

## BUILDING BOUNDARY EXTRACTION FROM LIDAR DATA USING A LOCAL ESTIMATED PARAMETER FOR ALPHA SHAPE ALGORITHM

R. C. dos Santos<sup>1\*</sup>, M. Galo<sup>2</sup>, A. C. Carrilho<sup>1</sup>

<sup>1</sup> São Paulo State University - UNESP, Graduate Program in Cartographic Sciences,  
Presidente Prudente, São Paulo, Brazil - renato\_cstos@hotmail.com, carrilho.acc@gmail.com

<sup>2</sup> São Paulo State University - UNESP, Dept. of Cartography,  
Presidente Prudente, São Paulo, Brazil – mauricio.galo@unesp.br

**KEY WORDS:** Building Boundary Extraction, LiDAR Data, Alpha Shape Algorithm, Delaunay Triangulation, Average Point Spacing, Point density.

### ABSTRACT:

The  $\alpha$ -shape algorithm is a very common option to extract building boundaries from LiDAR data. This algorithm is normally executed in 2D space considering a parameter  $\alpha$  as a binary classifier which controls the distinctiveness of points whether or not they belong to the object boundary. For point cloud data, this parameter is directly related to the local point density and the level of detail of building boundaries. Studies that have explored this concept usually consider a unique parameter  $\alpha$  to extract all buildings in the dataset. However, the point density can have a considerable variation along the point cloud and, in this case, the use of a global parameter may not be the best choice. Alternatively, this paper proposes a data-driven method that estimates a local parameter for each building. The method evaluation considered six test areas with different levels of complexity, selected from a LiDAR dataset acquired over the city of Presidente Prudente/Brazil. From the qualitative and quantitative analysis, it could be seen that the proposed method generated better results than when a global parameter is used. The proposed method was also able to withstand density variation among the LiDAR data, having a positional accuracy around 0.22 m, against 0.40 m of global parameter.

### 1. INTRODUCTION

LiDAR data has gained attention in several sciences, such as those related to the geodetic sciences, where the input data is considered for the extraction of different kinds of objects: buildings, trees, roads, power lines, planar surfaces and vehicles among other things. For cartographic purposes, the feature extraction is fundamental for digital terrain model representations and the acquisition and/or update of geographic information systems, which can be used as database in many practical applications.

In urban planning, building boundary extraction has an essential role due to the high demand of updated cartographic products in order to subsidize decision making. The automatic and semi-automatic extraction process has been researched by many authors (Kim and Habib, 2009, Galvanin and Dal Poz, 2012, Awrangjeb, 2016). The difficulty of developing a fully automatic method is related to the complexity of the buildings, for instance, buildings composed of concave and convex segments, inner and outer boundaries. The convex hull is a concept that has been explored for automatic building extraction from LiDAR data. Sampath and Shan (2007) proposed the modified convex hull formation algorithm by limiting the search space using a neighbourhood, which enables the extraction of some concavities in the shape. The challenge consists in the neighbourhood size determination, whereas the limitation corresponds to non-extraction of inner boundaries. In contrast, the alpha shape ( $\alpha$ -shape) has the ability to detect both inner and outer boundaries. Many works have been using this algorithm, as seen in Shen (2008), Jochem et al. (2009), Shen et al. (2011), Satari et al. (2012), He et al. (2014), Li et al. (2015), and Albers et al. (2016).

The  $\alpha$ -shape algorithm was introduced by Edelsbrunner et al. (1983), aiming at the extraction of the shape of a set of points in 2D space. It considers a parameter  $\alpha$ , responsible for selecting the straight segments that composes the object boundary. The challenge is associated with the determination of this parameter, since it depends directly on point cloud density and the desired level of detail.

According to Jochem et al. (2009), in the context of building boundary extraction from LiDAR data, the average point spacing is a good estimate of the parameter  $\alpha$ . Shen (2008), Shen et al. (2011), and He et al. (2014) consider a similar approach, as the value  $\alpha$  is obtained by multiplying the average point distance by a constant, which was determined empirically. Li et al. (2015) consider two  $\alpha$  values, one larger and the other smaller than the average point spacing. This method generates two boundaries for each building, and the application of a refinement process was necessary to obtain the final boundary. However, the use of a constant parameter to extract all buildings in the dataset might not be the best choice, since the point density can vary. There can be several causes for such point density variation: aircraft velocity change, recording of multiple returns (normally in trees and building edges), aircraft rotation change, and the presence of occlusions. Figure 1 presents examples of point density variation over the same dataset. Therefore, the use of a parameter  $\alpha$  that can withstand density variation becomes compelling.

