PH-shape: an adaptive persistent homology-based approach for building outline extraction from ALS point cloud data

Gefei Kong & Hongchao Fan

Published online: 05 Dec 2023.

Submit your article to this journal ⍐

Article views: 315

View related articles ⍐

View Crossmark data ⍐

Taylor & Francis
Taylor & Francis Group

PH-shape: an adaptive persistent homology-based approach for building outline extraction from ALS point cloud data

Gefei Kong and Hongchao Fan

Department of Civil and Environmental Engineering, Norwegian University of Science and Technology, Trondheim, Norway

ABSTRACT

Building outline extraction from segmented point clouds is a critical step of building footprint generation. Existing methods for this task are often based on the convex hull and α-shape algorithm. There are also some methods using grids and Delaunay triangulation. The common challenge of these methods is the determination of proper parameters. While deep learning-based methods have shown promise in reducing the impact and dependence on parameter selection, their reliance on datasets with ground truth information limits the generalization of these methods. In this study, a novel unsupervised approach, called PH-shape, is proposed to address the aforementioned challenge. The methods of Persistence Homology (PH) and Fourier descriptor are introduced into the task of building outline extraction. The PH from the theory of topological data analysis supports the automatic and adaptive determination of proper buffer radius, thus enabling the parameter-adaptive extraction of building outlines through buffering and "inverse" buffering. The quantitative and qualitative experiment results on two datasets with different point densities demonstrate the effectiveness of the proposed approach in the face of various building types, interior boundaries, and the density variation in the point cloud data of one building. The PH-supported parameter adaptivity helps the proposed approach overcome the challenge of parameter determination and data variations and achieve reliable extraction of building outlines.

1. Introduction

The extraction of 2D building boundaries plays a crucial role in generating building footprint data, which is widely applied in various fields, from mapping and navigation to 3D modeling, urban planning, and public strategy (Boo et al. 2022; Hu et al. 2022; Robinson et al. 2022; Zhou et al. 2022). There have been many approaches proposed for automatic building boundary extraction from point cloud data (Awrangjeb 2016; Li et al. 2022). However, the automation of this task remains a challenge because of the complexity of the buildings – which usually include both concave and convex segments and even inner boundaries of holes (dos Santos, Galo, and Carrilho 2019).

The most common algorithms applied to the task of tracing building boundaries are the modified convex hull algorithm (Jarvis 1977) and the α-shape algorithm (Edelsbrunner and Mücke 1994). However, they are sensitive to parameter selection, such as neighbor radius and the α-value, and perform unstably when facing the variation of point density. In addition, the former also lacks the ability of tracing inner boundaries. Other approaches, such as deep learning-based and grid-based approaches, are in the face of the

problem of relying on datasets and the same problem of parameter sensitivity.

To address these issues of existing unsupervised methods for boundary extraction from point cloud data, it is beneficial to consider a new strategy that is more robust to noise and parameter choice and can adapt to the point density variation. As an emerging method for point cloud data analysis (Akai, Hirayama, and Murase 2021), Topological Data Analysis (TDA) can capture the topological features of point cloud data, thus facilitating insight into the structure of data (Surrel et al. 2022). Persistent Homology (PH) is a common method of TDA (Wasserman 2018). It can extract the topological features and their birth–death times in data (e.g. connective components and holes) at different scales (resolutions). This ability of PH has helped shape matching (Poulenard, Skraba, and Ovsjanikov 2018), surface reconstruction (Dong, Chen, and Lin 2022), loop closure detection (Akai, Hirayama, and Murase 2021), and a lot of other research. Furthermore, PH has shown its feasibility and robustness in the face of data with noise or imbalanced distributions (Turkes, Montufar, and Otter 2022). Thus, employing PH for the analysis and extraction of boundary information can enable the

establishment of a less-sensitive and parameter-adaptive approach with great generalization.

Hence, in this study, an adaptive approach for building outline extraction based on the theory of TDA, PH-shape, is proposed. The new approach achieves stable building boundary tracing with adaptive parameters by using PH and Fourier Descriptor (FD) for the point cloud data. The former is for preliminary extraction of building outlines, and the latter is for reducing the zig-zag phenomenon and obtaining a smoother outline result. The main contributions of this study are as follows:

(1) The incorporation of topology technique, PH, into the work of building outline extraction overcomes the challenge of empirical parameter determination in existing approaches. In addition, the application of PH also provides the adaptive ability of the proposed method in the face of point cloud data with different point densities and density variation.

(2) The new proposed utilization method of FD enables the preliminary simplification of extracted building outlines in an unsupervised and non-parameter manner. The smoother extraction result of building outlines reduces the difficulty of the following outline regularization step for building footprint generation.

This paper is organized as follows: Section 2 reviews the related work and background of this study. Section 3 outlines the workflow of the proposed PH-shape. Section 4 describes the data and design of experiments and presents the experiment results of PH-shape. Section 5 discusses the conclusion and the further work of the study.

2. Related work and background

2.1. Building outline extraction (from point cloud data)

In the context of automatic extraction of building outlines from point cloud data, convex hull algorithms and modifications are widely used. Sampath and Shan (2007) proposed a modified version of the convex hull algorithm to achieve the extraction of concave outlines by limiting the search space to a local rectangular neighborhood. This method was also applied to other related research (Dai et al. 2017; Herve 2008). Wang and Shan (2009) further improved this algorithm by iteratively classifying and removing non-boundary points, and Cao et al. (2017) improved it by introducing the minimum number of neighbors (*minPts*) for a border point and changing the parameter setting of point spacing. However, the convex hull algorithm-based methods are sensitive to

parameter selection (Wang and Shan 2009) and cannot achieve the tracing of inner boundaries. Li et al. (2022) used the multiple-return attribute of point cloud data and neighborhood analysis to extract building boundary points and then proposed a new recursive convex hull algorithm to achieve the outline extraction. However, the recursive convex hull algorithm still requires the empirical threshold to determine "spurious lines".

The α-shape-based algorithms are also common in this context and can extract inner holes. However, the determination of a proper α-value remains a problem and is usually empirical and varies from case to case. Many researchers linked the α-value selection with point spacing of the data. Dorninger and Pfeifer (2008) and Shahzad and Zhu (2015) used twice the mean point spacing as the α-value, while He, Zhang, and Fraser (2014) used 1.5 times the mean point spacing. dos Santos, Galo, and Carrilho (2019) proposed an adaptive algorithm to automatically estimate the α-value for each point, but a new parameter neighborhood radius was required.

