optimal) set S , a subset of P , which adequately represented the vertices of the roof planes ($S \in P$), that is, that minimized the following objective function:

$$F(S) = \min, \text{with } S \in P \quad (5)$$

Considering the example in Figure 2, the solution to the problem is the restricted set of points that are highlighted with red circles. Since, at first, the minimum number of points needed to solve the problem is unknown, a subset S with a greater number of points than necessary can be used. This excess of points is not harmful, as it would only introduce redundancy into the model.

Based on this, and considering that an individual should represent a probable solution to the problem, the individual (or solution) stores the IN index of each point in the solution set S . The solution is then described by a vector with the NPs points selected to describe the roof:

$$SOL = [sol_1, sol_2, sol_3, \dots, sol_{NP}] \quad (6)$$

where sol_i is one of the points of subset S , and $sol_i \in IN$.

The solution set SOL was then represented as a binary string. The second set of points Q was formed by the points located in flat regions of the roof's planes. This set was used to verify the suitability of each solution (SOL). For each solution, it was checked whether the planes formed by the selected points ($sol_1, sol_2, sol_3, \dots, sol_n$) approached the points in the set Q . The solution that best represented the set Q based on points selected from the set P was sought:

$$Dist(T(SOL), Q) = \min \quad (7)$$

where Q is the set of all points and T is the Delaunay's triangulation computed with the selected points.

Because the set Q contains large numbers of points in the same plane of the roof, it is highly redundant. For this reason, instead of checking the hypothesis in the whole set Q , a random subset of points was used for the verification, which made the process faster.

3.2.2 Proposed fitness function

The fitness function should identify the triangulation points (SOL) that most closely match the shape of the roof, represented by PL . This means that the subset of points (PL) is contained in the planes that the set SOL defines through triangulation. In practice, this is not always the case due to the variation of the XYZ coordinates, as a consequence of the data acquisition process. Therefore, the objective was to minimize the average distance between the point cloud and the planes defined by the final triangulation:

$$Obj = \min (Dist (T(SOL), Q)), \text{ with } SOL \in P \text{ and } PL \in Q \quad (8)$$

In this context, the function to be minimized $Dist$, evaluated the proximity between the points PL and the triangulation $T(SOL)$. Therefore, for each solution, the 2D triangulation with the set of points SOL was calculated and a 3D triangulation was constructed by adding the Z coordinate. This triangulation is considered simpler because it is formed by a smaller number of points.

The similarity between the solution and the checkpoints is given by:

$$di (PL_i, T(SOL)) = (Z_i - ZT(X_i, Y_i)) \quad (9)$$

where X_i , Y_i and Z_i are the coordinates of the point PL_i , and ZT is the value of the interpolated height on the surface of the triangulation.

The average distance was given by:

$$Dist = \sum_{i=1}^{N_{PL}} di(PL_i, T(SOL)) / N_{PL} \quad (10)$$

Formally, the objective function could be described as:

$$SOL = \{SOL_j\} | Dist = \min \{Dist(PL, T(SOL))\} \quad (11)$$

Given a possible solution to the problem (SOL), an evaluation of the solution's suitability in each iteration went through the following steps:

- a) Creation of a 2D triangulation ($T(SOL)$) from the solution (SOL) points.
- b) For each point (P_i):
 - I. Identify the triangle (Tri_{ij}) in which the point P_i is inserted considering the coordinates XY ;
 - II. Interpolate the value of Z on the surface of the 3D triangulation;
 - III. Compute the distance between the point P_i and the triangle Tri_{ij} .
- c) Compute the mean distance and normalize it, ranging from zero to one.

Thus, it can be said that the goal here was: starting from random solutions, to converge to the optimal solution of the problem in the iterative process of genetic algorithms, as illustrated in Figure 4.

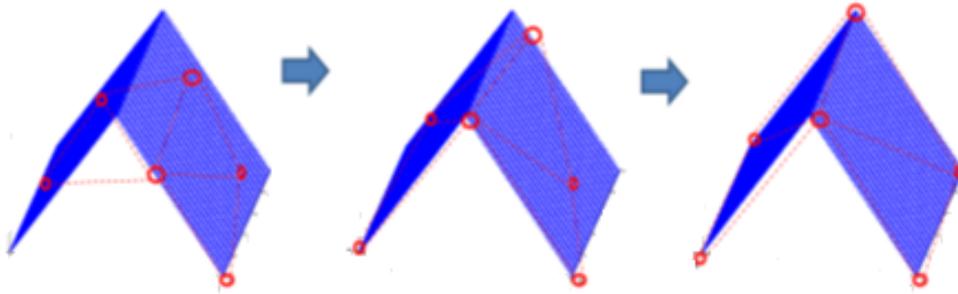


Figure 4: Iterative process of roof approximation.

The final fitness score should represent the proximity between the solution and the data. In Equation 12, max_i is a normalization factor that represents the greatest difference in height among the roof points (Equation 13).

$$Fitness = 1 - \frac{Dist}{max_i} \quad (12)$$

$$max_i = \max_z(U) - \min_z(U) \quad (13)$$

An alternative is to penalize larger differences by using a quadratic function:

$$Fitness = 1 - \frac{Dist^2}{max_i^2} \quad (14)$$

3.3 Model Simplification

As the necessary number of points or triangles to describe the roof is a priori unknown, a large quantity of them is used for modeling, which may result in redundant triangles. For this reason, an algorithm to eliminate these unnecessary triangles was developed. The angle between adjacent triangles was analyzed, and small vertices ($< 45^\circ$) and their composing edges were removed from the structure, resulting in a simplified model.

3.4 Analysis of the Generated Models

The generated models were compared to the original point clouds to evaluate their accuracy, both qualitatively and quantitatively. For the qualitative evaluation, the shape and overall characteristics of the roofs were analyzed. For the quantitative assessment, the coordinates of the significant points (found by the GA) were compared to their corresponding points in the real clouds (which were selected visually and manually for this).

4. Results

The method was evaluated using two different data sources: simulated point clouds and point clouds obtained by ALS. To simulate the point clouds, an algorithm that provided three different roof shapes in the form of point clouds was developed. The simulated dataset describes a controlled situation, where some parameters and variables can be analyzed. For example, the distance between lines (scans) and the noise in the signal were varied. The distance between scans refers to the longitudinal spacing in the direction of the flight, and the noise is a random component added to the simulated height data by varying the height in a range of 5 to 25 cm. This noise introduces variations in the surface, causing

the roof not to be a perfect plane. A similar effect could be achieved by varying the horizontal coordinates. Through this experiment it was also possible to assess the fitness score variation during the iterative process.

