

Automatic Building Footprint Extraction and Regularisation from LIDAR Point Cloud Data

Mohammad Awrangjeb and Guojun Lu

School of Engineering and Information Technology

Federation University Australia

Gippsland Campus, Churchill Vic 3842

Email: {Mohammad.Awrangjeb,Guojun.Lu}@federation.edu.au

Abstract—This paper presents a segmentation of LIDAR point cloud data for automatic extraction of building footprint. Using the ground height information from a DEM (Digital Elevation Model), the non-ground points (mainly buildings and trees) are separated from the ground points. Points on walls are removed from the set of non-ground points. The remaining non-ground points are then divided into clusters based on height and local neighbourhood. Planar roof segments are extracted from each cluster of points following a region-growing technique. Planes are initialised using coplanar points as seed points and then grown using plane compatibility tests. Once all the planar segments are extracted, a rule-based procedure is applied to remove tree planes which are small in size and randomly oriented. The neighbouring planes are then merged to obtain individual building boundaries, which are regularised based on a new feature-based technique. Corners and line-segments are extracted from each boundary and adjusted using the assumption that each short building side is parallel or perpendicular to one or more neighbouring long building sides. Experimental results on five Australian data sets show that the proposed method offers higher correctness rate in building footprint extraction than a state-of-the-art method.

I. INTRODUCTION

Automatic building footprint extraction from remote sensing data is a prerequisite for many GIS (Geographic Information System) applications, such as urban planning and disaster management. Building footprint is a digital form of the actual building boundary. It is extracted and regularised using the remotely sensed data such as aerial imagery and LIDAR (Light Detection And Ranging) point cloud data.

Based on the usage of the input data, there are three main categories of building extraction methods [1]. The first category of methods fully relies on high resolution aerial imagery and are generally not reliable enough for practical use. The second category of methods employs LIDAR data, and offers an improved level of automation when compared to image-based methods alone. Methods in the third category integrate aerial imagery and LIDAR data in order to exploit the complementary information from both data sources [3]. The study reported in this paper utilises LIDAR data only for building footprint extraction.

The application of a recently proposed LIDAR-based method [1] to complex urban scenes with hilly terrain and dense vegetation has shown that this method is unable to remove vegetation completely. Thus, the experimental results from this method [1] have shown low correctness rate for the extracted building boundaries.

This paper, which is an extension of [2], proposes a building footprint extraction method using a novel segmentation technique for LIDAR point cloud data. The point cloud data is first divided into ground and non-ground points using a height threshold above the ground height. The non-ground points are mainly buildings and trees and these are further processed to remove the trees and to separate individual buildings. Points on walls can be removed by applying a plane fitting technique and by projecting data onto a dense grid. The remaining non-ground points are then divided into clusters based on height and local neighbourhood. A rule-based procedure is then applied to remove tree planes, which are small in size and randomly oriented. The neighbouring planes are merged to obtain individual building boundaries, which are regularised by adjusting small lines with respect to long lines.

Compared to Awrangjeb et al. [2], this paper has the following contributions. Firstly, it details the different steps of the proposed LIDAR data segmentation technique. Secondly, it proposes a technique for building outline regularisation. Finally, it shows results on five Australian data sets with varying point density, terrain and vegetation. The experimental results from the proposed method have shown that it is capable of removing more vegetation in complex scenes and thus, has offered higher correctness rate than Awrangjeb and Fraser [1]. Moreover, while the existing method [1] produced irregular building boundaries, the proposed method proposes a new technique for regularisation of the extracted boundaries.

II. RELATED WORK

There are numerous methods for segmentation of LIDAR point cloud or its raster version for building extraction. Jochem et al. [4] proposed a roof segmentation technique from raster LIDAR data using a seed point based region growing technique. Oude Elberink and Vosselman [5] proposed a target based graph matching approach that can handle both complete and incomplete laser data. Sohn et al. [6] proposed a new algorithm to generalize noisy polylines comprising a rooftop model by maximizing a shape regularity (orthogonality, symmetry and directional simplications). Mongus et al. (2014) used a multi-scale decomposition of LIDAR point cloud data to separate ground points and they applied a local surface fitting technique for extraction of planar segments. Awrangjeb and Fraser [1] proposed a region-growing method based on a plane-fitting technique. A similar approach was proposed in [7]. A more comprehensive review of building extraction methods can be found in Awrangjeb and Fraser [1].

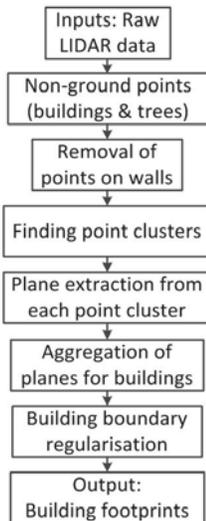


Fig. 1. The proposed building footprint extraction method.

Once the building boundary is obtained from the segmentation of LIDAR data, it has to be regularised to generate building footprint for practical applications. Among the regularisation methods, traditionally the Douglas-Peucker (DP) algorithm [8] has been extensively used for approximation of the extracted building contour with a number of straight lines [9]. However, the extracted lines are visually unattractive as they are not parallel or perpendicular to each other for original rectilinear building boundaries. In addition, the DP algorithm may remove critical points depending on the height threshold [7]. Weidner and Förstner [10] applied an optimization algorithm based on the local minimum description length (MDL) to the extracted DP lines to generate ten different orthogonal models. However, this method was sensitive to the initial point selection because of its local MDL approach and generated over-simplified footprints [9].

