
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Lehtola, Ville V.; Lehtomaki, Matti; Hyyti, Heikki; Kaijaluoto, Risto; Kukko, Antero; Kaartinen, Harri; Hyyppa, Juha

Preregistration classification of mobile LIDAR data using spatial correlations

Published in:
IEEE Transactions on Geoscience and Remote Sensing

DOI:
[10.1109/TGRS.2019.2909351](https://doi.org/10.1109/TGRS.2019.2909351)

Published: 01/09/2019

Document Version
Publisher's PDF, also known as Version of record

Published under the following license:
CC BY

Please cite the original version:
Lehtola, V. V., Lehtomaki, M., Hyyti, H., Kaijaluoto, R., Kukko, A., Kaartinen, H., & Hyyppa, J. (2019). Preregistration classification of mobile LIDAR data using spatial correlations. *IEEE Transactions on Geoscience and Remote Sensing*, 57(9), 6900-6915. Article 8700597. <https://doi.org/10.1109/TGRS.2019.2909351>

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

Preregistration Classification of Mobile LIDAR Data Using Spatial Correlations

Ville V. Lehtola¹, Matti Lehtomäki², Heikki Hyyti, Risto Kaijaluoto, Antero Kukko, Harri Kaartinen, and Juha Hyypä

Abstract— We explore a novel paradigm for light detection and ranging (LIDAR) point classification in mobile laser scanning (MLS). In contrast to the traditional scheme of performing classification for a 3-D point cloud after registration, our algorithm operates on the raw data stream classifying the points on-the-fly before registration. Hence, we call it preregistration classification (PRC). Specifically, this technique is based on spatial correlations, i.e., local range measurements supporting each other. The proposed method is general since exact scanner pose information is not required, nor is any radiometric calibration needed. Also, we show that the method can be applied in different environments by adjusting two control parameters, without the results being overly sensitive to this adjustment. As results, we present classification of points from an urban environment where noise, ground, buildings, and vegetation are distinguished from each other, and points from the forest where tree stems and ground are classified from the other points. As computations are efficient and done with a minimal cache, the proposed methods enable new on-chip deployable algorithmic solutions. Broader benefits from the spatial correlations and the computational efficiency of the PRC scheme are likely to be gained in several online and offline applications. These range from single robotic platform operations including simultaneous localization and mapping (SLAM) algorithms to wall-clock time savings in geoinformation industry. Finally, PRC is especially attractive for continuous-beam and solid-state LIDARs that are prone to output noisy data.

Index Terms— Classification algorithms, laser radar, machine vision, remote sensing, simultaneous localization and mapping.

I. INTRODUCTION

CLASSIFICATION is a process of categorizing data and is required for many purposes. For light detection and ranging (LIDAR) data captured during mobile laser scanning (MLS), the process can be thought to be roughly dividable into three *modi operandi*. The first and the common way is to perform postprocessing classification for a whole 3-D registered point cloud that may or may not be georeferenced (see [1], [2]). The second way, used typically with simultaneous localization and mapping (SLAM), is to perform real-time classification on a set of mutually aligned points, benefiting from 3-D *relative* registration [3] or data from a 3-D scanner [4]. While the first way, classification after registration (CAR), is likely to offer the best results with respect to classification metrics, the second, mutually aligned point subsets (MAPSs), enables online solutions. Hence, the time preference required for the task, i.e., be it during the registration, or after it, determines what is the most suitable classification approach for the job.

By taking the time preference into the short limit, there is also a third way. This third way means that the raw range measurement data stream from an MLS system is classified before the registration process begins. We call this *preregistration classification* (PRC). The enabling idea to do this is to look at the spatial correlations existing between chronologically neighboring range measurements when the scanner data stream contains the range measurements in the exact chronological order they were emitted. The two major differences between the raw range measurement data stream from a moving 2-D scanner and a 3-D registered points cloud are as follows. First, the scanner outputs data with a local built-in 2-D geometry (range and scan angle) while the registered point cloud has 3-D geometry (from the additionally known scanner poses). Second, in contrast to the 3-D geometry, the 2-D geometry is non-continuous because scanner movements break the continuity of the data. Nevertheless, these are surmountable challenges.

Spatial correlations exist in raw LIDAR data and can be benefited from. Previously, benefits from these have been obtained in outlier filtering before SLAM to enable accurate positioning for noisy scanners [5], [6] and to improve SLAM efficiency [6], [7], and in anomalous photon filtering

Manuscript received October 19, 2018; revised February 4, 2019; accepted March 28, 2019. Date of publication April 26, 2019; date of current version August 27, 2019. This work was supported in part by the Academy of Finland grants 272195 “Centre of Excellence in Laser Scanning Research (CoE-LaSR)” and 293389 / 314312 “the Competence-Based Growth Through Integrated Disruptive Technologies of 3-D Digitalization, Robotics, Geospatial Information and Image Processing/Computing Point Cloud Ecosystem.” (Corresponding author: Ville V. Lehtola.)