Many other methods not based on convex hull and α-shape algorithms were also proposed. Awrangjeb (2016) applied Delaunay triangulation to achieve the building outline extraction with more reasonable parameters, but the setting of point neighborhood threshold for removing long triangulation edges was still based on experience and undiscussed. Other researchers (Awrangjeb and Fraser 2014; Mahphood and Arefi 2017; Zhou and Neumann 2009) generated 2D grids from point cloud data to achieve a higher-efficiency extraction of building outlines. However, the parameter determination of sampling resolution significantly affects the accuracy of extraction results and the information loss caused by the sampling cannot be avoided. Mahphood and Arefi (2022) and Kong, Fan, and Lobaccaro (2022) reduced the effect of parameter determination, but the former still requires a proper resolution parameter and the latter is a deep learning-based method that relies on the training dataset. The generalization of these methods is questionable.

2.2. Persistent homology

PH has been applied in many fields, such as topological space classification and shape matching (Carlsson 2020; Otter et al. 2017). The research for analyzing point cloud data by using PH is also increasing, such as topological pattern recognition (Carlsson 2014), object detection (Syzdykbayev and Karimi 2020), point cloud description (Beksi and Papanikolopoulos 2018), and shape segmentation (Wong and Vong 2021).

As mentioned in Section 1, PH can capture the birth and death times of topological structures in

data. As the foundation of PH, the homology associates a series of vector spaces $\{H_k(X)|k \in N\}$ (i.e. homology groups) to a topological space X, where the k dimensional homology group $H_k(X)$ corresponds to the k dimensional holes (features) in X (Feng and Porter 2021). Consequently, we can know that the $H_0(X)$ describes the path-connected components in X and the $H_1(X)$ captures the holes (i.e. cycles) present in X.

To find the PH of a point cloud data, the data should be first turned into a sequence of simplicial complexes (subcomplexes) at different scales, where a simplicial complex is a collection of simplices, and 0-simplex denotes vertex, 1-simplex denotes edge, 2-simplex denotes triangle, and so on. This turning is achieved by providing a distance function (e.g. an increasing sequence of circle buffer radius $\{r\}$ around the points) for filtration, and an example of filtration is shown in Figure 1(a). Then, because each subcomplex has recorded the topological features at their corresponding scales, the PH of this data can be obtained by computing the homology of each subcomplex (Feng and Porter 2021). When computing the PH, as the distance value increases, new topological features appear and then disappear. For example, as shown in Figure 1(a), when $2r$ increases from 0.60 to 0.91, a hole appears and then disappears. The two distance values corresponding to the appearance and disappearance of a topological feature represent the <birth, death> of this feature (Malott, Sens, and Wilsey 2020), which also marks the persistence time of this feature. For k-dimensional homology group H_k, the <birth, death> pairs extracted from all subcomplexes at this dimension reveal the evolution of the point cloud data at this specific dimension, which helps to identify and track the information and change of significant structures at this dimension. These pairs can be represented on a two-dimensional surface,

which is referred to as Persistence Diagram (PD) (Akai, Hirayama, and Murase 2021), and an example of PD is shown in Figure 1(b). In a PD, its x- and y-axes represent birth and death distances, respectively, and all <birth, death> pairs appear at the upper side of the diagonal because the death value is always larger than its corresponding birth value. Hence, we can visualize when points are connected with each other by plotting H_0 PD and when holes (cycles) appear and disappear in the point cloud data by plotting H_1 PD.

3. Methodology

Aiming to extract the building outlines, the proposed PH-shape determines the adaptive parameter based on the analysis of PH results. The adaptive parameter is then applied to extract building outlines. Afterward, the extracted building outlines are smoothed and simplified based on FD, resulting in the final output.

3.1. The overview of PH-shape

As described earlier and shown in Figure 2, the proposed PH-shape consists of two major modules: (1) the adaptive extraction of preliminary building outlines by using PH and (2) the adaptive simplification of the preliminary building outlines by using FD.

In PH-shape, the 2D coordinates of the segmented point cloud data for a single building are used as input (Figure 2(a)). In the first module, the 0- and 1-dimensional (d) PHs of the input point cloud data are computed at first (Figure 2(b)), and an adaptive buffer radius for the input points is determined based on the $0d$ and $1d$ PHs (Figure 2(c)). By using the adaptive buffer radius, the union buffer of the input points is computed and then shrunk (Figure 2(d)). With the assistance of the
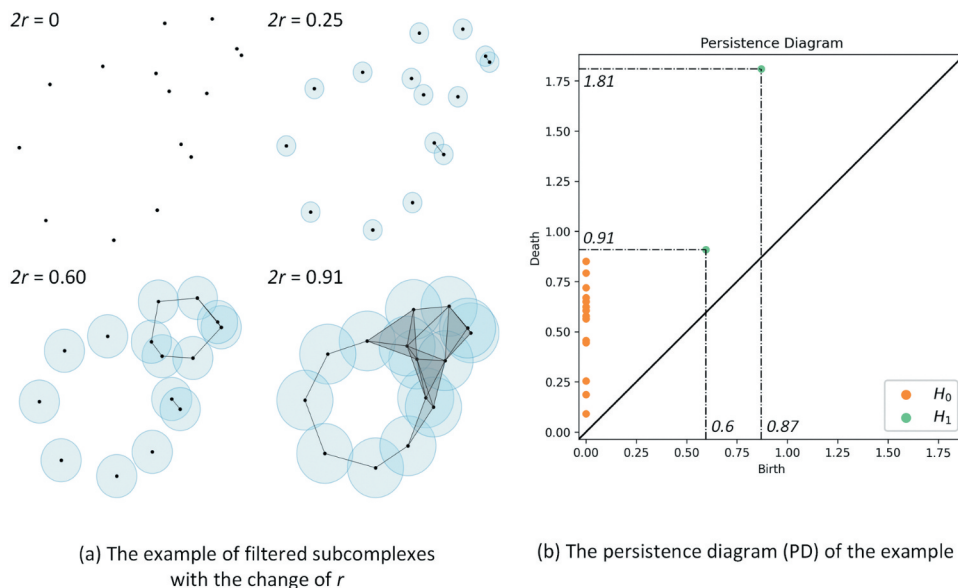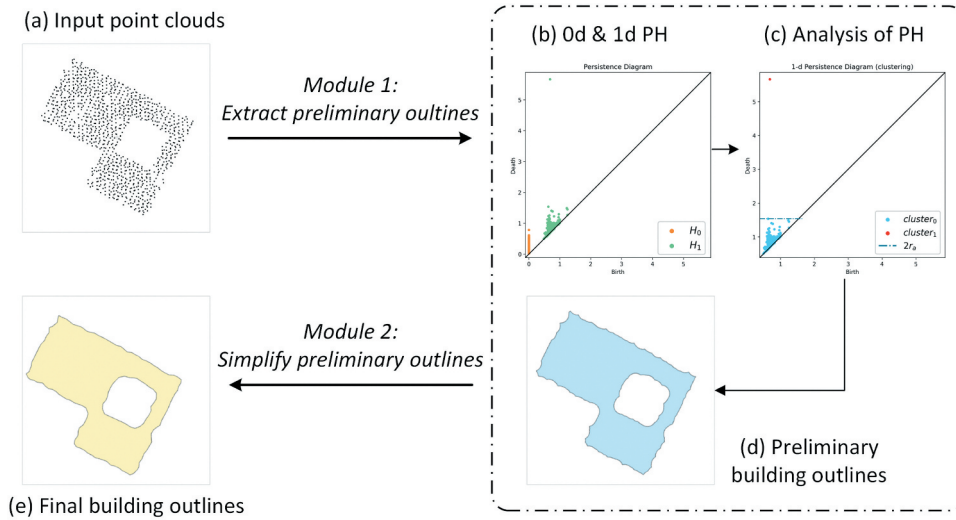

(a) The example of filtered subcomplexes with the change of r

(b) The persistence diagram (PD) of the example

Figure 1. The example of PH.