Jwa et al. [9] proposed a global optimization approach, where the optimum solution was achieved when a building footprint was maximally hypothesised as the repetition of identical line slope. Ameri [11] proposed a feature-based model verification method that applied the total least-squared adjustment based on linearity, connectivity, coplanarity and orthogonally. Although this method is flexible for maintaining linearity, connectivity, coplanarity and orthogonally, its performance is subject to the results from the DP algorithm. Moreover, it was found ineffective for high noisy boundary [11].

III. PROPOSED METHOD

Fig. 1 shows the work flow of the proposed building extraction technique. The proposed method first divides the input LIDAR point cloud data into ground and non-ground points. The non-ground points, representing objects above the ground such as buildings and trees, are further processed for building extraction. Points on walls are removed from the non-ground points, which are then divided into clusters. Planar roof segments are extracted from each cluster of points using a region-growing technique. Planar segments constructed in trees are eliminated using information such as area, orientation

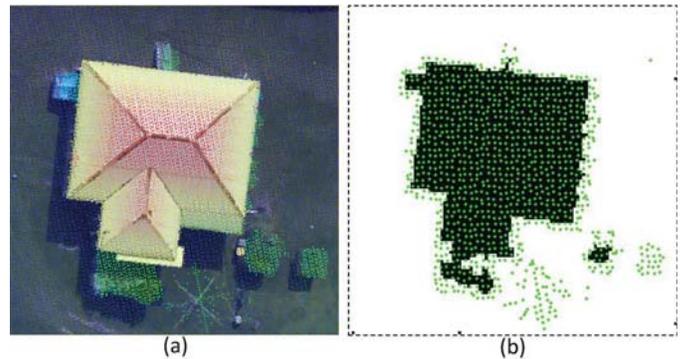


Fig. 2. A sample scene from the Aitkenvale data set: (a) LIDAR points overlaid on the orthoimage and (b) non-ground points overlaid at a low density on the primary building mask.

and unused LIDAR points within the plane boundary. Points on the neighbouring planar segments are accumulated to form individual building regions. A new algorithm is proposed to regularise the building boundary. The final output from the proposed method will be individual building footprints.

Fig. 2(a) shows a sample scene from the Aitkenvale data set, which has a point density of 40 points/m². In the following subsections, different steps of the proposed method are described using this sample scene.

All the parameter values used by the proposed method are either dependent on the input point cloud data or are commonly used in the literature, except the vertical distance threshold (Section III-A). This threshold is based on a reasonable assumption that the slope of a roof plane with respect to the ground is at most 75°; while walls are at 90° with respect to the ground.

A. Finding non-ground points and removing wall points

The non-ground points can be easily separated from the ground points using a bare-earth DEM (Digital Elevation Model). For each LIDAR point, the corresponding DEM height is used as the ground height H_g . A height threshold $T_h = H_g + h_c$, where h_c is a height constant that separates the non-ground points containing buildings and trees from the ground points comprising low height objects such as grass and road furniture, is then applied to the LIDAR data. In this study, $h_c = 1$ m has been set according to [3].

The wall points are then removed from the non-ground points by applying a plane fitting technique and by projecting data on a dense grid as follows. A fictitious horizontal plane \mathfrak{S} is formed. Considering the length and width of the test area to be l_a (Easting) and w_a (Northing), the following four points can be used to construct \mathfrak{S} : $(1, 1, 1)$, $(l_a, 1, 1)$, $(1, w_a, 1)$ and $(l_a, w_a, 1)$, where a constant height of 1 m is used. For each non-ground LIDAR point P , its neighbouring point set $S_n = \{Q_i\}$ is obtained, where each neighbour Q_i resides within a distance H_d from P and has a maximum height difference of V_d with P . The value of the horizontal distance threshold H_d is set at twice the maximum point-to-point distance (d_{max}) in the input LIDAR data [3]. Considering the maximum slope (with respect to \mathfrak{S}) for a roof plane is 75°, the value of the vertical distance threshold V_d is set at

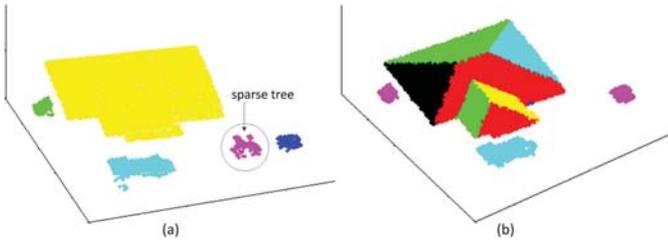


Fig. 3. (a) Clusters of points for the sample scene in Fig. 2 and (b) extracted planar segments.

$d_{max} \times \tan(75^\circ)$. S_n includes P and if $|S_n| \geq 3$, a plane ζ_p is constructed for P . If the angle between ζ_p and \mathfrak{S} is at most $\frac{\pi}{32}$, then ζ_p is determined as a vertical plane and P is removed as a wall point.

The above plane fitting procedure removes a large number of points reflected from walls. In order to remove the remaining wall points, all the surviving points are projected onto a dense grid of resolution 0.25 m [12]. Assuming that the grid resolution is less than the LIDAR point density (i.e., $d_{max} > 0.25$ m), this projection will transfer all points within a narrow area close to an imaginary vertical line to the same grid cell. If a grid cell has more than one projected point, points that are at the top and bottom of the line are kept and others are removed. Fig. 2(b) shows the non-ground points (at a low density) for the sample scene. An example with high rising buildings, where a large number of wall points were removed, was shown in Awrangjeb et al. [2].