To overcome this limitation, this paper proposes an adaptive approach to estimate a local parameter  $\alpha$  for each building instead for the entire dataset. The method assumes that point density does not vary significantly within each building. The

\* Corresponding author

input data consists of the set of points derived from the building roof. The method is divided into four steps. Firstly, a triangular mesh is generated over the input using the Constrained Delaunay Triangulation (CDT), then, a statistical procedure is executed to identify and eliminate long edges that are inconsistent with the spacing between the sampled points. In a third step, the parameter  $\alpha$  is estimated considering the average length of remaining edges. Finally, the  $\alpha$ -shape algorithm is performed considering the estimated parameter for each building.

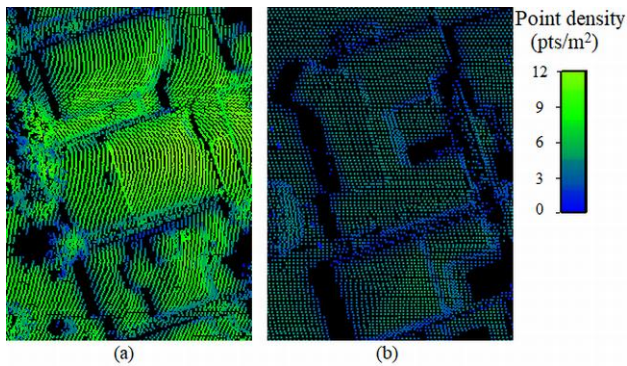


Figure 1. Point density variation on real data, from the same data set

## 2. METHOD

Figure 2 shows the simplified proposed method flowchart. The input data corresponds to the roof points of a given building, whereas the output is the building boundary. Considering that the main objective is the extraction of contours using a local parameter  $\alpha$ , the set of roof points of each building was selected manually using *CloudCompare* software, version 2.9.1. The method is composed of three steps for determination of a local parameter  $\alpha$ , followed by the execution of the  $\alpha$ -shape algorithm.

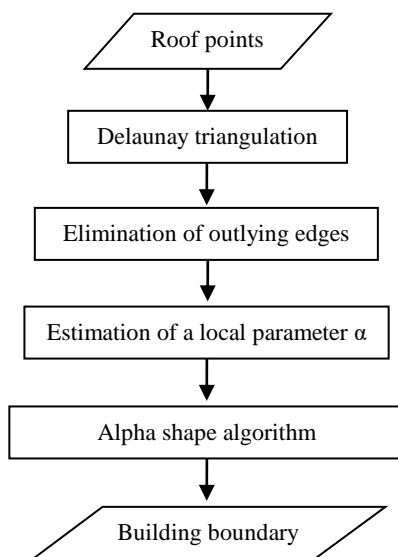


Figure 2. Flowchart of the steps involved in the local parameter  $\alpha$  estimation, used to perform the  $\alpha$ -shape algorithm

### 2.1 Local Alpha Parameter Estimation

The first step consists of the CDT over the building roof points. As the point cloud derived from the LASER scanning is represented in 3D space, the altimetric component was omitted in the process. The triangulation process can be impaired when there are points located in the same position. In order to eliminate the duplicated points, a pre-processing over roof points was performed using the *lasduplicate* tool from the *LAStools* package (Isenburg, 2018).

In this work, the local parameter  $\alpha$  estimation is based on the average point spacing which is not constant over the 3D point cloud. Before the estimation step, a statistical procedure is executed to detect and eliminate some lines obtained by the CDT located outside the outlines of the buildings. An example of a CDT considering a set of points sampled over a building is presented in Figure 3. Some long edges which can affect the determination of the average point spacing can be seen in Figure 3a.

Assuming the length of edges is normally distributed, a simple criterion to discard these long edges is considered, which is based on the average length of the edges ( $e_{avg}$ ) and their standard-deviation ( $s_e$ ). Edges longer than the average plus three standard-deviations are outlier candidates, and can be detected as follows:

If  $e_i < |3s_e + e_{avg}|$ , then  $e_i$  is not an outlier;  
 Otherwise,  $e_i$  is considered an outlier.

where  $e_i$  is the length of the edge  $i$ .