V. V. Lehtola is with the National Land Survey of Finland, Finnish Geospatial Research Institute, 00521 Helsinki, Finland, and also with the ITC Faculty, EOS Department, University of Twente, 7514 AE Enschede, The Netherlands (e-mail: ville.lehtola@iki.fi).

M. Lehtomäki, R. Kaijaluoto, A. Kukko, and J. Hyypä are with the National Land Survey of Finland, Finnish Geospatial Research Institute, 00521 Helsinki, Finland.

H. Hyyti is with the National Land Survey of Finland, Finnish Geospatial Research Institute, 00521 Helsinki, Finland, and also with the Department of Electrical Engineering and Automation, Aalto University, 00076 Aalto, Finland.

H. Kaartinen is with the National Land Survey of Finland, Finnish Geospatial Research Institute, 00521 Helsinki, Finland, and also with the Department of Geography and Geology, University of Turku, 20500 Turku, Finland.

This article has supplementary downloadable material available at <http://ieeexplore.ieee.org>, provided by the author.

Color versions of one or more of the figures in this article are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TGRS.2019.2909351

for solid-state scanners to increase the scanner signal to noise ratio [8]. However, classification from spatial correlations has not yet been tried. This “third” classification paradigm is called for because of several reasons.

First, a real-time classification may be of great benefit in robust positioning of an MLS system. Kukko *et al.* [9] use graphSLAM to show that points labeled as tree trunk can be used to correct a global navigation satellite system (GNSS)-based trajectory degraded by multipathing effects. If tree trunk points could be classified prior to registration, i.e., with PRC, the whole SLAM process could be done without GNSS allowing for solutions for the most challenging areas with respect to multipathing, e.g., dense and mountainous forests. Moreover, robust positioning is needed in autonomous driving [10]. Distinguishing smooth surfaces (e.g., concrete) from nonsmooth surfaces (e.g., vegetation) reduces the signal ambiguity for the benefit of a more reliable decision making, e.g., in collision avoidance and collision damage control.

Second, a fast classification with PRC widens data transmission bottlenecks and saves data processing time. Scanners output more and more data while the raw scanner data contain a lot of useless measurements for most representations. Good examples of this are overly large point densities [11] and noise, which is familiar for continuous-beam LIDARs and will be an increasing issue with the upcoming generation of single-photon LIDARs that are vulnerable to background illumination [8], [12]. A PRC scheme may be employed to omit these. On the one hand, PRC is able to remove those points that have an abnormal range value in contrast to their neighborhood, i.e., noise and outliers that follow from reflections or background illumination [6]. On the other hand, PRC could be used to downsample, for example, ground points that are abundant in most mobile mapping scenarios, such as in urban environments [13] and in forest operations [9], [14]–[16]. This would save subsequent online computation, transmission, and postcomputation time, leading to wall-clock time savings on a larger industry scale. In the near future, autonomous vehicles ideally would update the street maps—and possible extraordinary events on them—via car-to-car and car-to-infrastructure communication. On this grand scale, however, data transmission rates are likely to pose bottlenecks. It is known that the use of conventional (image) compression techniques to enhance data transmission rates for point clouds leads to artifacts [17]. In contrast, by reducing the amount of transmitted data in an intelligent way, PRC could widen this bottleneck. While advances in SLAM allow a low drift online solution [18], an online classification algorithm such as PRC could bring added benefit in preprocessing or simultaneously processing the data for registration purposes.

Third, in addition to reducing unwanted information to enhance subsequent data processing, a PRC scheme founding on the spatial correlations is likely to contain information between adjacent measurements that an extrinsically registered point cloud does not contain, e.g., when registered with a global navigation satellite system like in [13] and in [19].

For the above-mentioned reasons, in this paper, we set out to explore to what extent can these spatial correlations that exist in the raw LIDAR data stream of a moving 2-D scanner

be exploited for segmentation and classification purposes. The benefits of the PRC scheme that we are trying to reap are twofold. PRC may be used for a 2-D scanner in real time to detect noise and other class types for enhancing SLAM and other purposes, e.g., by intelligently reducing the amount of transmitted data, while storing the spatial correlations that are otherwise omitted in the registration process into classification labels to increase the amount of information and its density in the data.

The idea of the proposed PRC method is as follows. We process the data from the mobile LIDAR in blocks of multiple adjacent scan lines, with each scan line consisting of the points captured while the scanner mirror revolves once.¹ For each block, the unregistered scan lines are grouped together to form a (geometrically distorted) range image, which is then treated with image processing methods to examine the spatial correlations. The size of the range image is chosen sufficiently large so that the classification of different surface types is feasible inside a single block. The block sizes are, however, kept small enough to allow for a real-time processing capability, meaning that the computation time is lower than the data capture time.