B. Point clustering

All the non-ground points obtained after removal of wall points are now processed to generate clusters of points based on height. Initially, all these points are not assigned to any clusters. Considering the maximum LIDAR height as the current height, points at this height (within a specified tolerance) are found and one or more clusters are initialised depending on their locality. Then each of the cluster is extended until no points can be added to the cluster. Points in each cluster are marked so that they are not assigned to another cluster. Once all the clusters initialised from the current height are finalised, the points that are not yet assigned to any clusters are then processed to generate more clusters. The next current height will be the maximum height of these unassigned points.

Let the current maximum height be h_m and the set of points that has similar height (i.e., within $h_m \pm T_f$, where $T_f = 0.1$ m to allow the error in LIDAR generated heights [3]) is S_c . One or more clusters ξ_i , where $i \geq 1$, are initialised using S_c based on the locality. For a point $P \in \xi_i$, there is at least one neighbouring point Q in the same cluster such that the 2D Euclidean distance $|P.Q| \leq H_d$. If P and Q are in two different clusters then $|P.Q| > H_d$.

In order to extend a cluster ξ_i , let R be a neighbour of P , where $P \in \xi_i$ but R has been neither assigned to any cluster nor yet designated as a wall point. R is considered to be a neighbour of P , if $|P.R| \leq H_d$ and their height difference is within V_d . R is added to ξ_i , which is iteratively extended until no R is found as a neighbour of P .

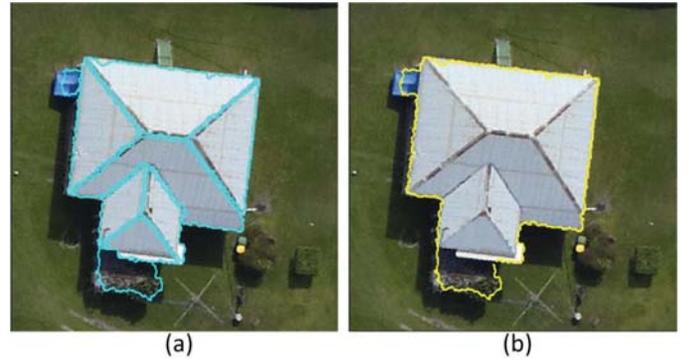


Fig. 4. (a) Extracted plane boundaries for the sample scene in Fig. 2 and (b) extracted building boundaries. Both are shown overlaid on the orthoimage.

Each of the extended clusters may represent an individual building, a roof plane or a roof section. Thereafter, an extended cluster ξ_i is considered a valid cluster if it is larger than 1 m² in area. To obtain the area of a cluster, its points are projected onto an initially white mask M_w of resolution 0.25 m [12]. For each point projected at location (x, y) of M_w , all the pixels within a $n \times n$ neighbourhood are made black. The value of n is determined based on d_{max} and the resolution of M_w such that all the pixels corresponding to an individual building, roof section or plane become black. The area of ξ_i is thus roughly estimated by counting the number of black pixels multiplied by the pixel size.

Fig. 3(a) shows a 3D view of all five clusters for the sample scene in Fig. 2.

C. Building extraction

Planar roof segments are extracted from each cluster of points. By using the Delaunay triangulation algorithm, a natural neighbourhood of points in the cluster can be generated. The neighbourhood of a point P consists of the points Q_i , $1 \leq i \leq n$, where each line PQ_i is a side of a Delaunay triangle. In order to avoid points which are far away from P , the following condition is applied: $|PQ_i| \leq T_d$, where $T_d = 2d_{max}$. The coplanarity of P is decided using its neighbouring points following the procedure in Sampath and Shan [13]. Points within a roof plane are found to be coplanar and those along the boundary of a plane are generally found to be non-coplanar [1].

For each cluster, let the two sets of the non-ground LIDAR points be S_1 , containing all the coplanar points, and S_2 , containing the rest (non-coplanar). The first planar segment can now be initialised using a coplanar point $P \in S_1$ and its neighbours. This new planar segment is extended using the neighbouring points from S_1 and S_2 . Once the extension is complete, all the coplanar points in the extended planar segment are marked so that none are latter used for initiating another planar segment. As a result, the points in S_2 , which mainly reside along the plane boundaries, can be used by more than one extracted plane. The second planar segment is grown by using an unused coplanar point (from S_1). The iterative procedure continues until no coplanar point remains unused [1].

Fig. 3(b) shows all the extracted planes for the sample

scene. There were no plane constructed on a sparse low height tree shown in Fig. 3(a).

In order to remove false-positive planes, mostly constructed on trees, a rule-based procedure is applied. For an extracted LIDAR plane, its area, straight line-segments along its boundary, and neighbourhood information, as well as any LIDAR spikes within its boundary, are used to decide whether it is a false alarm. For a given point on the extracted LIDAR plane, the mean height difference with its neighbouring points is also used. This height difference is large for a tree plane, but small for a roof plane. The average height difference for a plane is estimated from individual height differences within the plane. A LIDAR plane fitted on a tree is usually small in size and there may be some LIDAR spikes within its boundary. Moreover, there may be a large number of unused (i.e., not on any of the extracted planes) LIDAR points within the boundary of a tree plane. The number of points used by the extracted planes is usually low on a tree cluster, but high on a building cluster. Moreover, there may be some long straight line-segments (at least 3m long, the minimum building width W_{min} [14]) along the boundary of a roof plane.