The parameter  $\alpha$  of a given building  $j$  is computed by means of Equation 1. In this equation, only the edges marked as non-outliers are considered in the calculation. Figure 3b presents an example which used the above criterion to eliminate the edges considered as outliers.

$$\alpha_j = 1/n \sum e_i \quad (1)$$

where  $\alpha_j$  = estimated parameter for the building  $j$   
 $n$  = number of edges considered  
 $e_i$  = length of the edge  $i$  that satisfies the criterion

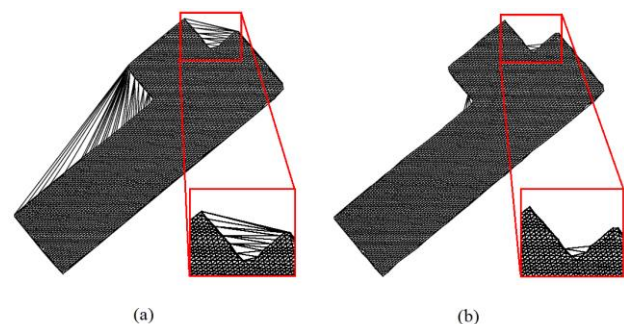


Figure 3. Representation of triangular mesh before (a) and after (b) the execution of procedure for the detection and elimination of outliers

### 2.2 Alpha Shape Algorithm

The  $\alpha$ -shape algorithm (Edelsbrunner et al., 1983) has been used in many tasks to obtain the approximate building boundaries. The general idea consists of the selection of

straight-line segments that compose the boundary. For that, the 2D set of points and the parameter  $\alpha$  are considered as input. In this work, the set of points corresponds to the LiDAR points sampled over the building roof, and the parameter  $\alpha$  is estimated considering the proposed method (Subsection 2.1).

Considering the points  $p_i$  and  $p_j$ , and the parameter  $\alpha$  as the radius of a circle passing through these points, it is possible to calculate two circles, as shown in Figure 4. If no point exists inside at least one circle, then points  $p_i$  and  $p_j$  are considered as edge points and can be connected to obtain a boundary segment (Figure 4a). If this is not the case, points  $p_i$  and  $p_j$  do not form a boundary segment (Figure 4b).

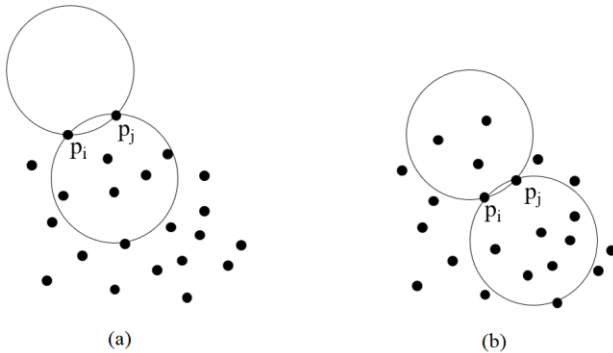


Figure 4. Alpha shape algorithm criterion. Points  $p_i$  and  $p_j$  form a boundary segment (a), whereas  $p_i$  and  $p_j$  do not form a boundary segment (b).

### 2.3 Quality Assessment

In this work, the building boundaries assessment was performed both qualitatively and quantitatively. The qualitative analysis was carried out by means of a visual inspection, whereas the quantitative analysis was performed using some quality measures: completeness (*Comp*), correctness (*Corr*),  $F_{score}$ , Hausdorff distance (*H*), and the PoLiS metric. Considering these quality indicators, the following references are suggested: Wiedemann et al. (1998), Sokolova et al. (2006) and Avblj et al. (2015).