Figure 5 shows a representation of the simulated roofs, highlighting their significant points. They were named as follows: (a) roof 1 – gable roof, 2 sloping planes; (b) roof 2 – hip roof, 4 sloping planes; (c) roof 3 – intersecting hip roof, 8 sloping planes.

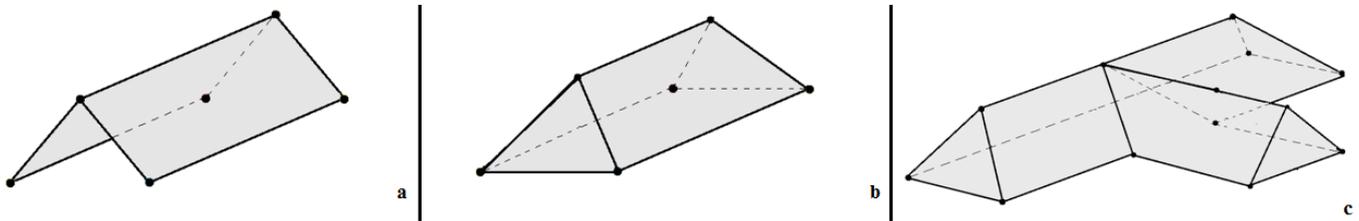


Figure 5: Simulated building models: (a) roof 1; (b) roof 2; (c) roof 3.

The LIDAR point clouds were kindly provided by Engefoto and they were preprocessed using the LAsTools software, so it was possible to separate the roofs from the other elements. These point clouds, by their turn, represented a real situation with several types of noise and points in different planes. Five different geometries, with varied complexities, were chosen as examples.

Using the curvature parameter proposed by Weinmann, Jutzi e Mallet (2013), it was possible to identify the points located in flat regions in all the point clouds (both simulated and real). Then, it was assumed that the rest of the points were located in edges or ridges, therefore being good candidates to describe the roof's shape.

4.1 Experiment I: effect of noise

The aim of this first experiment was to evaluate the effect of noise in the final result. For this, different point clouds from Roof 1 were generated by randomly varying the height coordinate by a maximum value of 5; 10; 15; 20; e 25 cm. As the expected altimetric accuracy of a typical ALS survey is around 10-20 cm, such values are reasonable. Line spacing was set at 5 m and distance between same scan points was 50 cm.

Figure 6 shows the results from two stages of the first experiment (named Test *a*): maximum noise level of 5 cm with (a) low and (b) medium density of internal points. These internal points, located in flat areas, are shown in red. The blue points represent the roof's edges and ridge points (located in the borders and in the center of the roof, respectively), identified through curvature analysis, and they are the same for all three stages. Right next to the point cloud representation is a graph that shows the evolution of the fitness score during the process.

After the first iterations, there were significant changes in the fitness scores (from approximately 0.89 to 0.98). However, after the 100th iteration, this improvement was less pronounced, and the situation remained stable. Figure 5b shows the progress during the second stage, using the medium-density cloud and the best solution from the previous stage. The first stage's contribution to this step was clearly important, as the score reached its maximum value and stabilized much sooner. Results from the third stage (using high density) were not included here, since there were no significant changes.

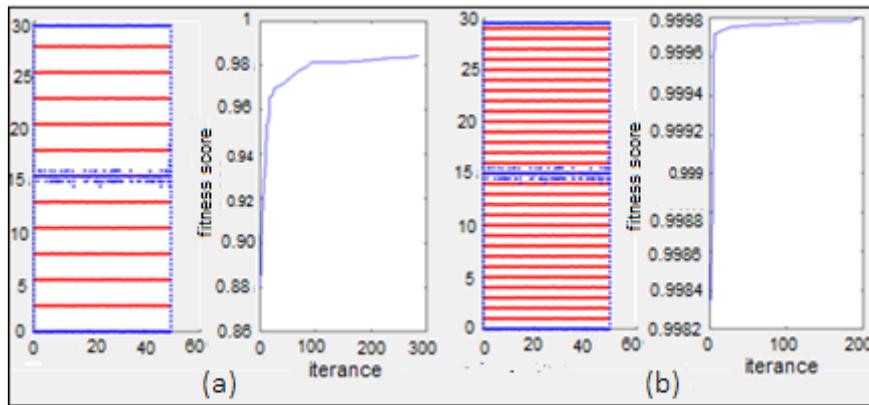


Figure 6: Iterative evolution for the modeling of roof 1, with scan spacing of 5 m and noise of 5cm with (a) low and (b) medium density of internal points.

Figure 7 presents the model obtained in this first experiment. The model accurately represents the proposed roof, both in terms of shape and dimensions.

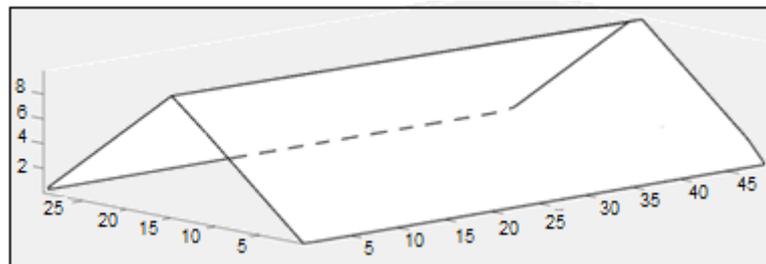


Figure 7: Final model from Test a.

To quantitatively assess this result, the significant points obtained by the algorithm were compared to those originally used to generate the point cloud. The differences between their coordinates are shown in Table 1. The resulting distance represents the vector addition of the three components.