In order to obtain the boundary of an extracted plane, the procedure described in Awrangjeb and Fraser [15] is followed. Fig. 4(a) shows the final extracted roof planes. All the LIDAR points from the neighbouring roof planes are now merged in order to obtain an individual building segment. The building boundary is thus extracted following the same procedure [15]. Fig. 4(b) shows the building boundary for the sample scene.

D. Building boundary regularisation

As can be seen in Fig. 4(b), there is a significant irregularity or noise in the LIDAR derived building boundary. Thus, the first step of the proposed regularisation method is to smooth the extracted boundary and to find the corners in order to decompose the boundary into line-segments. It has been observed that neither the contour-based corner detector [16] nor the DP algorithm [8] offers an acceptable set of line-segments alone. While the corner detector misses round corners, the DP algorithm generates a large number of small line-segments. Consequently, both of the algorithms are employed in the proposed regularisation method. The line-segments are then adjusted assuming their orthogonal relation (parallel or perpendicular) with the neighbouring long line-segments.

For smoothing there are both direct and indirect smoothing techniques [17]. While the direct smoothing, e.g., the Gaussian smoothing, physically changes the point locations by applying a convolution function, the indirect smoothing, e.g., chord-length, does not change the point locations but overlooks the details (i.e., small changes) along the boundary during estimation of curvature (corner strength).

A Gaussian smoothing with scale $\sigma = 3$ [16] is first applied to the extracted boundary. Then corners (point locations where considerable change in curve direction happens, i.e., curvature peaks) along the smoothed boundary are obtained by applying an indirect smoothing-based corner detector [16]. Fig. 5(a) shows the smoothed boundary in magenta colour and detected corners in yellow dots within cyan circles.

Assuming that a building side is at least 1 m long, small curve-segments between consecutive corners are not

considered at this point. A straight line is then fit to each of the remaining curve-segments using a least-square technique. Since the least-squares solution of straight line fitting does not always offer the appropriate direction of the estimated line, the direction of each line is adjusted following an iterative procedure to minimize the mean perpendicular distance d_{mean} of points to the line. Starting with an arbitrary angle, say, $\theta = 10^\circ$, the line is rotated (with respect to its centre) and d_{mean} is estimated. If the current d_{mean} is smaller than the previous d_{mean} , the direction of rotation (clockwise or anti-clockwise) is accepted; otherwise, the line is rotated to the opposite direction at the same angle. If the a smaller d_{mean} is still not found, θ is halved at each iteration until $\theta \leq 1^\circ$. If a smaller d_{mean} is never found, the original direction of line is considered correct. If a smaller d_{mean} is found, the line is kept rotating at the same angle to the same direction of rotation so long as a smaller d_{mean} is found. Consequently, the total angle of rotation δ is increased. If at some iteration, a smaller d_{mean} is not found the rotation angle is halved to test if a smaller d_{mean} is still can be found until $\theta \leq 1^\circ$. Once a minimum d_{mean} is found, the line is rotated by δ . If the minimum d_{mean} is lower than T_d the line is accepted as an appropriate line-segment along the building boundary. If d_{mean} is high, the line is not accepted. This can be due to missing of a corner point within the corresponding curve-segment. Fig. 5(b) shows all the accepted lines in cyan colour and an unaccepted line in red colour.

At this moment, the DP algorithm is applied to each unused (where no line has been fit yet) curve-segment, if any, between two accepted line-segments in order to find more line-segments. The perpendicular distance for the DP algorithm threshold is set at d_{max} . Fig. 5(c) shows the detected corners as black dots within red circles, where the unused curve-segment was decomposed. The line-segments, shown in yellow colour in Fig. 5(c), are fit and adjusted to these curve-segments as discussed above.

In order to obtain a regular polygonal boundary, small line-segments are now adjusted to their neighbouring long line-segments. Lines which are at least 6 m in length, twice the minimum building length [14], are considered long and kept fixed. This length threshold is based on the assumption that buildings mainly have two principal directions along its length and width. For a building having more than two principal directions, the third or later directions are considered only if the corresponding building-parts are at least 6 m long. For an extracted building boundary, if at least one long line-segment is not found the length-threshold is decreased by 1 m at each step to 3 m until at least one long line is found (minimum building width or length W_{min} is 3 m [14]).

The long lines are kept fixed. If there is only one fixed line, all other lines are adjusted (made parallel or perpendicular) to that line. Two nearest short lines on two sides of the fixed line are adjusted first. The next two nearest short lines are then adjusted with respect to the nearest already fixed line and the procedure continues until no short lines are left unadjusted.

If two or more fixed lines are found, each short line in between two long lines is made parallel or perpendicular to its nearest (in terms of number of lines) fixed line. If a short line is at the same distance from its two nearest fixed lines, then it

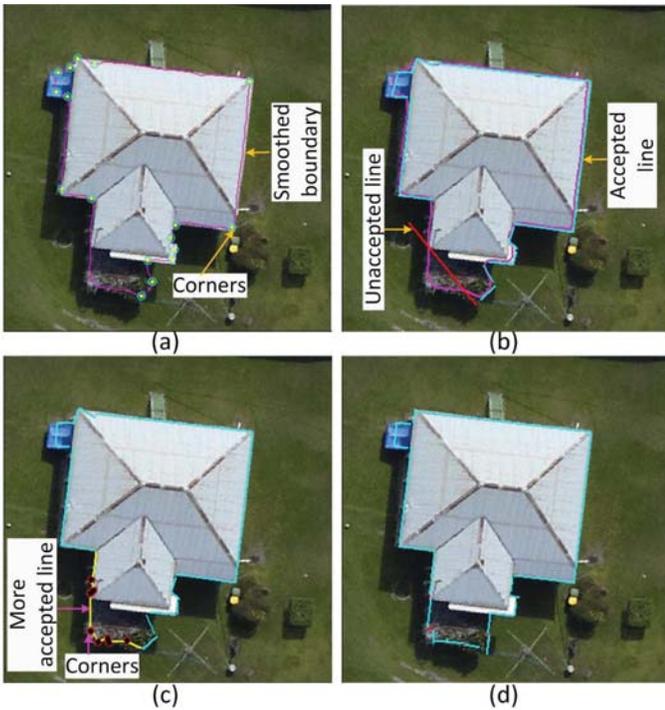


Fig. 5. Regularisation of a building boundary: (a) smoothed boundary and corners, (b) extracted lines and (c) more extracted lines and (d) adjusted lines.