Assuming an extracted polygon (*A*) and the reference polygon (*B*), the parameters of completeness, correctness and  $F_{score}$  are computed by means of the following equations:

$$Comp = ar(TP) / ( ar(TP) + ar(FN) ) \quad (2)$$

$$Corr = ar(TP) / ( ar(TP) + ar(FP) ) \quad (3)$$

$$F_{score} = (2 Comp Corr) / (Comp + Corr) \quad (4)$$

where  $ar(.)$  = area measurement

$$ar(TP) = A \cap B$$

$$ar(FN) = ar(B) - A \cap B$$

$$ar(FP) = ar(A) - A \cap B$$

*TP* = true positive

*FN* = false negative

*FP* = false positive

The Hausdorff distance  $H(A, B)$  between two sets *A* and *B* is defined as follows (Huttenlocher et al., 1993):

$$H(A, B) = \max(h(A, B), h(A, B)) \quad (5)$$

where

$$h(A, B) = \max \min \|a - b\| \quad (6)$$

and  $\|a - b\|$  is the Euclidean distance between points *a* and *b*, being  $a \in A$  and  $b \in B$ .

The PoLiS metric  $p(A, B)$  between two polygons *A* and *B* is defined by Avbelj et al. (2015) as:

$$p(A, B) = \frac{1}{2q} \sum_{a_j \in A} \min_{b \in \partial B} \|a_j - b\| + \frac{1}{2r} \sum_{b_k \in B} \min_{a \in \partial A} \|b_k - a\| \quad (7)$$

where *q* and *r* correspond to the number of vertices of polygons *A* and *B*, respectively.  $\partial A$  and  $\partial B$  denote the boundary of polygons *A* and *B*, respectively.

## 3. RESULTS AND DISCUSSION

### 3.1 Experiments and Results

This section presents the experiments and results obtained using the proposed method. The experiments were performed considering a LiDAR dataset acquired over the city of Presidente Prudente/Brazil by the company *Sensormap Geotecnologia*, which belongs to the *Engemap Group*, as can be seen in Tommaselli et al. (2018). The data set was acquired from a flight height of 900 m with an average density of 5 points/m<sup>2</sup>. The airborne LASER scanning system used was a RIEGL LMS-Q680i. This system, which has a precision of up to 2 cm in the range measurement, uses the principles of LiDAR to measure distances. It has a LASER pulse repetition rate of up to 400 kHz and an effective measurement rate of up to 266 kHz at a 60° scan angle. The system also has the ability to store multiple returns. Two clippings of areas from the dataset used are presented in Figure 5.

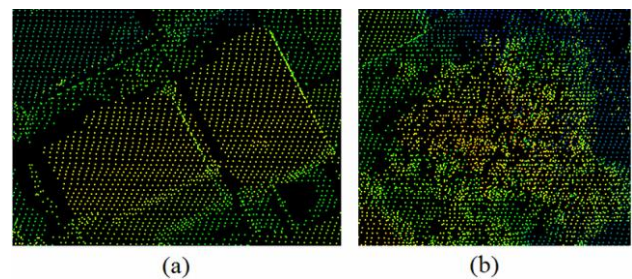


Figure 5. Two clippings of areas from the dataset used. Points sampled over buildings and ground (a), and vegetation (b)

The reference building boundaries were obtained from stereo restitution using digital aerial images of 80 megapixel acquired by one Phase One medium format camera (model iXA 180), with an average ground sample distance (GSD) of 10 cm.

To verify the performance of the proposed method, the building boundaries were extracted following two strategies. In the first strategy, the  $\alpha$ -shape algorithm was performed using a global parameter  $\alpha$ , which corresponds to the average point spacing of the dataset. In this case, the average point spacing was calculated considering all the points of the cloud point, that is, points sampled over vegetation, ground, and other anthropic objects. For the second strategy, the proposed method was used to estimate a local parameter  $\alpha$  for each building by means of Equation 1. Both strategies were implemented in C



programming language using *Code::Blocks* version 16.01. The kd-tree storage structure and search functions, which are implemented in the Fast Library for Approximate Nearest Neighbors - FLANN library (Muja and Lowe, 2013) version 1.8.0, were used to perform neighbourhood search operations.

The experiments were performed considering six different buildings (Figure 6) from the same dataset. Building 1 corresponds to a simple convex building of rectangular shape (Figure 6a). Building 2 has a rectangular shape and is composed of a small concave region (Figure 6b). Building 3 has a greater complexity, being formed by many concave regions (Figure 6c). Building 4 has a high complexity, its contour being composed of convex and concave segments (Figure 6d). In addition, there is a little gap on the left side, as can be observed in the profile generated from LiDAR data (Figure 6e). Building 5 also has some complexity, as its design includes both outer and inner boundaries (Figure 6f). Building 6 (Figure 6g) is located in a region of overlapping LiDAR strips, and a change of point density can be noted in the building (Figures 7k and 7l).