Table 1: Comparison between the original significant points and the ones obtained through the algorithm.

| Point | Test a | | | | | | | | | |
|-------|-----------------------------|-------|-------|--------------------|-------|------|-----------------|------------|------------|-------------------------|
| | Theoretical Coordinates (m) | | | GA coordinates (m) | | | Difference (cm) | | | Resulting distance (cm) |
| | x_o | y_o | z_o | x | y | z | Δx | Δy | Δz | D |
| P1 | 0.02 | 29.53 | 0.24 | 0.51 | 29.55 | 0.21 | 48.6 | 2.0 | -3.0 | 48.7 |
| P2 | 0.03 | 15.00 | 9.87 | 0.03 | 15.00 | 9.87 | 0.0 | 0.0 | 0.0 | 0.0 |
| P3 | 0.01 | 0.05 | 0.04 | 0.01 | 0.05 | 0.04 | 0.0 | 0.0 | 0.0 | 0.0 |
| P4 | 49.51 | 0.04 | 0.04 | 49.03 | 0.01 | 0.03 | -48.0 | -3.0 | -1.2 | 48.1 |
| P5 | 49.51 | 15.02 | 9.90 | 49.52 | 14.51 | 9.67 | 1.0 | -51.0 | -23.3 | 56.1 |
| P6 | 49.52 | 29.54 | 0.22 | 49.52 | 29.02 | 0.55 | 0.0 | -52.0 | 33.4 | 61.8 |
| | Mean (cm) | | | | | | 0.3 | -17.3 | 1.0 | 35.8 |
| | RMS (cm) | | | | | | 27.9 | 29.8 | 16.7 | 44.1 |

The same procedure was repeated, but introducing noise to the dataset, with different maximum noise values (10, 15, 20 e 25 cm, which were named Tests *b* to *e*). Deviations higher than 25 cm were considered unnecessary because the distance accuracy of airborne LIDAR is around 15 cm (Baltsavias, 1999). Table 2 summarizes the numerical results for Tests *b* to *e*, and Figure 8 shows the obtained models.

Table 2: Numeric results of Tests *b* to *e*.

| Noise (cm) | Mean difference (cm) | | | Resulting distance (cm) | RMS (cm) | | | Total RMS (cm) |
|------------|----------------------|------|-------|-------------------------|----------|------|------|----------------|
| | x | y | z | | x | y | z | |
| 10 (b) | 1.4 | 6.6 | 29.9 | 67.7 | 2.6 | 92.3 | 58.9 | 109.5 |
| 15 (c) | 1.9 | -2.2 | 17.8 | 56.0 | 32.5 | 54.0 | 35.0 | 72.1 |
| 20 (d) | -14.8 | 22.0 | -11.8 | 39.8 | 25.7 | 60.2 | 39.8 | 76.6 |
| 25 (e) | -0.2 | 13.6 | 17.6 | 51.3 | 4.9 | 55.1 | 37.9 | 67.1 |

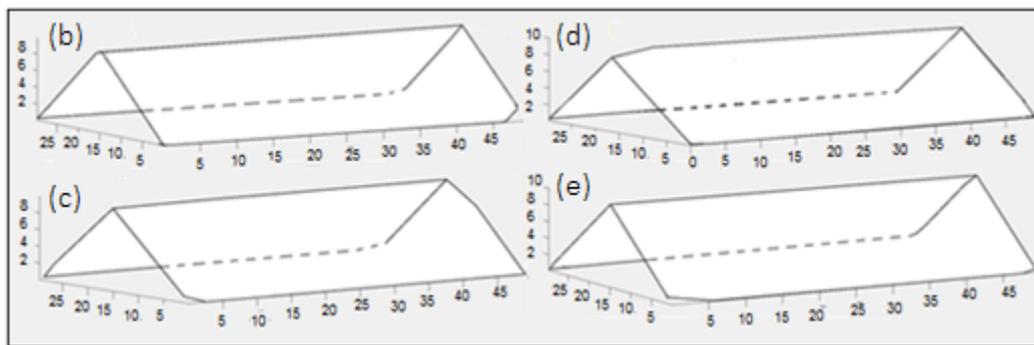


Figure 8: Models obtained in Tests *b* to *e*.

Models from Tests *b* to *e* were visually similar, which means that the proposed method can adequately model this roof shape even with different noise levels. There were differences on the roof ridges, because the modeling process failed to identify the optimal point and picked another one that is close and provides a good solution. Errors were smaller than the distance between scan lines (5 m).

As it can be seen in Table 2, however, this difference is not that large. Generally, a RMS of less than 1 m was obtained, except for Test *b* (maximum 10 cm noise).

The RMS values and resulting difference between coordinates were expected to be directly proportional to the noise level, but this was not the case here. This does not necessarily mean that all points were misidentified, and has probably to do with the fact that the sample size (the analyzed significant points) was relatively small, so a single mistake can cause an abnormal increase/decrease in the results. In spite of that, the error was within an acceptable value (considering the equipment's resolution), and all roofs were correctly outlined. The resulting distance, on the other hand, tends to be around 50 cm in average. This proves that the proposed method is capable of processing noisy data, but, evidently, the best results are achieved when there is less noise.

4.2 File format for graphics

Experiment II was performed using Roofs 2 and 3 as examples and the original distance between scan lines was varied. The distances were of 3 m and 5 m. Noise level was kept at 5 cm for both roofs. Once again, it was

noticed that the fitness score significantly rises during the first iterations and tends to stabilize as the iterative process advances. In this case, unlike before, there was a slight change in stage 3, when all the points from the cloud were analyzed (highest density).

Table 3 summarizes the comparison between the points obtained by the algorithm and those originally used to generate the point cloud, considering scan line spacing of 3 and 5 m for Roof 2. The results are satisfactory and prove that the proposed method can adequately model roof 2. There were slight differences in the ridges, as seen in Experiment I. The total RMS was generally below 40 cm. In addition, the mean resulting distance is around 18 cm. This shows that the proposed method was capable of processing data with low noise level and different point densities. As expected, the best results were obtained when the density was higher, but this resulted in more processing time.

Table 3: Error analysis of Roof 2 models using different scan line spacing.

| Spacing (m) | Mean difference (cm) | | | Resulting distance (cm) | RMS (cm) | | | Total RMS (cm) |
|-------------|----------------------|------|------|-------------------------|----------|------|------|----------------|
| | x | y | z | | x | y | z | |
| 3 | -10.0 | 5.8 | -1.0 | 24.2 | 30.6 | 13.9 | 10.8 | 35.3 |
| 5 | -0.3 | 10.4 | -0.2 | 10.4 | 20.0 | 28.4 | 8.1 | 35.7 |

For Roof 3, because of its complex shape, it was necessary to identify a larger number of points, and therefore, the genetic chain must contain more individuals. Figure 9 shows that the edge and ridge detection was efficient, identified significant points and eliminated redundant ones. The fitness score varied more during the first iteration (of approximately 0.77 to 0.95) when compared to the second (from 0.997 to 0.998), as observed in the previous experiments.