is adjusted with respect to the fixed line with which it makes the smaller angle.

Perpendicular lines are then added in between successive parallel lines. Fig. 5(d) shows all adjusted lines in cyan colour and the inserted perpendicular lines in red colour. Finally, the intersection points of consecutive lines are obtained and the nearest LIDAR point heights are assigned to the intersection points to obtain the regularised 3D building boundary. Fig. 6(b) shows the regularised building footprints for a test area.

IV. PERFORMANCE STUDY

In the performance study conducted to assess the proposed approach, five Australian data sets from different geographic locations were employed. For all three data sets, bare-earth DEMs of 1 m horizontal resolution were available. The objective evaluation followed an automatic and threshold-free evaluation system [15].

In the evaluation system, three categories of evaluations (object-based, pixel-based and geometric) have been considered. A number of metrics are used in the evaluation of each category. While the object-based metrics (completeness, correctness, quality, detection and reference cross-lap rates, split and merge operations) estimate the performance by counting the number of buildings, the pixel-based metrics (completeness, correctness, quality, area omission and commission errors, branching and miss factors) show the accuracy of the extracted buildings by counting the number of pixels. In addition, the geometric metric (root mean square error, RMSE) indicates the accuracy of the extracted boundaries with respect to the reference entities. The definitions and how these metrics are estimated have been adopted from [14].

Once the reference and extracted building footprints are obtained, these metrics are automatically determined via the involved performance evaluation technique [15]. The minimum areas for small and large buildings have been set at 10 m² and 50 m², respectively. Small buildings are usually between 10 to 50 m² in size and mainly include garage and garden sheds. Large buildings are at least 50 m² in area and constitute main buildings such as houses and industries. In most test data sets, there were also buildings, mainly carports and garden sheds, which were less than 10 m² in area. Thus, the object-based completeness, correctness and quality values will be separately shown for all, small and large buildings.

In this section, the performance exhibited by the proposed method will be presented and discussed first and then a comparative performance analysis with the state-of-the-art method [1] will be presented.

A. Data sets

The first data set Aitkenvale (AV) has a high point density of 40 points/m² and the second data set Hervey Bay (HB) has a medium point density (12 points/m²) [3]. The AV scene covers an area of 66 m × 52 m and contains 5 buildings. The HB data set covers 108 m × 104 m and contains 28 buildings (3 were between 4 to 5 m² and 6 were between 5 to 10 m²). These two sites contain mostly residential buildings and they can be characterized as urban with medium housing density and moderate tree coverage that partially covers buildings. In terms of topography, AV is flat while HB is moderately hilly.

The other three data sets, Eltham (EL), Hobart (HT) and Knox (KN), have point densities of five, two and one points/m², respectively. These three test areas cover 393 m × 224 m, 303 m × 302 m and 205 m × 204 m, respectively. The EL data set contains 75 buildings (nine were less than 10 m², including five within three to 5 m²). The HT data set has 69 buildings (13 were less than 10 m², including four within one to 5 m²) and the KN data set contains 52 buildings (eight were less than 10 m², including four within two to 5 m²). These three data sets have dense vegetation and are in hilly areas. Many of the buildings are severely occluded by the surrounding trees. Moreover, in the KN data set, some parts of the building are at a similar height to the surrounding sloping grounds. Consequently, these building parts could not be extracted.

For all the data sets, 2D reference data sets were created by monoscopic image measurement using the Barista software [18]. All visible buildings were digitized as polygons irrespective of their size. The reference data included garden sheds, garages, etc. These were sometimes as small as less than 5 m² in area.

B. Results and Discussions

Tables I and II show the object- and pixel-based evaluation results, respectively, for all five test data sets. Figs. 6 to 8 show the detection results for the data sets.

For all buildings, in terms of both object- and pixel-based and geometric evaluation indices, the proposed method worked the best in the AV data set. There are two main reasons: the point density in this data is 40 points/m² and the buildings

TABLE I. OBJECT-BASED EVALUATION RESULTS FOR THE AITKENVALE (AV), HERVEY BAY (HB), ELTHAM (EL), HOBART (HT) AND KNOX (KN) DATA SETS IN PERCENTAGE. C_m = COMPLETENESS, C_r = CORRECTNESS AND Q_l = QUALITY ($C_{m,10}$, $C_{r,10}$, $Q_{l,10}$ AND $C_{m,50}$, $C_{r,50}$, $Q_{l,50}$ ARE FOR BUILDINGS OVER 10 M² AND 50 M² RESPECTIVELY); C_{rd} = DETECTION CROSS-LAP (UNDER-SEGMENTATION) AND C_{rr} = REFERENCE CROSS-LAP (OVER-SEGMENTATION) RATES; N_m AND N_s ARE NUMBERS OF MERGE AND SPLIT OPERATIONS DURING EVALUATION.