The building boundaries extracted using the  $\alpha$ -shape algorithm are presented in Figure 7. The results related to the global parameter  $\alpha$  are presented in Figures 7a, 7c, 7e, 7g, 7i, and 7k (left side), whereas the results considering the method proposed are shown in Figures 7b, 7d, 7f, 7h, 7j and 7l (right side).

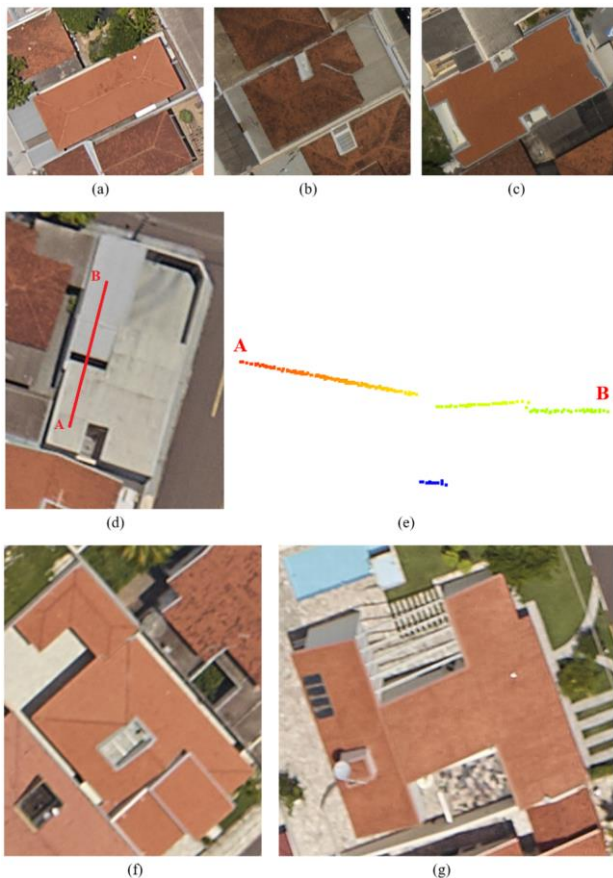


Figure 6. Images of the buildings selected for the experiments (a-d, and f-g) and the profile generated from LiDAR data considering the segment A-B drawn on Building 4 (e)