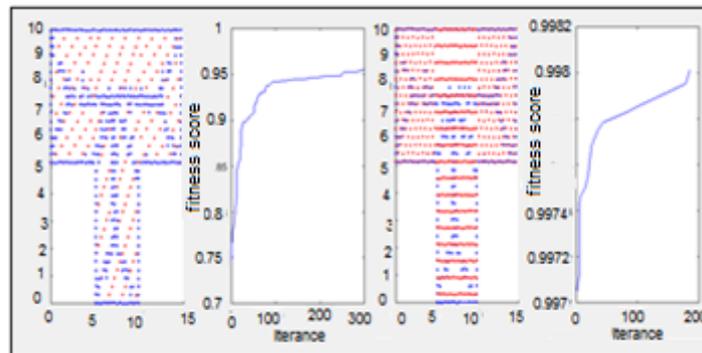


Figure 9: Evolution of iterations for the modeling of Roof 3, with line spacing of 3 m and noise level of 5 cm.

Table 4 summarizes the quantitative data from Roof 3 modeling, and Figure 10 shows the final Roof 3 models using different scan line spacing.

Table 4: Error analysis of Roof 3 models using different scan line spacings.

| Spacing (m) | Mean difference (cm) | | | Resulting distance (cm) | RMS (cm) | | | Total RMS (cm) |
|-------------|----------------------|-----|-----|-------------------------|----------|------|------|----------------|
| | x | y | z | | x | y | z | |
| 3 | -2.3 | 1.8 | 0.3 | 13.6 | 15.0 | 21.2 | 3.9 | 26.3 |
| 5 | 11.5 | 1.8 | 3.6 | 17.6 | 32.7 | 28.9 | 26.7 | 51.2 |

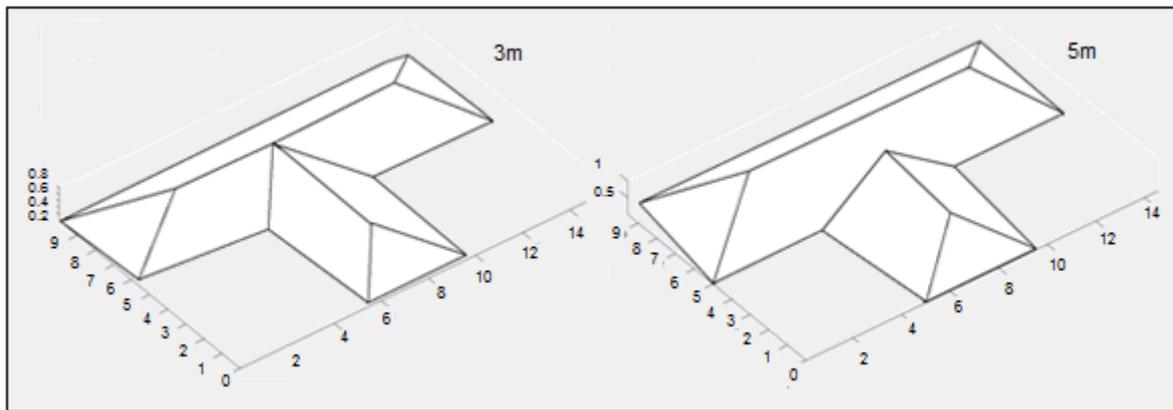


Figure 10: Roof 3 model considering different scan line spacing of 3 m and 5 m.

For both distances there were slight displacements concerning the ridge of roof's intercepting area, but the scan line spacing of 5 m generated more significant distortions, which means this alteration affects the method's accuracy. However, all results were considered satisfactory because the general shape and number of roof faces were correct.

Nevertheless, these simulated point clouds were generated parallel to the roof faces, so they may describe the roof's shape better. This virtually never happens in real data acquisition and is a factor that may influence the outcome of the experiment. That is why, if the scan line spacing is too large, a registration error of ridges and edges may occur, consequently generating inaccurate models.

4.3 Real point clouds (LIDAR)

For these experiments, a LIDAR point cloud from an area in the city of Recife-Brazil was used. The point cloud was preprocessed using software LAsTools, to select only the roofs of interest. Five roof point clouds, of different complexities, were chosen as examples. These point clouds were named as follows: Roof house 1; Roof house 2; Roof house 3; Roof house 4; Roof house 5. The modeling process was the same used for the simulated points.

To enable an easier understanding of the experiments, the results obtained roof 1 will be described in detail. Then, the results of the remaining roofs will be presented.

Roof house 1 is similar to the simulated hip roof. It is a simple roof, with 4 sloping sides, but its line distribution is not uniform, probably because it was the result of two or more scans. Figure 11 shows the original point cloud from Roof house 1's test. It is known that the spacing between points and lines are approximately 46 cm and 1 m, respectively. Figure 12 shows the evolution of the iterative process during its three stages.