Areas	C_m	C_r	Q_l	$C_{m,10}$	$C_{r,10}$	$Q_{l,10}$	$C_{m,50}$	$C_{r,50}$	$Q_{l,50}$	N_m	N_s	C_{rd}	C_{rr}
AV	100	100	100	100	100	100	100	100	100	0	0	0	0
HB	86.4	100	86.4	95	100	95	100	100	100	1	0	0	4.6
EL	67.4	96.9	66	79.5	96.9	77.5	100	96.9	96.9	1	19	25.6	1.3
HT	68.4	92.9	65	83	92.9	78	97.5	92.9	90.7	4	1	2.1	6.8
KN	53.2	96.2	52.1	62.5	96.2	61	96.2	96.2	96.2	1	3	10.3	1.9
Average	75.1	97.2	73.9	84	97.2	82.3	98.7	97.2	96.8	1.4	4.6	7.6	2.9

TABLE II. PIXEL-BASED AND GEOMETRIC EVALUATION RESULTS FOR THE AITKENVALE (AV), HERVEY BAY (HB), ELTHAM (EL), HOBART (HT) AND KNOX (KN) DATA SETS. C_{mp} = COMPLETENESS, C_{rp} = CORRECTNESS, Q_{lp} = QUALITY, A_{oe} = AREA OMISSION ERROR, A_{ce} = AREA COMMISSION ERROR, B_f = BRANCHING FACTOR AND M_f = MISS FACTOR IN PERCENTAGE; $RMSE$ = PLANIMETRIC ACCURACY IN METRES.

Areas	C_{mp}	C_{rp}	Q_{lp}	A_{oe}	A_{ce}	B_f	M_f	$RMSE$
AV	95.1	97.1	92.4	4.9	3.0	3.1	5.2	0.57
HB	91.7	95.6	87.9	8.4	4.4	4.6	9.1	1.42
EL	83.5	87.4	74.5	17	13.2	14.4	19.8	3.02
HT	70.1	91.1	65.6	29.9	8.9	9.8	42.7	2.50
KN	59	91.7	56	42.7	8.6	9.1	69.4	2.82
Average	79.9	92.6	75.3	20.6	7.6	8.2	29.2	2.1

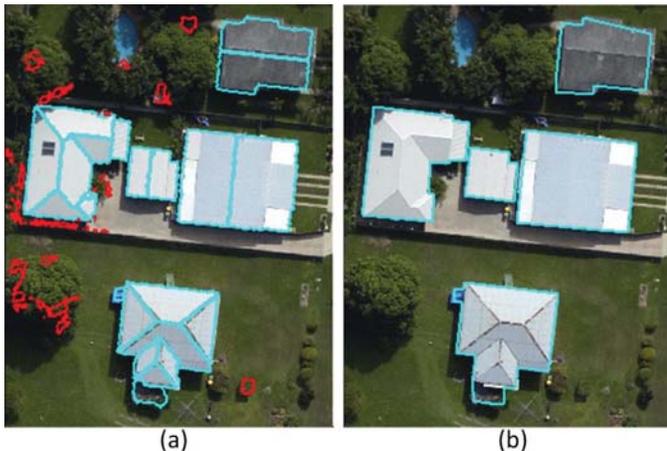


Fig. 6. Building extraction for the Aitkenvale data set: (a) all the extracted roof planar segments (red = removed as trees and cyan = accepted as true building planes) and (b) building footprints overlaid on the orthoimage.

here are big and well separated from each other. There are only 3 to 5% omission and commission errors and branching and miss factors.

The HB data set is the second where the proposed algorithm performed the best. Although the object-based correctness was 100%, which indicates that the method did not detect any false buildings, the completeness and quality values were lower than the maximum. The main reason is that due to low point density the method could not extract small garden sheds as shown in Fig. 7a. However, the method was able to extract all large buildings 50 m² in area.

The same trend of missing small buildings has been observed in other data sets. Fig. 7 show some examples of missing small buildings from the KN and HT data sets. Some of the missing buildings were also occluded by the surrounding trees as shown in a magnified snap-shot of the Knox data set in Fig. 7(c). Consequently, in these data sets the

completeness values are higher than the correctness values in both object- and pixel-based evaluation.

In the EL data set, a large number of buildings were found merged by the proposed method resulting in the highest detection cross-lap rate for the EL data set. Thus, during evaluation a large number of split operations were required. This phenomenon is illustrated by a number of snap-shots in Fig. 8. In fact, in the EL data set, buildings are close to each other and connected by the surrounding dense vegetation. The opposite scenario was observed in the HT data set, where due to low point density some of the building-parts on the same buildings were detected separately and the evaluation system required merge operations to make appropriate correspondences with the reference buildings. This has significantly contributed to the highest reference cross-lap rate for the HT data set among all the test data sets.

Compared to the HT data set, the EL and KN areas are more hilly and have more occluded buildings. However, the proposed algorithm performed better in the EL data set than in the HT and KN data sets. This is because the point density in the EL data set is higher than that in the HT and KN data sets. In all three, most of the large trees were removed, except in the HT data set where some trees were detected as shown in a red circle in Fig. 7b. As a result, the correctness values are more than 95% for the EL and HT data sets, but lower than 95% for the HT data set, even for buildings larger than 50 m² in area. Some of the detected trees in the HT data sets were very dense, so laser points hardly penetrated the canopy. Some of the detected tree tops in the HT data set were not only very dense, but also shaped such that they appeared to be flat planes. The extracted false buildings on these trees were thus too large to be removed.