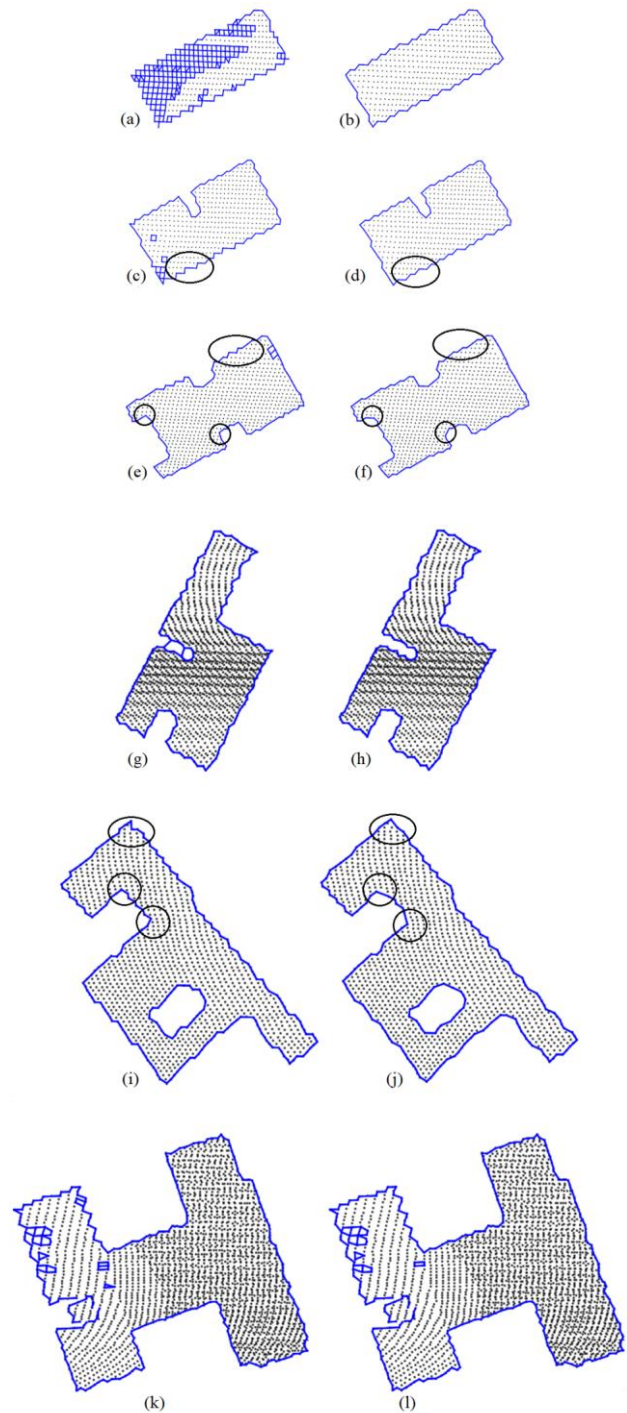


Figure 7. Building boundaries extracted using the  $\alpha$ -shape algorithm. Results considering a global parameter (a, c, e, g, i, k – left side), and results considering a local parameter estimated for each building (b, d, f, h, j, l – right side)

The quality parameters estimated for the different strategies are presented in Table 1. Two values of  $\alpha$  are shown for each building. The first corresponds to the global parameter ( $\alpha=0.45$  m, same value for all buildings), whereas the second corresponds to the local parameter, which was estimated considering the proposed method. The quality parameters were computed considering the Equations 2-7, which were described in subsection 2.3. In Table 1, the first column corresponds to the building identification, for example, building 1 is represented by B1 and so on. The completeness, correctness and

$F_{score}$  parameters are presented in the third, fourth and fifth columns, whereas the PoLiS and Hausdorff distance metric are presented in the sixth and seventh columns. Some fields are missing from the table (cells with a hyphen), since the algorithm failed to yield a unique closed polygon using those  $\alpha$  parameters and, as consequence, the computation of completeness, correctness and  $F_{score}$  parameters was not possible. The reference building boundaries were manually-derived on aerial images using a photogrammetric restitution procedure using the ERDAS IMAGINE LPS system.

Building	Strategy	Comp (%)	Corr (%)	$F_{score}$ (%)	PoLiS (m)	H (m)
B1	$\alpha=0.45$ m	-	-	-	1.18	4.12
	$\alpha=0.79$ m	95.1	99.3	97.1	0.26	0.53
B2	$\alpha=0.45$ m	-	-	-	0.33	2.36
	$\alpha=0.71$ m	93.7	99.4	96.5	0.24	0.95
B3	$\alpha=0.45$ m	-	-	-	0.19	0.78
	$\alpha=0.62$ m	96.8	98.7	97.8	0.19	0.55
B4	$\alpha=0.45$ m	-	-	-	0.15	0.57
	$\alpha=0.40$ m	95.1	98.1	96.6	0.14	0.57
B5	$\alpha=0.45$ m	94.8	98.6	96.7	0.19	0.78
	$\alpha=0.58$ m	94.4	98.1	96.0	0.17	0.68
B6	$\alpha=0.45$ m	-	-	-	0.32	3.08
	$\alpha=0.47$ m	-	-	-	0.31	2.80

Table 1. Quality parameters for different buildings and strategies

### 3.2 Discussion of Results

By comparing the local parameter estimated for each building with the global parameter estimated for the cloud point, i. e.  $\alpha=0.45$  m (Table 1), it is possible to verify that the latter is smaller for all cases, except for building 4. The reason for the  $\alpha$  value being smaller is directly related to the use of all points in the estimation process, including multiple returns, which usually occurs in border and vegetation regions, as can be seen in Figure 5. In contrast, the local parameter is estimated considering only the roof points of the building of interest.

As mentioned, the buildings presented in Figure 7 were selected from the same dataset. Analyzing the values of  $\alpha$  computed by means of the proposed method, it can be seen that each building had a different local parameter. The difference between the highest and lowest values is approximately 0.40 m. This difference indicates the point density variation along the dataset and reinforces the idea of the use of a parameter  $\alpha$  for each building.