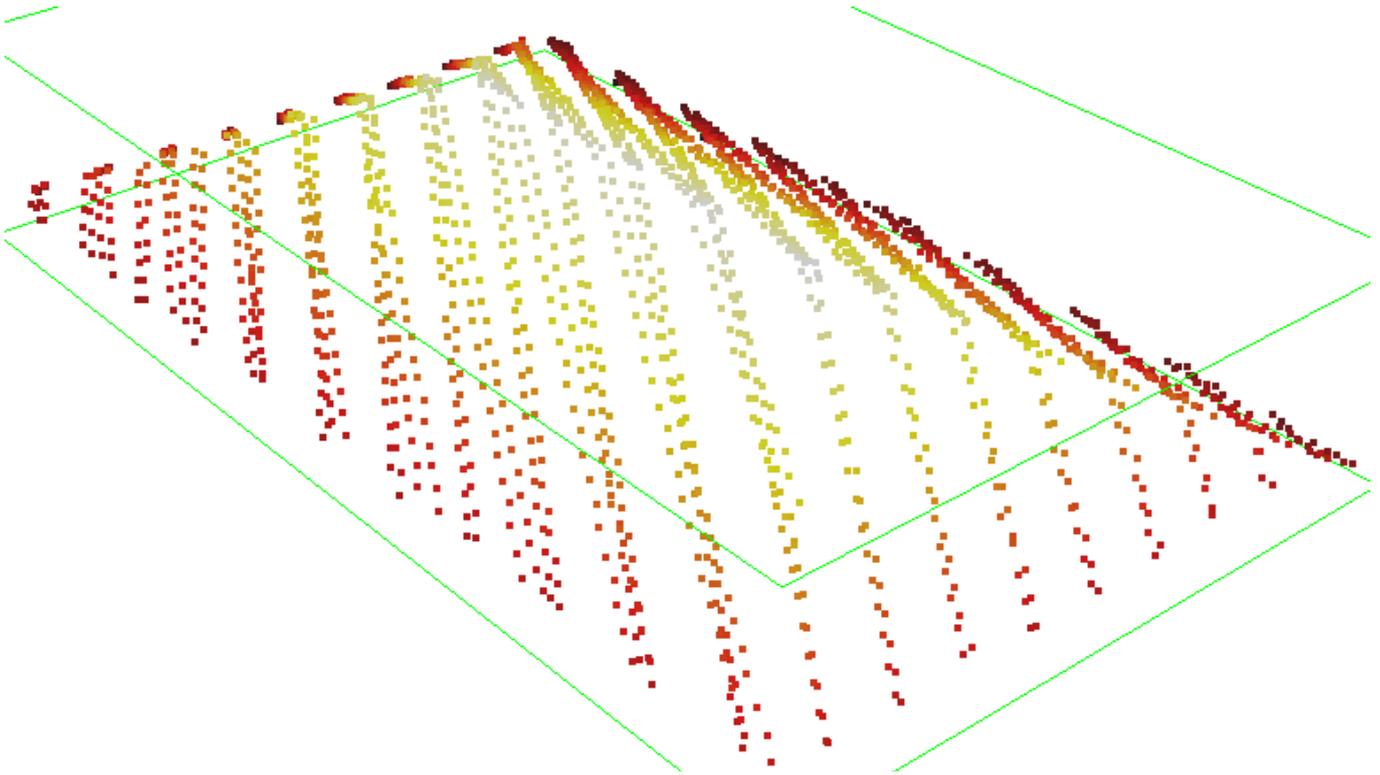


Figure 11: Original Roof house 1 point cloud.

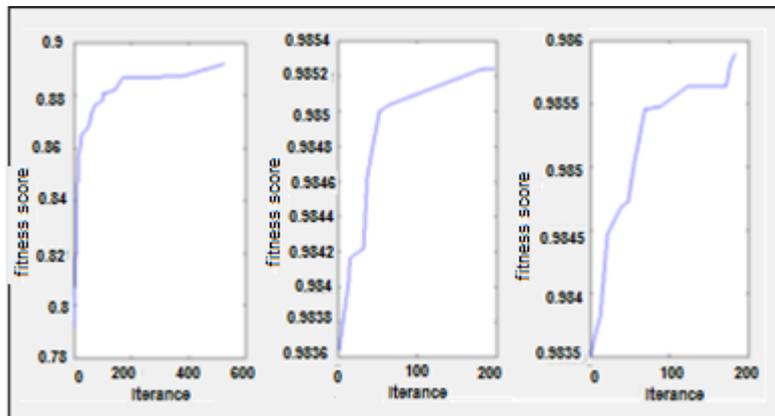


Figure 12: Evolution of the iterative process of Roof house 1 modeling: low, medium and high point density.

The initial guess using the bounding box is used to start the process with the low-density point cloud. This initial solution has a fitness score of 0.78, which is already a good approximation and reduces initial iterations. After the iterative search, fitness scores of 0.88 were obtained, showing that the model was improved.

In the second stage (using the best solution found in the previous step and the medium-density cloud), the fitness score was maximized, but not as much as in the previous stage (from approximately 0.9836 to 0.9852 and 0.9835 to 0.985), as shown in Figure 12.

Finally using all internal points and the solution found in the second stage, there was a progressive improvement, but the variation is not wide (0.9835 a 0.985). Nevertheless, this stage is important to correct mistakes from the previous step.

Figure 13 show the results of the aforementioned stages. The first one visibly generates a good approximation, and the following steps only optimize it.

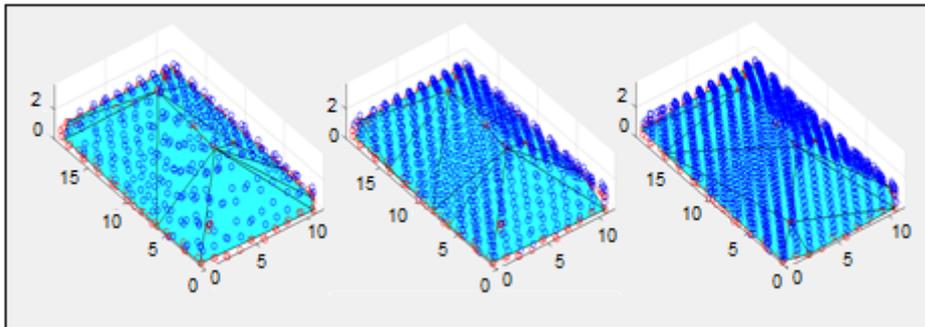


Figure 13: Modeling during the GAs evolution.

As a larger quantity of points was proposed for the model, redundant triangles were produced. They were eliminated in the last step by analyzing the angle between adjacent triangles: small vertices and its composing edges ($< 45^\circ$) were removed from the model. For example, Figure 14a shows the solution of this first experiment, and Figure 14b shows the same solution after this criterion was applied.

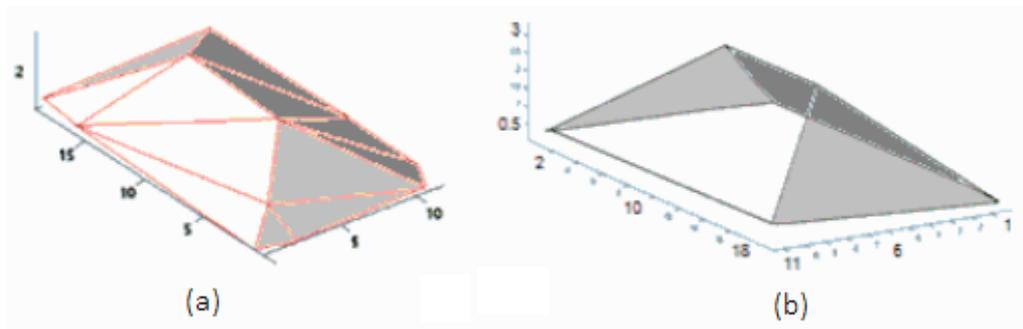


Figure 14: Final model for Roof house 1 and its appearance after removing vertices $< 45^\circ$.

The elimination of coplanar triangles simplifies the model. It does not improve it in terms of new faces or edges, but reducing the number of planes enables faster visualization and reduces storage space.