Figure 2. The overview of PH-shape.

union operation of each point's buffer, this process achieves the conversion of the point set to a polygon, and the exterior and interior boundaries of the polygon are regarded as the preliminary building outline result of the input data. After that, the polygon is input into the second module and its FD is computed. Then, the simplified polygon is extracted based on the FD and is ultimately output as the final extracted building outline (Figure 2(e)).

3.2. The extraction of preliminary building outlines

An example of the detailed process of module 1 is shown in Figure 3. The crucial step of this module is determining the adaptive buffer radius of the point cloud data. To obtain this adaptive radius, both the $0d$ PH and $1d$ PH of the point cloud data are computed at first, as shown in Figure 3(b). In this study, the Vietoris–Rips (VR) complex (Otter et al. 2017) is
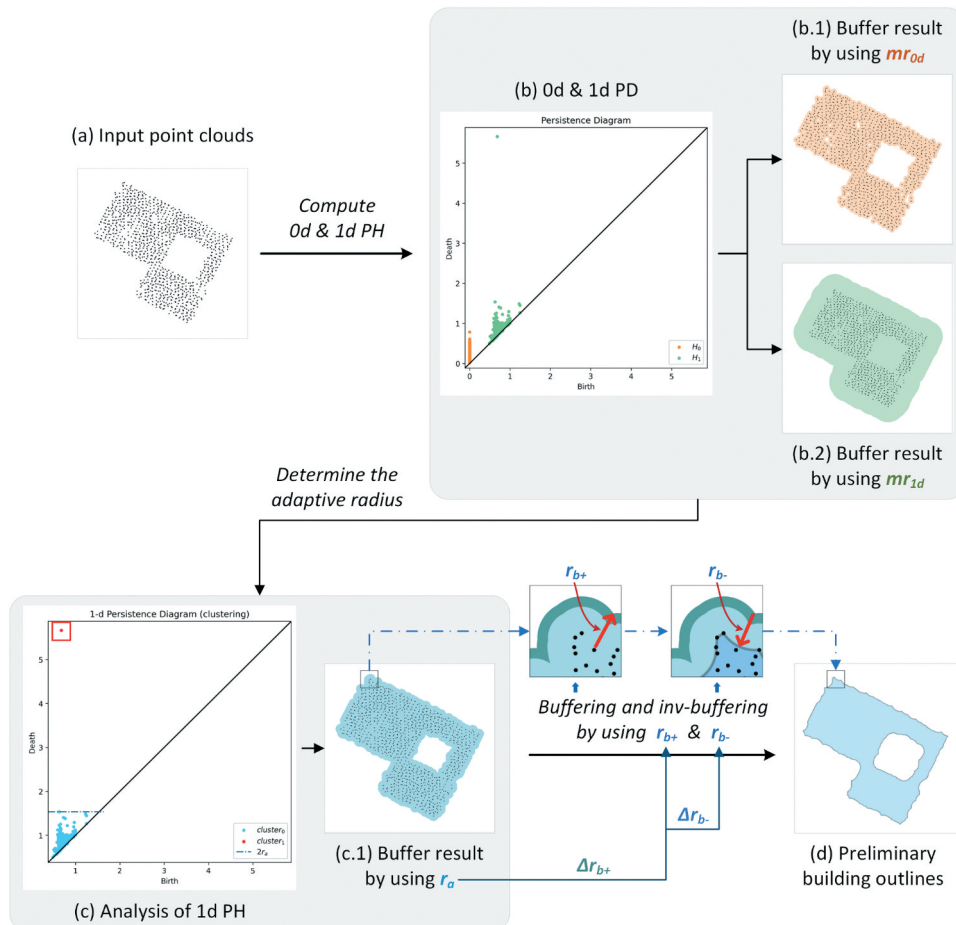
Figure 3. The detailed workflow of module 1: extracting preliminary building outlines.

chosen to build the simplicial complex for PH's filtration. By computing $0d$ PH, the distance when each point connected with its neighbors can be tracked, which is the "death" time of each <birth, death> pair in $0d$. The half of the maximum (max) death value in $0d$, mr_{0d}, corresponds to the buffer radius that the component covering all points generates, as shown in Figure 3(b.1). By computing $1d$ PH, the distances when inner holes disappear can be tracked, which also correspond to the "death" times in $1d$. The half of the max death value in $1d$, mr_{1d}, corresponds to the buffer radius that all inner holes disappear, as shown in Figure 3(b.2).

In general, the mr_{1d} can be regarded as the proper buffer radius to generate the union buffer of the input point data, and the shrunk exterior boundary of this union buffer can be output as the preliminary building outlines. However, the inner holes could not be traced in this situation as shown in Figure 3(b.2). Hence, the further analysis of the buffer radius is necessary, to choose a proper radius that can save the complete building outline with the significant inner holes. In this study, all holes can be identified by analyzing the persistence and death time of <birth, death> pairs in the $1d$ PH result. The longer persistence time with a larger death time represents a more significant feature disappearing slowly, meaning that this pair corresponds to the appearance and disappearance of a significant inner hole, as shown by the red dots in Figure 3(c). Correspondingly, the shorter persistence time and the smaller death time mean that the points related to these pairs have close neighbors and do not fill the inner holes, as shown by the blue dots in Figure 3(c). Based on this principle, the significant holes can be found by separating the <birth, death> pairs in $1d$ PH into several clusters based on the persistence and death times, and the detailed workflow for finding the adaptive radius is as follows:

(1) Compute the complex of the whole input point cloud data, and subsequently compute the $0d$ PH and $1d$ PH of the input point cloud data, and further extract mr_{0d} and mr_{1d} from them, as mentioned in the first paragraph of this subsection and shown in Figure 3(b). In this step, $0d$ PH and $1d$ PH consist of the <birth, death> pairs in their corresponding dimensions, which are denoted by D_k in this study. The detailed description of D_k is shown in Equation (1), where k denotes the dimension of PH and N_k denotes the number of pairs at k dimension.

$$D_k = \left\{ \langle birth, death \rangle_{k,i} | i \in [1, N_k] \in \mathbb{Z}^+, k \in \{0, 1\} \right\} \quad (1)$$

(2) Compute all persistence times $T_1 = \left\{ t_{1,i} = death_{1,i} - birth_{1,i} | i \in [1, N_1] \right\}$ of D_1 at first. Then, cluster T_1 and death times in D_1 by using the algorithm Density-Based Spatial Clustering of Applications with Noise (DBSCAN), where two DBSCANs' epsilons are adaptively set as mr_{0d}. The cluster with both minimum means of the persistence and death times is selected, which covers all topological features that are not inner holes. An example of this step is shown in Figure 3(c).