It can be seen that the results obtained for building 1 (Figures 7a and 7b) using the global parameter were not satisfactory, since many incorrect segments were extracted. This problem occurred due to the point spacing on building 1 which is much larger than the global parameter estimated. On the other hand, the use of a local parameter overcame this problem.

Some incorrect segments were also extracted when the global parameter was used on building 2 (Figures 7c and 7d). The extracted boundary also presented a more irregular appearance (shape with aliasing), as highlighted in Figures 7c and 7d, since the global parameter is smaller. In contrast, the use of the local parameter produced more regular boundaries and did not extract incorrectly in the middle of the building, as observed in Figure 7d.

In building 3 (Figures 7e and 7f) only a few incorrect segments were extracted when the global parameter was considered. Despite this limitation, the global parameter generated better results in the extraction of concave segments. However, in the convex regions, the use of the local parameter produced better results.

It can be seen in Figures 7g and 7h that the use of the global parameter was not able to extract the small concave region located on the left side of building 4. In contrast, the use of the local parameter extracted this detail, since the estimated local parameter  $\alpha=0.40$  m is smaller than the global parameter ( $\alpha=0.45$  m).

As can be observed visually in building 5 (Figures 7i and 7j), the results using a global and local parameter are very similar, having only a few discrepancies. The global parameter generated better results in some concave regions, whereas the local parameter produced better results in some convex regions, as highlighted in Figures 7i and 7j. These results are directly related to the fact that the use of a smaller  $\alpha$  extracts more details, whereas a higher  $\alpha$  generated a smoother boundary.

In building 6 (Figures 7k and 7l), it can be clearly observed that the point distance in this building changes a lot. It is possible to observe from the results that neither the global nor the local parameter generated good results, since many incorrect segments were extracted. These results indicate that neither the global nor the local parameters work well when there is a significant density variation along the building.

In terms of completeness, correctness and  $F_{score}$ , it is not possible to perform a comparative analysis between the two strategies, since for most cases, when the global parameter was considered, it was not possible to calculate the quality parameters, as can be seen in Table 1. Considering the local parameter, the average values of completeness, correctness and  $F_{score}$  were 95%, 98% and 97%, respectively. These results indicate a high superposition between the extracted and reference boundaries.

Considering the PoLiS metric (Table 1), the proposed method presented average value of 0.22 m, against 0.40 m for the global parameter. For the global parameter, the worst case presented a PoLiS value of 1.18 m (B1), whereas the best case presented a PoLiS value of 0.15 m (B4). In contrast, the proposed method presented values of 0.31 m and 0.14 m for the worst (B6) and best (B5) case, respectively. Analysing the Hausdorff distances (Table 1), it can be seen that the average value for the global parameter is around 1.95 m, against 1.01 m for the proposed method. The lower value of the Hausdorff distance is always related to the use of the local parameter. From this, it can be concluded that the proposed method is suitable for extracting building roof boundaries.

In summary, the qualitative analysis indicates that the proposed method can be used to estimate the parameter  $\alpha$ , necessary in the extraction of building boundaries with the  $\alpha$ -shape algorithm. The results showed an improvement when compared with the use of a global parameter.

## 4. CONCLUSION

This paper proposes a data-driven approach to estimate a per-building parameter  $\alpha$ , that is, a local  $\alpha$ -shape estimation. This adaptive method makes use of the average point spacing within

each building domain. The main advantage of the proposed method lies in the determination of a parameter that can withstand density variation in the LiDAR data.

The results using real data show that the proposed method generated more detailed boundaries even for complex buildings composed of convex and concave segments, and with inner boundaries. Additionally, it was observed that if a considerable density variation occurs within a building, a single alpha value might be insufficient.

As for future research, the application of this method is suggested considering the same area with different point density. The development of a method that can overcome the problem highlighted in Section 3.2, for situations where there is a significant density variation within a single building, is also recommended.

### ACKNOWLEDGEMENTS

The authors would like to thank *Sensormap Geotecologia* for providing the LiDAR data used in the experiments, the São Paulo Research Foundation - FAPESP (grant n° 2016/12167-5) for financial support and National Council for Scientific and Technological Development - CNPq (304189/2016-2).