Figure 15 presents the original LIDAR point clouds of the other four roofs. Figures 15a and 15c show Roof house 2 and Roof house 4, which are L-shaped and Z-shaped, respectively. Roof house 3, shown in Figure 15b, is rectangular, with four sloping sides and two protuberances on the largest surfaces, which makes modeling more difficult. Figure 15d, which shows Roof house 5, is the top of a building (probably a roof terrace), with a big flat roof and an elevation, also flat, where the water tank is. This last example was included to verify whether the developed method was capable of detecting flat roofs. It is known that the spacing between points and lined is of approximately 45 cm and 1 m, respectively.

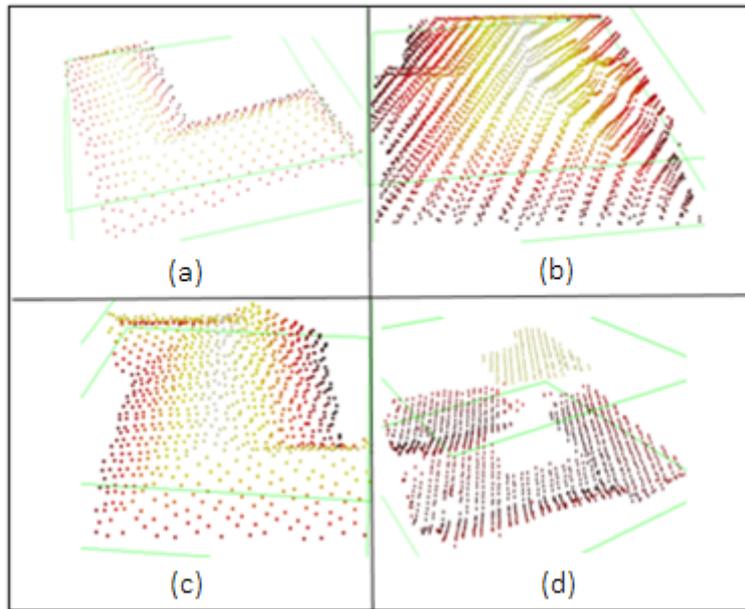


Figure 15: Original point clouds of Roofs (a) house 2, (b) house 3, (c) house 4 and (d) house 5.

As it happened in the experiment with roof 1, it was possible to see a similar tendency in the fitness evolution in all four cases, where the highest rise in the fitness score happened during the first stage. Therefore, the algorithm worked effectively with a reduced point cloud, as long as all roof planes had internal points. The detection process of ridges and edges was effective, because the points which were considered best candidates for significant points were sufficiently identified. In the second and third iteration, the evolution of the fitness score was less pronounced and not as fast.

Figure 16 shows the final results of these experiments, the reconstructed models. First, a qualitative analysis is made, based on the characteristics of the obtained models, and then a quantitative analysis of the results is presented.

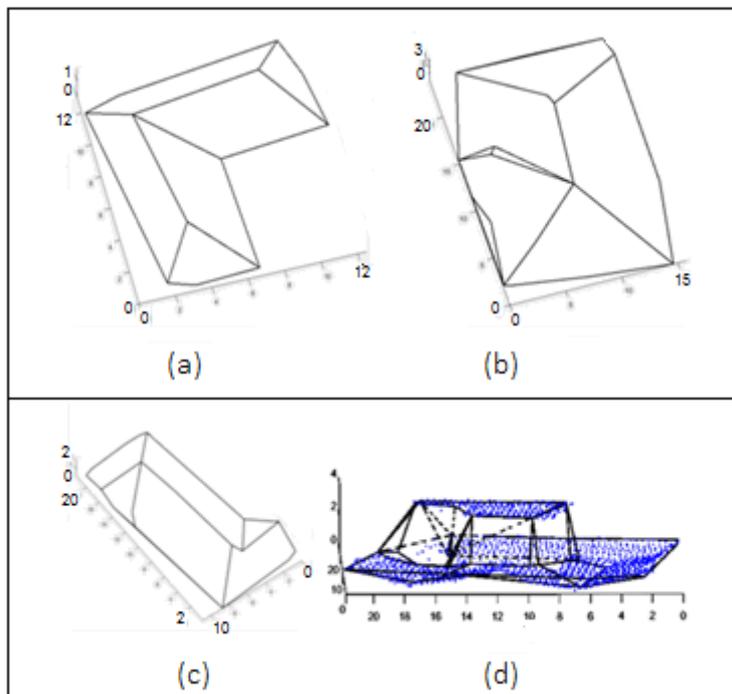


Figure 16: Final roof modeling of Roofs (a) house 2, (b) house 3, (c) house 4 and (d) house 5.

Roof house 2 was well modeled. It is possible to see the roof is composed of six planes and their intersections were well detected. Also, the edges of the roofs were not identified as straight lines. This is an inherent characteristic of the data and the scan pattern, and does not depend on the proposed method.

The results obtained for Roof house 3 were not completely accurate, which is justified by the complexity of this particular roof. Its general outline was modeled, but the proposed method failed to detect the details on the larger surfaces. In this case, the two protuberances were not identified, because they are not sufficiently higher than the roof itself, so they were considered to be on the same plane. However, a part of this detail was detected on the left side, as shown in Figure 16b.

The third case (Roof house 4) was successful. By looking at Figure 16c, it is possible to see that the roof’s ridge has two central points instead of one. This may indicate that more iterations would be necessary in the third stage. Indeed, it is noticeable that the fitness score rises in the third stage, but does not reach a maximum value neither stabilizes.

Modeling of Roof house 5 (Figure 16d) was satisfactory, for all the planes were identified. However, the detection process of the inferior plane should be improved. A possible issue in this case is the absence of points that represent the water tank walls, which can interfere in the method’s efficiency, as shown in Figure 17. Figures 17a and 17b show different views of Roof house 5’s plane identification. As Roof 5 is completely flat, what made it quite different from the other roofs, numerical results were not calculated.