(3) The half of the max death value of the selected cluster in step (2) is computed and used as the final adaptive buffer radius result, which is denoted by r_a as shown in Figure 3(c.1).

After obtaining the adaptive buffer radius r_a of this roof's point cloud data, the union buffer polygon $Poly_{b+}$ of these points in 2D is computed by using a buffer radius of r_{b+} and union operation. The calculation of r_{b+} is shown in Equation (2):

$$r_{b+} = r_a + \Delta r_{b+}, \text{where } \Delta r_{b+} = \lceil 10 \times PS_{dt} \rceil / 10 \quad (2)$$

where Δr_{b+} is a tolerance for r_{b+}, which is equal to PS_{dt} denoting the point spacing of the dataset, and the operation of $\lceil 10 \times PS_{dt} \rceil / 10$ is used to round it up to one decimal place. The reason for adding the tolerance Δr_{b+} is that, due to the definition of VR complex, the filtration of complex will end when the balls of two neighbored points touch each other, so the r_a may be a few smaller than the radius required to fill all small gaps between neighbored points, especially when there is no inner hole that needs to be considered, and the size of these small gaps varies depending on the point density of the data. According to the simple example shown in Figure 4, we can note that after the buffering by using r_a, there would still be some small gaps (holes) in the union buffer, as shown by the red box area in Figure 4. Following the explanation of Δr_{b+}, as mentioned before, it can be adaptively set as the point spacing of the dataset PS_{dt}. In addition, it is rounded up to one decimal place to avoid the appearance of small gaps as much as possible and enhance the generalization of PH-shape by ignoring the slight variations in point densities across datasets.

Ultimately, in this module, $Poly_{b-}$ can be obtained by performing the inverse-buffering operation on $Poly_{b+}$ using the buffer radius r_{b-}. The "inverse-buffering operation" is defined as the buffering operation with a buffer radius smaller than 0 in this study. An example result of this step is shown in Figure 3(d). The exterior boundary and boundaries of inner holes of $Poly_{b-}$ are output as the preliminary building outline result. The calculation of r_{b-} is shown in Equation (3):
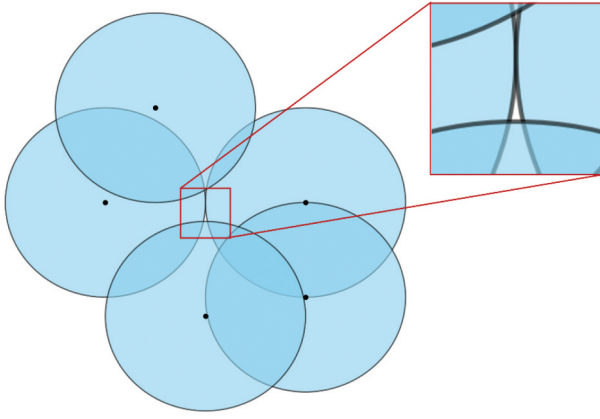
Figure 4. The example of the small gaps after buffering based on VR complex.

$$r_{b-} = -(r_{b+} - \Delta r_{b-}), \text{ where } \Delta r_{b-} = \left\lfloor 10 \times \frac{PS_{dt}}{3} \right\rfloor \bigg/ 10 \tag{3}$$

where Δr_{b-} denotes the tolerance distance of inverse buffer. This tolerance is added to consider the error in footprint labeling caused by the point spacing: the actual footprint boundary usually lies between the tightest boundary of the segmented point cloud data and a slightly larger buffer boundary of the tightest boundary, with a larger buffer radius of PS_{dt}. The setting of Δr_{b-} is based on $\frac{PS_{dt}}{3}$, because the distribution of actual footprint can be regarded as a normal distribution when its mean is at the tightest boundary and $\frac{PS_{dt}}{3}$ is its one standard deviation.

3.3. The simplification of building outlines

The simplification and smoothing of the preliminary building outlines are essential due to the inwardly sunken circle edges introduced by buffering. In this module, first, for the preliminary outline of each building, the FD of each exterior or interior boundary in the outline is computed to support the simplification and smoothing. Subsequently, the simplification result is obtained by extracting the boundary shape formed by the top-m coordinates in each FD. This extraction is performed when the similarity between the top-m FD coordinates and the FD's corresponding boundary arrives at a specific adaptive threshold. The detail of this module is described as follows:

(1) For the preliminary outline of each building including exterior and interior boundaries, define its each boundary as o_p and extract the coordinate set $\{(x_m, y_m)|m \in [1, M]\}_p$ of o_p. In the mathematical representation, p denotes this

boundary's index and M is the number of (x, y) coordinates in a boundary o_p.

(2) For each boundary o_p, compute the FD FD_p of the coordinate set $\{(x_m, y_m)\}_p$. To achieve the computation, first, the coordinate set is reorganized to a sequence of complex numbers $S_p = \{s_m = x_m + jy_m | m \in [1, M]\}_p$, where j denotes the imaginary unit of s_m. FD_p can be subsequently obtained by computing the $1d$ discrete Fourier transform (DFT) of S_p. The low-frequency components in FD_p present the global shape features of o_p, and the high-frequency components describe the details of the shape.

(3) Iteratively truncate and reconstruct top-m low-frequency FD_p from $m = 1$ to M until $m=M$ or the similarity (sim) between the boundary o_p and its simplified result $o_{p,m}^s$ formed by the m coordinates reconstructed from the top-m low-frequency FD_p achieving the specific threshold th_{sim}. The $o_{p,m}^s$ achieving the requirement of similarity, $o_{p,m*}^s$, is output as the simplification result of o_p. The truncation process is achieved by extracting top-m low-frequency components, and the reconstruction process is achieved by computing the $1d$ inverse DFT and separating real and imaginary parts and scaling of the top-m truncated complex components. The sim is defined by the Hausdorff Distance (HD) (Huttenlocher, Klanderman, and Rucklidge 1993). The smaller HD represents the higher sim, hence the iteration stops when $HD\left(o_{p,m}^s, o_p\right) \leq th_{sim}$. The th_{sim} is adaptive for each building and is calculated by Equation (4).

$$th_{sim} = -(1 - \cos 30°) \ r_{b-} = (1 - \cos 30°)(r_a + \Delta r_{b+} - \Delta r_{b-}) \tag{4}$$

This threshold value corresponds to the straight-boundary situation that the distance between the neighbored points at the actual boundary (e.g. the segment between c_m and c_{m+1}) is equal to the buffering radius r_{b-}, as illustrated in Figure 5.