For buildings larger than 10 m², the method did not do well for the EL, HT and KN data sets. Firstly, these three data sets had low point density and secondly, many of the garden sheds were occluded by the dense vegetation in the backyards. However, for buildings larger than 50 m², the performance

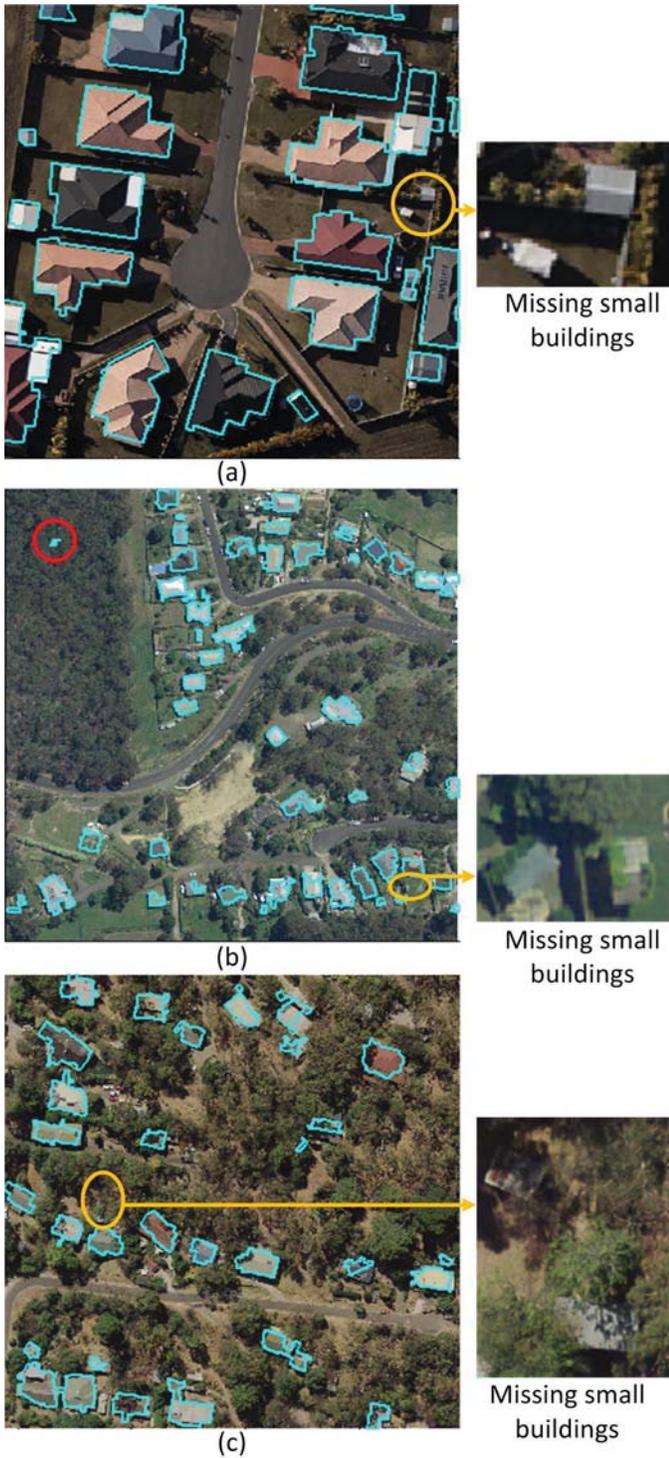


Fig. 7. Building footprints overlaid on the orthoimage: (a) Hervey Bay, (b) Hobart and (c) Knox. Red circle in (b) shows a false detection (tree). A magnified snap-shot for each scene illustrates the miss detections of small garage and garden sheds.

was more than 90% for all data sets in terms of correctness, completeness and quality. The largest jump was observed in the KN data set (61% quality for buildings larger than 10 m^2 to 96% quality for buildings larger than 50 m^2). This ensures that a large number of buildings in the KN data set were small

garage and garden sheds, which sometimes were occluded by trees as shown in Fig. 7(c).

The trend of the object-based results has been confirmed by the pixel-based results in Table II (for all buildings). The correctness values were higher than the completeness values for all data sets. The highest omission error was observed for the KN data set (43%), followed by the HT (30%) and EL (17%) data sets, due to a large number of small buildings being missed, which has also increased the miss factors for these data sets (69%, 43% and 20%, respectively). The highest area commission error and branching factor, but higher correctness than completeness, for the EL data set entails that many extracted buildings also partly included the neighbouring and occluded trees as shown within red circles in the snap-shots of Fig. 8.

The inclusion of nearby trees with the actual extracted boundary in the EL data set also increased the geometric inaccuracy (RMSE). For other data sets, the RMSE was equivalent to 2 to 3 times of the maximum point-to-point distance d_{max} in the input point cloud data.

C. Comparative results

Since the proposed algorithm is an automatic method and works solely with LIDAR data, one of the existing methods proposed by Awrangjeb and Fraser [1] that used the same data sets has been considered for comparison.

Compared to the existing method [1], the proposed method offered better correctness values in object-based evaluation. This performance difference was observed to be significant for the KN (75% for [1] vs 96% for the proposed method), HT (81% vs 93%) and EL (88% vs 97%) data sets. In pixel-based evaluation, it also showed better correctness in the KN (58% vs 92%) and HT (80% vs 91%) data sets, but slightly lower in the EL data set (90% vs 87%). This is a strong indication that the proposed method successfully removed almost all the vegetation in these complex scenes, but Awrangjeb and Fraser [1] detected some trees as buildings. However, Awrangjeb and Fraser offered better completeness in these data sets.