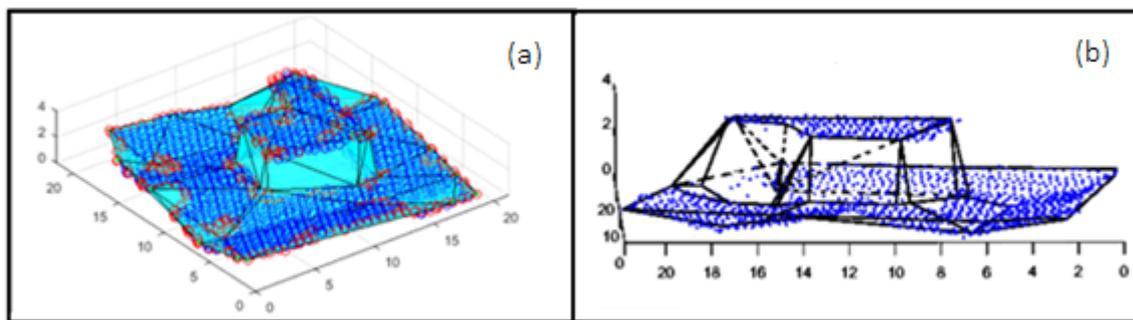


Figure 17: Edge (red) and internal (blue) points superimposed on the triangulation of the first partial result. (a) perspective view and (b) side view.

To assess whether the roofs’ significant ridges and points found by the method were close to the actual points, the points detected by the algorithm and the ones manually selected were compared. Each cloud was analyzed visually and the points that best described the roof were selected. Table 5 summarizes this comparison for Roof houses 1 to 4.

Table 5: Error analysis of the models for Roof houses 1 to 4.

| Roof | Spacing(cm) | Mean distance (cm) | | | Resulting distance (cm) | RMS (cm) | | | Total RMS (cm) |
|---------|-------------|--------------------|------|------|-------------------------|----------|------|------|----------------|
| | | x | y | z | | x | y | z | |
| House 1 | 100 | 2.0 | -5.0 | -1.4 | 10.1 | 10.7 | 13.4 | 1.3 | 17.2 |
| House 2 | 96 | -6.7 | -6.7 | -6.8 | 11.6 | 28.3 | 28.3 | 26.9 | 48.2 |
| House 3 | 100 | -5.7 | 0.0 | 0.0 | 5.7 | 9.8 | 0.0 | 0.0 | 9.8 |
| House 4 | 92 | -4.5 | -6.4 | 1.5 | 8.3 | 14.1 | 20.4 | 8.8 | 26.4 |

The obtained errors are not large when compared to the distance between the scan lines. Those values were, on average, close to the distance between points along the same line, which varied from 45 to 47 cm. This means that the solution found a neighboring point to the one considered visually adequate. Although this is a quantitative analysis, it should be highlighted that even reference points are hard to define.

5. Final considerations

This paper presented a new method for automatic 3D reconstruction of flat roof faces by processing point clouds using GAs. First, an edge detection algorithm was developed to outline the roofs. Then, a search process based on genetic algorithms was used to find their significant points, which are the ones essential to the representation of the geometries using GAs.

In conclusion, the proposed method effectively models flat roof faces. Results were considered satisfactory because although the resulting geometries were not completely accurate, they were very close to the actual roof shapes, for both real and simulated data. The difference between the roof's real coordinates and the ones obtained by the algorithm, as well as the RMS values, were within the expected range and were practically insignificant.

Also, using simulated point clouds, it was possible to assess the effect of some parameters, such as noise and scan line spacing, in the reconstructed models. Errors were small and within the expected values for all noise ranges, and inaccuracies were only found in edges and ridges. Even when the noise range was of 25 cm, which is considered unusually high (since LIDAR accuracy is around 15 cm), the algorithm was capable of correctly outlining the roofs. Unlike expected, however, the range of 25 cm did not correspond to the largest distance between coordinates nor RMS values, and, in fact, this variation seemed random. It is very probable, though, that there is a threshold where the aforementioned conclusions are not valid anymore, but they are not likely to happen under normal conditions.

As for scan line spacing, the roof's geometry was correctly identified using both 3 m and 5 m. Errors were small and within the expected values. Theoretically, the best results are usually obtained when using a higher density point cloud. However, during this experiment, a variation of only 2 m (from 3 to 5 m) was not sufficient to cause significant alterations neither in distance between coordinates nor in RMS values. The fact that this variation did not truly affect the results also suggests that the proposed method is capable of processing data with different densities.

As for real LIDAR point clouds, results were also satisfactory, especially for flat roof faces. When the roofs presented some peculiar design details, the method's efficiency was reduced as the algorithm failed to identify these points among the others. Four of these real roofs were quantitatively analyzed, and the RMS values found were close to the distance between two points in a same scan line (which is around 45 to 47 cm). This means that instead of finding the most adequate point (visually speaking) the algorithm found one of its neighboring points. It is also important to mention that the significant points chosen in the real point clouds (as reference for comparison) are hard to define, which could also be an error source.

It should be highlighted that the algorithm was also able to identify a flat roof with a water tank on top. A quantitative assessment was not made in this case, but the roof planes correctly identified. Some inaccuracies were detected, which means that the method should still be improved to better model the planes, but it is important to mention that the uncertainty regarding the water tank's limits negatively affected the method's efficiency.

Compared to other existing methods, the use of GAs presented some advantages: supplementary information (such as maps, structure libraries or plane detection) was not required; only an effective edge detection algorithm was needed. And because it did not require a predefinition of the roof's shape, it could be used to model irregular geometries and with any number of sloping sides. Also, dividing the GA's initial process in three stages (using different cloud densities) was proven to reduce processing time.

Future works include further processing of the roof lines, because most of the inaccuracies in the models presented in this work could be solved by applying restrictions regarding parallelism and right angles.

AUTHOR'S CONTRIBUTION

Natália Sabariego and Jorge Centeno conceived of the presented idea. Natália Sabariego developed the analytical methods and carried out the experiments. All authors discussed the results and contributed to the final manuscript. Jorge Centeno supervised the project as advisor of the M.Sc. thesis.