(4) Obtain the simplification result of the preliminary outline for one building by repeating steps (2) and (3) for its each boundary.

Ultimately, by repeating the above steps for the outlines of all buildings, the simplification results of building outlines can be obtained. These simplified building outlines are output as the final result of the building outline extraction.
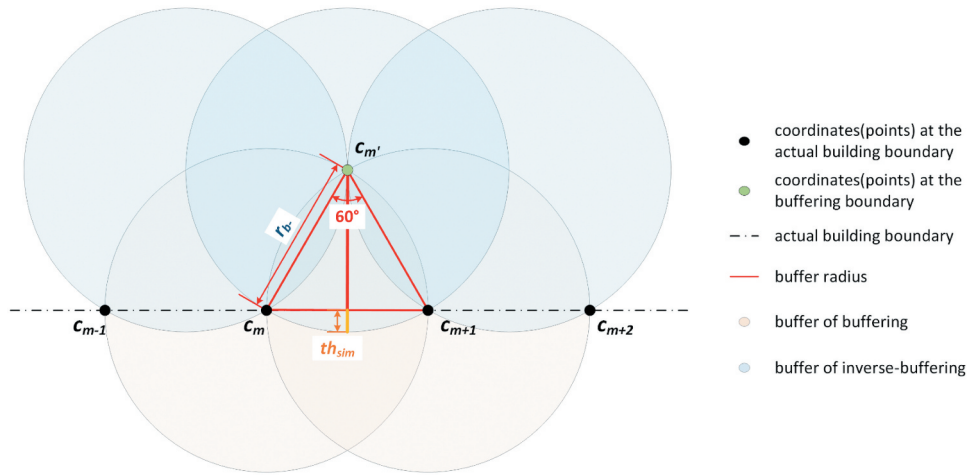
Figure 5. The geometrical meaning of th_{sim}.

4. Experiments

4.1. Experiment details

In this study, two datasets with different point densities are applied for the experiments. The first dataset is a custom dataset in Trondheim with a standard point density of 12–20 points/m^2. This dataset is named as "Trondheim" dataset in the following sections. As described in Kong, Fan, and Lobaccaro (2022), the original Airborne Laser Scanning (ALS) point cloud data of this dataset is provided by the mapping authority of Trondheim Municipality, and its corresponding ground truth footprints are from the national open geographical data provided by the Norwegian Mapping Authority (FKB-Buildings Dataset 2021). The second dataset is the International Society for Photogrammetry and Remote Sensing (ISPRS) benchmark dataset (Vaihingen) (Cramer 2010) with a lower point density of 4–6.7 points/m^2 (Cao et al. 2017). This dataset has contained both the point cloud data and the ground truth footprint information.

To compare with our previous work (Kong, Fan, and Lobaccaro 2022), the test data including 93 roofs from the Trondheim dataset and 34 roofs from the ISPRS dataset were used. General building footprint types with various shapes are covered by this test data, including rectangle-shape, L-shape, T-shape, more complex combined-shape, and so on. Buildings of various sizes are also covered in it.

There are many metrics for quantitatively evaluating the accuracy of extracted building outlines, such as mean intersection over union (mIoU), root-mean-square error, HD (Huttenlocher, Klanderman, and Rucklidge 1993), PoLiS (Avbelj, Müller, and Bamler 2015), and robust corner correspondence (Dey and Awrangjeb 2020). In this study, three metrics, mIoU, HD, and PoLiS, are applied for the quantitative evaluation. The mIoU is applied to evaluate the area similarity between the extracted building outlines

and the ground truth data. The HD and PoLiS are applied to evaluate the shape similarity between them by calculating the distance error between the corresponding two polygons. The calculation method of mIoU refers to the study of Kong, Fan, and Lobaccaro (2022), and the calculation methods of HD and PoLiS can refer to the corresponding cited papers.

The experiment on the two datasets is performed to evaluate PH-shape quantitatively and qualitatively on the two datasets, and its result is shown in Section 4.2. Furthermore, to evaluate the PH-shape more comprehensively, the quantitative comparative experiment on the two datasets is conducted to compare the performance of PH-shape with that of α-shape and generative adversarial network (GAN) -based methods (Kong, Fan, and Lobaccaro 2022). PH-shape and all the comparative strategies are implemented in Python. The Python version of Gudhi (Maria et al. 2014) is used to support the PH computation.

4.2. Experiment result of PH-shape

The quantitative and qualitative evaluation results of PH-shape on two datasets are shown in Table 1 and Figure 6, respectively. As shown in Table 1, on two datasets with different point densities, PH-shape achieves reliable performance. On the Trondheim dataset with relatively high point density (i.e. smaller point spacing), PH-shape achieves over than 90% of mIoU and lower than 1 m of HD. For the strict shape parameter, PoLiS, a quite good result of 0.17 m is achieved. On the ISPRS benchmark dataset with

Table 1. Quantitative evaluation result of PH-shape on datasets with different point densities.

Dataset	mIoU (%)	HD (m)	PoLiS (m)
Trondheim	93.66	0.60	0.17
ISPRS	85.36	1.57	0.38
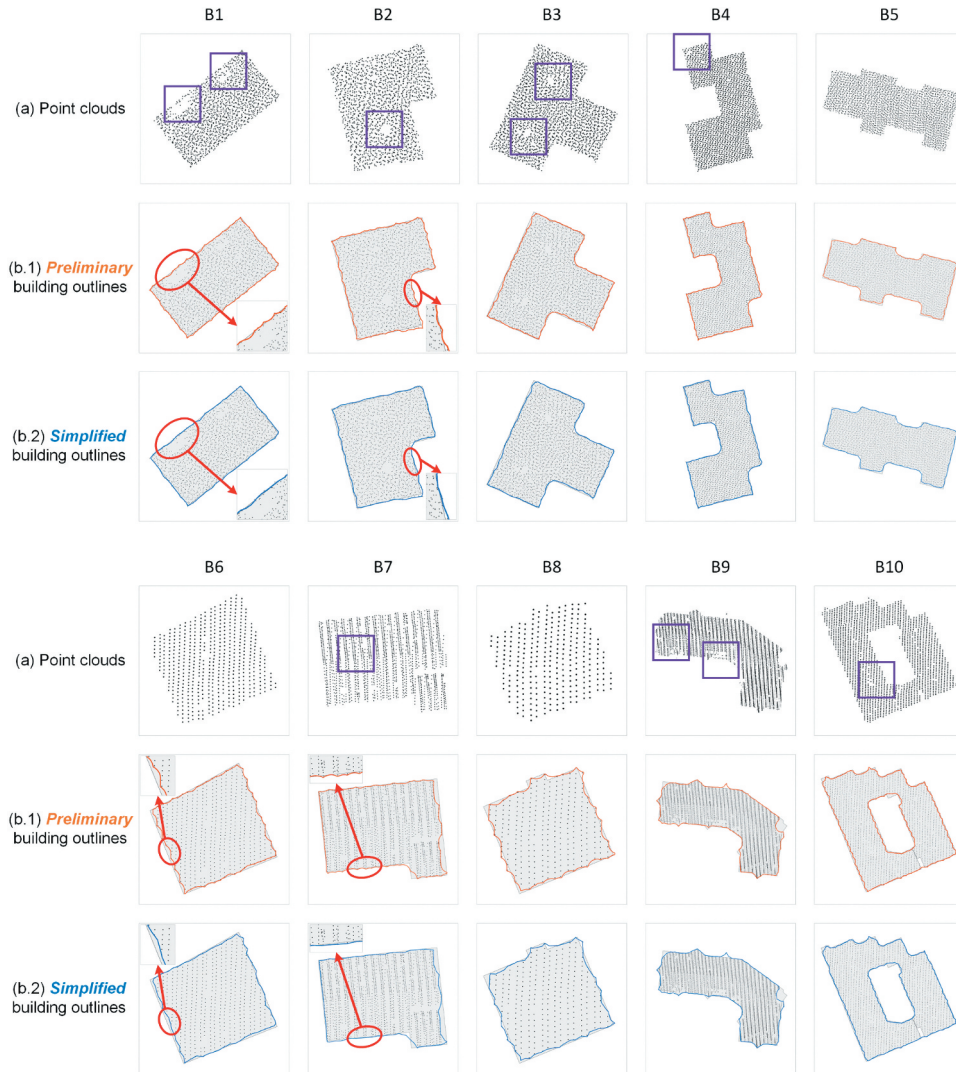
Figure 6. Qualitative evaluation result of PH-shape on datasets with different point densities (B1–B5: on the Trondheim dataset, B6–B10: on the ISPRS dataset).