The same performance trend has been observed for buildings larger than 10 m^2 and 50 m^2 in area. Awrangjeb and Fraser showed higher detection cross-lap rates in the HT and KN data sets, while the proposed method offered higher reference cross-lap rates in the HB and HT data sets.

V. CONCLUSION

A method for segmentation of LIDAR point cloud data for automatic extraction of building footprints has been proposed. Experimental results have shown that the proposed method is capable of removing trees and thus yielding a higher correctness rate in building footprint extraction than a recently published article [1], especially in complex urban scenes, as exemplified here by Knox, Hobart and Eltham data sets.

However, since the method uses LIDAR data alone, the high RMSE error indicates that the planimetric accuracy is limited by the LIDAR point density. Moreover, the proposed method will not work on curved roofs. Future work aims to look at the development of a new LIDAR segmentation technique for the curved roof and to reconstruct building roof



Fig. 8. Building footprints overlaid on the orthoimage for the Eltham data set. A number of magnified snap-shots illustrate the merged detections of neighbouring buildings that require split operations during evaluation.

models. The integration of image data will also facilitate better object extraction where LIDAR information is missing.

ACKNOWLEDGMENT

Dr. Awrangjeb is the recipient of the Discovery Early Career Researcher Award by the Australian Research Council (project number DE120101778). The Aitkenvale and Hervey Bay data sets were provided by Ergon Energy (<http://www.ergon.com.au>) in Queensland, Australia. The Hobart data set was provided by Photomapping Services (<http://www.photomapping.com.au/>) in Melbourne, Australia. The Knox and Eltham data sets were provided by the Department of Environment and Primary Industries (DEPI, <http://www.depi.vic.gov.au/>) of Victoria, Australia.

REFERENCES

- [1] M. Awrangjeb and C. S. Fraser, "Automatic segmentation of raw LIDAR data for extraction of building roofs," *Remote Sensing*, vol. 6, no. 5, pp. 3716–3751, 2014.
- [2] M. Awrangjeb, G. Lu, and C. S. Fraser, "Automatic building extraction from LIDAR data covering complex urban scenes," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL, no. 3, pp. 25–32, 2014.
- [3] M. Awrangjeb, C. Zhang, and C. S. Fraser, "Automatic extraction of building roofs using LIDAR data and multispectral imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 83, no. 9, pp. 1–18, 2013.
- [4] A. Jochem, B. Höfle, V. Wichmann, M. Rutzinger, and A. Zipf, "Area-wide roof plane segmentation in airborne LIDAR point clouds," *Computers, Environment and Urban Systems*, vol. 36, no. 1, pp. 54–64, 2012.
- [5] S. Oude Elberink and G. Vosselman, "Building reconstruction by target based graph matching on incomplete laser data: analysis and limitations," *Geodetski Vestnik*, vol. 9, no. 8, pp. 6101–6118, 2009.
- [6] G. Sohn, Y. Jwa, J. Jung, and H. B. Kim, "An implicit regularization for 3D building rooftop modelling using airborne LIDAR data," in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. I-3, Melbourne, Australia, 2012, pp. 305–310.
- [7] K. Zhang, J. Yan, and S.-C. Chen, "Automatic construction of building footprints from airborne LIDAR data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 9, pp. 2523–2533, 2006.
- [8] D. Douglas and T. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *The Canadian Cartographer*, vol. 10, no. 2, pp. 112–122, 1973.
- [9] Y. Jwa, G. Sohn, V. Tao, and W. Cho, "An implicit geometric regularization of 3D building shape using airborne LIDAR data," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, no. 5, pp. 69–76, 2008.
- [10] U. Weidner and W. Förstner, "Towards automatic building extraction from high-resolution digital elevation models," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 50, no. 4, pp. 38–49, 1995.
- [11] B. Ameri, "Feature based model verification (FBMV): A new concept for hypothesis validation in building reconstruction," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 19, no. 3, pp. 24–35, 2000.
- [12] M. Awrangjeb, C. Zhang, and C. S. Fraser, "Building detection in complex scenes thorough effective separation of buildings from trees," *Photogrammetric Engineering & Remote Sensing*, vol. 78, no. 7, pp. 729–745, 2012.
- [13] A. Sampath and J. Shan, "Segmentation and reconstruction of polyhedral building roofs from aerial LIDAR point clouds," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 3, pp. 1554–1567, 2010.
- [14] M. Awrangjeb, M. Ravanbakhsh, and C. S. Fraser, "Automatic detection of residential buildings using LIDAR data and multispectral imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 65, no. 5, pp. 457–467, 2010.
- [15] M. Awrangjeb and C. S. Fraser, "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 and Remote Sensing*, vol. doi: 10.1109/JSTARS.2014.2318694, 2014.
- [16] M. Awrangjeb, G. Lu, C. S. Fraser, and M. Ravanbakhsh, "A fast corner detector based on the chord-to-point distance accumulation technique," in *Proc. Digital Image Computing: Techniques and Applications*, Melbourne, Australia, 2009, pp. 519–525.
- [17] M. Awrangjeb, G. Lu, and C. S. Fraser, "Performance comparisons of contour-based detectors," *IEEE Transactions on Image Processing*, vol. 21, no. 9, pp. 4167–4179, 2012.
- [18] Barista. (2011) The barista software. [Online]. Available: www.baristasoftware.com.au. Accepted on 21 Jul 2014.