### REFERENCES

Albers, B., Kada, M., and Wichmann, 2016. Automatic extraction and regularization of building outlines from airborne lidar point clouds. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Prague, Czech Republic, pp. 555-560. DOI: <https://doi.org/10.5194/isprsarchives-XLI-B3-555-2016>.

Avbelj, J., Muller, R., and Bamler, R., 2015. A metric for polygon comparison and building extraction evaluation. *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 1, pp. 170-174. DOI: <https://doi.org/10.1109/LGRS.2014.2330695>.

Awrangjeb, M., 2016. Using point cloud data to identify, trace, and regularize the outlines of buildings. *International Journal of Remote Sensing*, v. 37, n. 3, pp. 551-579.

Edelsbrunner, H., Kirkpatrick, D., and Seidel, R., 1983. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4), pp. 551-559.

Galvanin, E. A. S., and Dal Poz, A. P., 2012. Extraction of building roof contours from LiDAR data using a Markov-Random-Field-Based approach. *IEEE Transactions on Geoscience and Remote Sensing*, v. 50, n. 3, pp. 981-987.

He, Yuxiang., Zhang, C., and Fraser, C. S., 2014. An energy minimization approach to automated extraction of regular building footprints from airborne LiDAR data. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3, pp. 65-72. DOI: <https://doi.org/10.5194/isprsnals-II-3-65-2014>.

Huttenlocher, D. P., Klanderman, G. A, and Rucklidge, W. J, 1993. Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 9.

Isenburg, M. 2018. LAStools: Software for rapid converting, filtering, viewing, gridding, and compressing of LiDAR. <http://www.cs.unc.edu/~isenburg/lastools/> (19 June 2018)

Jochem, A., Höfle, B., Rutzinger, M., and Pfeifer, N., 2009. Automatic roof plane detection and analysis in airborne LiDAR point clouds for solar potential assessment. *Sensors*, 9(7), pp. 5241-5262.

Kim, C., and Habib, A., 2009. Object-based integration of photogrammetric and LiDAR data for automated generation of complex polyhedral building models. *Sensors*, v. 9, pp. 5679-5701.

Li, Y., Gao, G., Cao, B., Zhong, L., and Liu, Y, 2015. Building boundaries extraction from point clouds using dual-threshold alpha shapes. In: *23rd International Conference on Geoinformatics 2015*. Wuhan, China, pp. 1-4. DOI: <https://doi.org/10.1109/GEOINFORMATICS.2015.7378619>.

Muja, M., and Lowe, Q. 2018. FLANN – Fast Library for Approximate Nearest Neighbours, Version 1.8.0, <https://www.cs.ubc.ca/research/flann/> (11 June 2018).

Satari, M., Samadzadegan, F., and Azizi, A., 2012. A multi-resolution hybrid approach for building model reconstruction from LiDAR data. *The Photogrammetric Record*, 27(139), pp. 330-359.

Sampath, A., and Shan, J, 2007. Building boundary tracing and regularization from airborne LiDAR point clouds. *Photogrammetric Engineering and Remote Sensing*, v. 73, n. 7, pp. 805-812.

Shen, W., 2008. Building boundary extraction based on LiDAR point clouds data. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Beijing, China, Vol. XXXVII, Part B3p, pp. 157-162.

Shen, W., Zhang, J., and Yuan, F., 2011. A new algorithm of building boundary extraction based on LiDAR data In: *19th International Conference on Geoinformatics*, Shanghai, China, pp. 1-4. <https://doi.org/10.1109/geoinformatics.2011.5981049>

Sokolova, M., Japkowicz, N., and Szpakowicz, S, 2006. Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation. In: *Proceedings of the AAAI'06 workshop on Evaluation Methods for Machine Learning*. p. 24–29.

Tommaselli, A. M. G., Galo, M., Reis, T. T., Ruy, R. S., Moraes, M. V. A., and Matricardi, W. V., 2018. Development and assessment of a data set containing frames images and dense airborne LASER scanning point clouds. *IEEE Geoscience and Remote Sensing Letters*, v. 15, n. 2, pp. 192-196. DOI: <https://doi.org/10.1109/LGRS.2017.2779559>

Wiedeman, C., Heipke, C., Mayer, H., and Jamet, O, 1998. Empirical evaluation of automatically extracted road axes, in *Empirical Evaluation in Computer Vision*. Wiley-IEEE Computer Society Press, pp. 172-187.