a lower point density, although the performance of PH-shape reduces compared with that on the Trondheim dataset, it still achieves over than 80% of mIoU, 1.57 m of HD, and 0.38 m of PoLiS. The comparative quantitative result on the two datasets implies that PH-shape can achieve better performance on the dataset with higher point density. "The quantitative analysis of PH-shape will be discussed more in Section 4.3 based on the comparative experiment.

Through the qualitative evaluation result as shown in Figure 6, the performance of PH-shape is intuitively described. In Figure 6, (a) shows the input point cloud data with the highlighting of the density variation parts, (b.1) and (b.2) with different-colored lines show the results of building outlines extracted at the different steps (i.e. modules) of PH-shape. In addition, the black points in Figure 6 correspond to the segmented point cloud data of buildings and the gray polygons in Figure 6 correspond to the ground truth footprint data of buildings. Different building shapes are included in Figure 6. B1 and B6 are rectangle-shape; B2 and B7 are L-shape; B3 and B8 are

T-shape; and B4 and B5 and B9 and B10 correspond to the more complex combined shapes. As shown in Figure 6, the extraction result shown in B2–B5 and B7–B10 demonstrates that the concave parts of the point cloud data can be traced well and stably by PH-shape, even though these concave parts have different scales. B1–B4, B7, B9, and B10 exist the density variation in one building's point cloud data, as shown in Figure 6(a), where the purple boxes highlight the density variation part of these point clouds. The results of these buildings with density variation indicate that the PH-shape can address this issue successfully. In the case of B10, PH-shape further demonstrates its ability to trace the interior boundaries of building outlines. However, it is important to note that the actual building footprint of B10 is without inner holes. Despite this, when just considering the extraction of building outlines from the point cloud data, PH-shape actually performs the correct extraction and achieves a satisfactory result. The orange and blue lines shown in Figure 6(b.1-b.2) show the comparison between the preliminary building outlines and the

final building outlines, which are the results of module 1 and module 2 of PH-shape, respectively. The qualitative comparison result indicates that the simplification module (module 2) can effectively simplify and smooth the "zig-zag" parts of the preliminary building outlines, as shown by the exampled red circles and their "zoom-in" views in Figure 6. Overall, the qualitative evaluation result further demonstrates that PH-shape can achieve the accurate building outline extraction in the face of point cloud data with different densities and the density variation in one building's point clouds.

4.3. Comparative experiment result

The quantitative comparative result of two existing methods, α-shape and GAN, and our PH-shape is shown in Table 2. As shown in Table 2, on both datasets, PH-shape achieves better performance than the other two comparative methods. On the Trondheim dataset, compared with α-shape, the mIoU, HD, and PoLiS in PH-shape improve by 0.86%, 0.26 m, and 0.02 m, respectively; when compared with GAN, PH-shape also exhibits the improvement of mIoU and HD by 0.14% and 0.04 m, respectively. PH-shape achieves comparable and improved results for three metrics on the Trondheim dataset. In particular, the significant improvement of HD indicates that when extracting building outlines, PH-shape can more effectively preserve shape details and reduce the presence of significant undesirable burring segments.

On the ISPRS dataset, compared with α-shape, PH-shape improves mIoU, HD, and PoLiS by 2.26%, 0.26 m, and 0.03 m, respectively; and compared with GAN, PH-shape also improves these three metrics by 0.18%, 0.23 m, and 0.05 m, respectively. Compared with the results on the Trondheim dataset, besides the still significant improvement of HD, the increase of mIoU and decrease of PoLiS are also more significant than those on the Trondheim dataset. The more significant improvement on the ISPRS dataset is because that the Trondheim dataset has better data quality and higher point density. It also indicates that PH-shape can better adapt to the dataset with low point density than the comparative methods.

Furthermore, comparing the performance differences of the three methods on two datasets, the performance differences of PH-shape in mIoU, HD, and PoLiS are 8.30%, 0.97 m, and 0.21 m, respectively. By contrast, α-shape shows the differences of 9.70%, 0.97 m, and 0.22 m in these metrics and GAN shows the differences of 8.34%, 1.16 m, and 0.26 m, respectively. PH-shape performs smaller overall performance difference than the other two methods in the face of datasets with different point densities, which implies its better stability and generalization.

Overall, PH-shape can effectively and stably extract building outlines in the face of different point cloud data. Moreover, the comparative results also show the generalization capability of PH-shape, making it a robust and reliable method for the task of building outline extraction.