References

- ADELEKE, A. K.; SMIT, J. L. Intergration of LIDAR data with aerial imagery for estimating rooftop solar photovoltaic potentials in city of cape town. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, v. 41, 2016.
- AWRANGJEB, M.; ZHANG, C.; FRASER, C.S. Automatic extraction of building roofs using LIDAR data and multispectral imagery. *ISPRS journal of photogrammetry and remote sensing*, v. 83, p. 1-18, 2013.
- BALSAVIAS, E.P. Airborne laser scanning: existing systems and firms and other resources. *ISPRS Journal of Photogrammetry & Remote Sensing*. 54 1999 164–198, 1999.
- BEASLEY, D., BULL, D.R., Martin, R.R. An overview of genetic algorithms: Part 1, fundamentals. *University Computing* 15 (2), pp. 56-69, 1993.
- BRENNER, C. Building Reconstruction From Images And Laser Scanning, *International Journal Of Applied Earth Observation And Geoinformation*, Vol 6, Pages 187-198, 2005.
- CHENG, L., GONG, J., CHEN, X., & HAN, P. Building boundary extraction from highresolution imagery and lidar data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, v. 37, n. PART B3, p. 693-698, 2008.
- COX, E. *Fuzzy modeling and genetic algorithms for data mining and exploration*. Elsevier, 2005.
- DU, S. et al. Automatic building extraction from LiDAR data fusion of point and grid-based features. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 130, p. 294-307, 2017.
- ERCOLIN FILHO, L.; CENTENO, J. A. S.; MITISHITA, E. A. Automatic detection of flat areas on building roofs using LIDAR point cloud *Revista Brasileira de Cartografia*, v. 68, n. 3, 2016.
- FORLANI, G., NARDINOCCHI, C., SCAIONI, M., ZINGARETTI, P. Building Reconstruction And Visualization From LiDAR Data. *The International Archives Of The Photogrammetry, Remote Sensing And Spatial Information Sciences*, Vol. Xxxiv, Part 5/W12, 2004.
- JARZĄBEK-RYCHARD, M.; BORKOWSKI, A. 3D building reconstruction from ALS data using unambiguous decomposition into elementary structures. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 118, p. 1-12, 2016.
- JUTZI, B.; GROSS, H. Nearest neighbour classification on laser point clouds to gain object structures from buildings. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, v. 38, n. Part 1, p. 4-7, 2009.
- KABOLIZADE, M.; EBADI, H.; MOHAMMADZADEH, A. Design and implementation of an algorithm for automatic 3D reconstruction of building models using genetic algorithm. *International Journal of Applied Earth Observation and Geoinformation*, v. 19, p. 104-114, 2012.

- KIM, K.; SHAN, J. Building roof modeling from airborne laser scanning data based on level set approach. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 66, n. 4, p. 484-497, 2011.
- KRAMER, O. *Genetic Algorithm Essentials*. Volume 679 of *Studies in Computational Intelligence*. Springer, 2017. 92 p.
- LI, Y.; WU, H.; AN, R.; XU, H.; HE, Q.; & XU, J. An improved building boundary extraction algorithm based on fusion of optical imagery and LiDAR data. *Optik-International Journal for Light and Electron Optics*, v. 124, n. 22, p. 5357-5362, 2013.
- MAAS, H., VOSSelman, G.: 1999 "Two Algorithms For Extracting Building Models From Raw Laser Altimetry Data", *ISPRS Journal Of Photogrammetry & Remote Sensing* Vol. 54, Pp 153–163.
- MALLET, C.; BRETAR, F.; ROUX, M.; SOERGEL, U.; & HEIPKE, C. Relevance assessment of full-waveform lidar data for urban area classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 66, n. 6, p. S71-S84, 2011.
- MARTÍN-JIMÉNEZ, J. et al. Multi-scale roof characterization from LiDAR data and aerial orthoimagery: Automatic computation of building photovoltaic capacity. *Automation in Construction*, v. 109, p. 102965, 2020.
- MITCHELL, M. *Genetic Algorithms: An Overview*. *Complexity*, v. 1, n. 1, p. 31-39, 1995.
- OVERBY, J., BODUM, L., KJEMS, E., IISOE, P.M.: 2004. Automatic 3d Building Reconstruction From Airborne Laser Scanning And Cadastral Data Using Hough Transform. *International Archives Of Photogrammetry And Remote Sensing*, Vol. 35, Part. B3, Pp. 296-301.
- NIEMEYER, J.; ROTTENSTEINER, F.; SOERGEL, U. Contextual classification of lidar data and building object detection in urban areas. *ISPRS journal of photogrammetry and remote sensing*, v. 87, p. 152-165, 2014.
- PHAM, D. T. AND LIU X. 1995. *Neural networks for identification, prediction and control*, Springer-Verlag, London, U.K., 1995, 238pp.
- POTJE, G. A.; PAZOTI, M. A.; SILVA, F. A. D.; PEREIRA, D. R.; ARTERO, A. O.; & PITERI, M. A. Extração de contorno de telhados de edificações utilizando dados laser e imagens aéreas. (Extraction of contour roofs from buildings using laser data and aerial images) In: *Colloquium Exactarum*. 2014. p. 104-119.
- ROTTENSTEINER, F. AND BRIESE, C: 2003. "Automatic Generation of Building Models From LiDAR Data and the Integration of Aerial Images", *ISPRS*, Vol. Xxxiv, Dresden.
- ROTTENSTEINER, F., TRINDER, J CLODE, S, KUBIK, K.: 2005, Automated Delineation Of Roof Planes From LiDAR Data, *ISPRS Wg III/3, V/3 Workshop "Laser Scanning 2005"*, Enschede, The Netherlands, Sept 12-14.
- RYCHARD, M.J.; BORKOWSKI, A. 3D building reconstruction from ALS data using unambiguous decomposition into elementary structures. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 118, p. 1-12, 2016.
- TOSHEV, A.; TASKAR, B.; DANILIDIS, K. Object detection via boundary structure segmentation. In: *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010. p. 950-957.
- WEINMANN, M.; JUTZI, B.; MALLET, C. Feature relevance assessment for the semantic interpretation of 3D point cloud data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, v. 5, p. W2, 2013.
- WEST, K. F.; WEBB, B.N.; LERSCH, J. R.; POTHIER, S.; TRISCARI, J. M.; & IVERSON, A.E. Context-driven automated target detection in 3D data. In: *Automatic Target Recognition XIV*. International Society for Optics and Photonics, 2004. p. 133-144.
- XIAO, Y.; WANG, C.; LI, J.; ZHANG, W.; XI,X.; WANG, C.; DONG, P. 2015 Building segmentation and modeling from airborne LiDAR data, *International Journal of Digital Earth*, 8:9, 694-709, DOI: 10.1080/17538947.2014.914252.
- XU, C.; MORGENROTH, J.; MANLEY, B. Integrating data from discrete return airborne LiDAR and optical sensors to enhance the accuracy of forest description: A review. *Current Forestry Reports*, v. 1, n. 3, p. 206-219, 2015.
- ZHANG, Jixian; LIN, Xiangguo. Advances in fusion of optical imagery and LiDAR point cloud applied to photogrammetry and remote sensing. *International Journal of Image and Data Fusion*, v. 8, n. 1, p. 1-31, 2017.