5. Conclusion and future work

In this study, PH-shape, an adaptive approach for extracting building outlines is proposed. The novel method PH for TDA is introduced into the task of extracting building outlines, to automatically and adaptively find a proper buffer radius for a segmented roof point cloud data. Based on this radius, the building outlines can be preliminarily extracted by buffering and "inverse" buffering. The final building outlines are ultimately obtained by smoothing and simplifying the preliminary building outlines based on FD and an adaptive HD-based threshold. The experiment results demonstrate that PH-shape can effectively and stably extract building outlines from the segmented point cloud data, no matter in the face of convex and concave building shapes, buildings with exterior and interior boundaries, and point cloud data with different point densities. Given the generalization and effectiveness of PH-shape shown in the experiments, it has the potential to extend beyond the task of building outline extraction to the broader task of boundary tracing. However, while PH-shape performs well on both low- and high-point-density datasets and shows better balance and generalization, the difference in performance between PH-shape and other strategies is not as significant on the higher-quality dataset (with higher point density) compared to the low-quality dataset. This suggests that for the dataset with high quality and density, PH-shape is just an alternative rather than the first choice compared to other methods. In the future, the study of a better simplification strategy is suggested to further improve the accuracy of extracted building outlines by PH-shape.

Table 2. Comparative result of existing strategies and PH-shape on two datasets with different point densities.

Dataset	Method	mIoU (%)	HD (m)	PoLiS (m)
Trondheim	α-shape	92.80	0.86	0.19
	GAN	93.52	0.64	**0.17**
	PH-shape	**93.66**	**0.60**	**0.17**
ISPRS	α-shape	83.10	1.83	0.41
	GAN	85.18	1.80	0.43
	PH-shape	**85.36**	**1.57**	**0.38**

The bold entries signify the best comparison results achieved in each dataset.
The italic entries present that the method mentioned in this row is the proposed method in this study.

Furthermore, experiments on point clouds with more various point densities and from different sources will be considered with the development of open-source datasets to further validate the practicality of PH-shape.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Notes on contributors

Gefei Kong received her master's degree in Cartography and Geography Information System at Wuhan University in China and is currently pursuing her PhD at Norwegian University of Science and Technology (NTNU). Her research interests include 3D city modelling, LiDAR data processing, and computer vision.

Hongchao Fan is professor for 3D Geoinformatics in the Department of Civil and Environmental Engineering at the Norwegian University of Science and Technology (NTNU). He received his master's degree in Geodesy and Geoinformatics at the University of Stuttgart and obtained his PhD at the Technical University of Munich in Germany. After that he worked as Group Leader for 3D Data Infrastructure at the Heidelberg University for six years. In 2018, he started his work as professor at NTNU in Trondheim, Norway. His research interests include 3D city modelling, spatial data mining from VGI data, and laser scanning.

ORCID

Hongchao Fan 🔟 http://orcid.org/0000-0002-0051-7451

Data availability statement

The authors confirm that the data and source codes used in this paper can be available upon individual request.

References

Akai, N., T. Hirayama, and H. Murase. 2021. "Persistent Homology in LiDAR-Based Ego-Vehicle Localization." In *2021 IEEE Intelligent Vehicles Symposium (IV)*, 889–896. https://doi.org/10.1109/IV48863.2021.9575312.

Avbelj, J., R. Müller, and R. Bamler. 2015. "A Metric for Polygon Comparison and Building Extraction Evaluation." *IEEE Geoscience & Remote Sensing Letters* 12 (1): 170–174, January. 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* 37 (3): 551–579, February. https://doi.org/10.1080/01431161.2015.1131868.

Awrangjeb, M., and C. S. Fraser. 2014. "An Automatic and Threshold-Free Performance Evaluation System for Building Extraction Techniques from Airborne LIDAR Data." *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing* 7 (10): 4184–4198, October. https://doi.org/10.1109/JSTARS.2014.2318694.

Beksi, W. J., and N. Papanikolopoulos. 2018. "Signature of Topologically Persistent Points for 3D Point Cloud Description." In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 3229–3234. https://doi.org/10.1109/ICRA.2018.8460605.

Boo, G., E. Darin, D. R. Leasure, C. A. Dooley, H. R. Chamberlain, A. N. Lázár, K. Tschirhart, et al. 2022. "High-Resolution Population Estimation Using Household Survey Data and Building Footprints". *Nature Communications* 13 (1). March. Nature Publishing Group: 1330. https://doi.org/10.1038/s41467-022-29094-x.

Cao, R., Y. Zhang, X. Liu, and Z. Zhao. 2017. "3D Building Roof Reconstruction from Airborne LiDar Point Clouds: A Framework Based on a Spatial Database." *International Journal of Geographical Information Science* 31 (7): 1359–1380. Taylor & Francis. https://doi.org/10.1080/13658816.2017.1301456.

Carlsson, G. 2014. "Topological Pattern Recognition for Point Cloud Data." *Acta numerica* 23: 289–368, May. https://doi.org/10.1017/S0962492914000051.

Carlsson, G. 2020. "Topological Methods for Data Modelling." *Nature Reviews Physics* 2 (12): 697–708, December. Nature Publishing Group. https://doi.org/10.1038/s42254-020-00249-3.

Cramer, M. 2010. "The DGPF-Test on Digital Airborne Camera Evaluation Overview and Test Design." *Photogrammetrie - Fernerkundung - Geoinformation* 2010 (2): 73–82, May. https://doi.org/10.1127/1432-8364/2010/0041.

Dai, Y., J. Gong, Y. Li, and Q. Feng. 2017. "Building Segmentation and Outline Extraction from UAV Image-Derived Point Clouds by a Line Growing Algorithm." *International Journal of Digital Earth* 10 (11): 1077–1097, November. Taylor & Francis. https://doi.org/10.1080/17538947.2016.1269841.

Dey, E. K., and M. Awrangjeb. 2020. "A Robust Performance Evaluation Metric for Extracted Building Boundaries from Remote Sensing Data." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13:4030–4043. https://doi.org/10.1109/JSTARS.2020.3006258.

Dong, Z., J. Chen, and H. Lin. 2022. "Topology-Controllable Implicit Surface Reconstruction Based on Persistent Homology." *Computer-Aided Design* 150:103308, September. https://doi.org/10.1016/j.cad.2022.103308.

Dorninger, P., and N. Pfeifer. 2008. "A Comprehensive Automated 3D Approach for Building Extraction, Reconstruction, and Regularization from Airborne Laser Scanning Point Clouds." *Sensors* 8 (11): 7323–7343, November. Molecular Diversity Preservation International. https://doi.org/10.3390/s8117323.

dos Santos, R. C., M. Galo, and A. C. Carrilho. 2019. "Extraction of Building Roof Boundaries from LiDar Data Using an Adaptive Alpha-Shape Algorithm." *IEEE Geoscience & Remote Sensing Letters* 16 (8): 1289–1293, August. https://doi.org/10.1109/LGRS.2019.2894098.

Edelsbrunner, H., and E. P. Mücke. 1994. "Three-Dimensional Alpha Shapes." *ACM Transactions on Graphics* 13 (1): 43–72, January. https://doi.org/10.1145/174462.156635.

Feng, M., and M. A. Porter. 2021. "Persistent Homology of Geospatial Data: A Case Study with Voting." *SIAM Review* 63 (1): 67–99, January. Society for Industrial and Applied Mathematics. https://doi.org/10.1137/19M1241519.

FKB-Buildings Dataset. 2021. Accessed March 5, 2022. https://kartkatalog.geonorge.no/metadata/fkb-bygning/8b4304ea-4fb0-479c-a24d-fa225e2c6e97.

He, Y., C. Zhang, and C. S. Fraser. 2014. "An Energy Minimization Approach to Automated Extraction of Regular Building Footprints from Airborne LiDAR Data." *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II–3: 65–72. Copernicus GmbH. https://doi.org/10.5194/isprsannals-II-3-65-2014.

Herve, L. 2008. *Outlining Buildings Using Airborne Laser Scanner Data*. Netherlands: Master Theis.

Hu, Y., C. Liu, Z. Li, J. Xu, Z. Han, and J. Guo. 2022. "Few-Shot Building Footprint Shape Classification with Relation Network." *ISPRS International Journal of Geo-Information* 11 (5): 311, May. Multidisciplinary Digital Publishing Institute. https://doi.org/10.3390/ijgi11050311.

Huttenlocher, D. P., G. A. Klanderman, and W. J. Rucklidge. 1993. "Comparing Images Using the Hausdorff Distance." *IEEE Transactions on Pattern Analysis & Machine Intelligence* 15 (9): 850–863, September. https://doi.org/10.1109/34.232073.

Jarvis, R. 1977. "Computing the Shape Hull of Points in the Plane." In *Proceedings of the IEEE Computing Society Conference on Pattern Recognition and Image Processing*, 231–241. Troy, NY.

Kong, G., H. Fan, and G. Lobaccaro. 2022. "Automatic Building Outline Extraction from ALS Point Cloud Data Using Generative Adversarial Network." *Geocarto International* 37 (27): 15964–15981, December. Taylor & Francis. https://doi.org/10.1080/10106049.2022.2102246.

Li, X., F. Qiu, F. Shi, and Y. Tang. 2022. "A Recursive Hull and Signal-Based Building Footprint Generation from Airborne LiDar Data." *Remote Sensing* 14 (22): 5892, January. Multidisciplinary Digital Publishing Institute. https://doi.org/10.3390/rs14225892.

Mahphood, A., and H. Arefi. 2017. "A Data Driven Method for Flat Roof Building Reconstruction from Lidar Point Clouds." *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-4/W4:167–172, September. https://doi.org/10.5194/isprs-archives-XLII-4-W4-167-2017.

Mahphood, A., and H. Arefi. 2022. "Grid-Based Building Outline Extraction from Ready-Made Building Points." *Automation in Construction* 139:104321, July. https://doi.org/10.1016/j.autcon.2022.104321.

Malott, N. O., A. M. Sens, and P. A. Wilsey. 2020. "Topology Preserving Data Reduction for Computing Persistent Homology." In *2020 IEEE International Conference on Big Data (Big Data)*, 2681–2690. https://doi.org/10.1109/BigData50022.2020.9378216.

Maria, C., J.-D. Boissonnat, M. Glisse, and M. Yvinec. 2014. "The Gudhi Library: Simplicial Complexes and Persistent Homology." In *Mathematical Software – ICMS 2014*, edited by H. Hong and C. Yap, 167–174. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer. https://doi.org/10.1007/978-3-662-44199-2_28.

Otter, N., M. A. Porter, U. Tillmann, P. Grindrod, and H. A. Harrington. 2017. "A Roadmap for the Computation of Persistent Homology." *EPJ Data Science* 6 (1): 1–38, December. SpringerOpen. https://doi.org/10.1140/epjds/s13688-017-0109-5.

Poulenard, A., P. Skraba, and M. Ovsjanikov. 2018. "Topological Function Optimization for Continuous Shape Matching." *Computer Graphics Forum* 37 (5): 13–25. https://doi.org/10.1111/cgf.13487.

Robinson, C., A. Ortiz, H. Park, N. L. Gracia, J. K. Kaw, T. Sederholm, R. Dodhia, and J. M. L. Ferres. 2022. "Fast Building Segmentation from Satellite Imagery and Few Local Labels." In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1462–1470. New Orleans, LA: IEEE. https://doi.org/10.1109/CVPRW56347.2022.00152.

Sampath, A., and J. Shan. 2007. "Building Boundary Tracing and Regularization from Airborne LIDAR Point Clouds." *Photogrammetric Engineering & Remote Sensing* 73 (7): 805–812, July. https://doi.org/10.14358/PERS.73.7.805.

Shahzad, M., and X. X. Zhu. 2015. "Reconstruction of Building Footprints Using Spaceborne Tomosar Point Clouds." *ISPRS Annals of the Photogrammetry, Remote Sensing & Spatial Information Sciences* II-3/W5:385–392, August. https://doi.org/10.5194/isprsannals-II-3-W5-385-2015.

Surrel, T. D., F. Hensel, M. Carrière, T. Lacombe, Y. Ike, H. Kurihara, M. Glisse, and F. Chazal. 2022. "RipsNet: A General Architecture for Fast and Robust Estimation of the Persistent Homology of Point Clouds." In *Proceedings of Topological, Algebraic, and Geometric Learning Workshops 2022*, 96–106. PMLR.

Syzdykbayev, M., and H. A. Karimi. 2020. "Persistent Homology for Detection of Objects from Mobile LiDar Point Cloud Data in Autonomous Vehicles." In *Advances in Computer Vision*, edited by K. Arai and S. Kapoor, 458–472. Advances in Intelligent Systems and Computing. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-17798-0_37.

Turkes, R., G. F. Montufar, and N. Otter. 2022. "On the Effectiveness of Persistent Homology." *Advances in Neural Information Processing Systems* 35:35432–35448, December. New Orleans, LA.

Wang, J., and J. Shan. 2009. "Segmentation of Lidar Point Clouds for Building Extraction." In *American Society for Photogramm. Remote Sens. Annual Conference*, 1–13. Baltimore, MD.

Wasserman, L. 2018. "Topological Data Analysis." *Annual Review of Statistics and Its Application* 5 (1): 501–532. https://doi.org/10.1146/annurev-statistics-031017-100045.

Wong, C.-C., and C.-M. Vong. 2021. "Persistent Homology Based Graph Convolution Network for Fine-Grained 3D Shape Segmentation." In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 7078–7087. Montreal, QC: IEEE. https://doi.org/10.1109/ICCV48922.2021.00701.

Zhou, Q.-Y., and U. Neumann. 2009. "A Streaming Framework for Seamless Building Reconstruction from Large-Scale Aerial LiDar Data." In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2759–2766. https://doi.org/10.1109/CVPR.2009.5206760.

Zhou, Y., M. Verkou, M. Zeman, H. Ziar, and O. Isabella. 2022. "A Comprehensive Workflow for High Resolution 3D Solar Photovoltaic Potential Mapping in Dense Urban Environment: A Case Study on Campus of Delft University of Technology." *Solar RRL* 6 (5): 2100478. https://doi.org/10.1002/solr.202100478.