The proposed method significantly differs from the CAR and MAPS methods—also with respect to the requirements for the input data. The raw scanner data stream must not have been filtered in any way, and the range measurements must be in an emission-based chronological order. For traditionally obtained data, this is typically not the case as the scanner may have firmware filters or the data is otherwise postprocessed on the computer end. Hence, in this paper, we employ two raw data sets from which 3-D point clouds have been derived for CAR studies in urban and forest environments [13], [20]. This accomplishes the two goals that must be fulfilled simultaneously. On the one hand, we obtain the raw data that PRC requires, and on the other hand, we can scientifically compare our results against published CAR results and data.

The rest of this paper is organized as follows. First, the related work is reviewed. In Section II, our PRC method is introduced. Test sites and reference data taken from previous works are briefed in Section III. Control parameter calibration, results presentation, and their comparison against CAR method data from built environment and forest are given in Section IV. Section V holds the discussion, and Section VI concludes the paper.

A. Related Work

We wrote about two *modi operandi* for the classification of MLS data in the introduction. The first *modus*, CAR, is to run classification algorithms on 3-D point clouds. For urban point clouds, the focus is on detecting building facades [21], road surfaces and edges [22]–[24], and distinguishing road inventory, including lighting poles and traffic signs [13], [19]. Vosselman [1] studies several postprocessing methods, and Weinmann *et al.* [2] focus on optimal pointwise interpretation of urban 3-D point clouds. For the purposes of precision

¹At a frequency of dozens or even hundreds of Hertz.

forestry, single-tree inventory samples are obtained by classifying tree stems using mobile [25] and personal [20] scanning equipment. Samples are used to enhance large-scale data captured from aircraft, cf. [26]. In the forest, the detection of ground level has been attempted by studying the behavior of the point subset captured in an enlarging circle around the base of a static terrestrial laser scanner [27].

The second *modus*, MAPS, is to classify points belonging to a set of mutually aligned points, consisting of one or multiple adjacent scan lines. The role of ground detection is essential for these methods. Mutually aligned 3-D point subsets have been used to create $2\frac{1}{2}$ D occupancy grids of $40\text{ m} \times 40\text{ m}$ size in order to segment objects from the ground based on point altitude [3]. One strategy is to classify the lowest points in 3-D scanner snapshots as ground and subsequently segment the separated nonground point subsets in an urban environment [4]. In forest, Hyyti and Visala [14] differentiate points obtained from an all-terrain vehicle platform to ground and tree classes, labeling ground first with a line fitting method discussed in the following. Ground detection is, however, not always a prerequisite to all classification, as also shall be the case for noise and vegetation for PRC. Lalonde *et al.* [15] perform an online 3-D registration incrementally so that offroad vehicles can detect salient object features, which are then labeled as cylindrical trees, ground, and soft vegetation. The ground data per se can be further classified: road classification for surface modeling [28] and lane detection [29] are relevant for, e.g., robotic vehicles. Also, geometrically salient features may be identified and extracted to enhance the computational efficiency of SLAM [18], [30].

In addition to pointwise classification [2], [15], the existing CAR and MAPS methods use line, plane, or cylinder fitting. Line fitting consists of multiple different techniques [30] and has been used in road edge detection [23] and forest ground classification [14]. In the work in [14], ground labeling is conducted by an iterative line fit that is used to find the longest straight subset within a scan profile while also identifying tree stem edges with a finite-difference derivative. Plane extraction methods have been used, for example, in indoor environments [31]. Tree stem classification is based on cylinder (object) detection for continuous registration [15] or cylinder fitting for whole point clouds [20], [25]. In contrast to these line-, plane-, or cylinder-based methods, PRC provides pointwise classification from local point neighborhoods while the scanner pose is unknown.

Instead of classifying points into named categories, they can be segmented into homogeneous regions so that points in the same region have similar properties [32]. To this end, Chauca *et al.* [33] employ image morphology and connected component techniques on snapshots taken from a whole 3-D registered point cloud. This line of thought can be taken toward real time, for example, if a depth camera (RGB-D) or a Velodyne scanner using 16 or more scan lines that are physically fixed to each other is employed. Then, a local 3-D geometry exists online for each point. Bogoslavskyi and Stachniss [4] use this 3-D local geometry to perform ground classification and the following nonground object segmentation with 3-D scanner data. Trevor *et al.* [34] used connected components on organized point clouds obtained from RGB-D data in

an indoor environment. They report plane segmentation and Euclidean clustering to be feasible in real time because the natural RGB-D ordering of the point cloud saves CPU time in an otherwise costly neighborhood search. Specifically, vegetation-free indoor RGB-D data are ideal for machine learning-based classification [35]. Trevor *et al.* [34] constructed binary images for connected components processing benefiting from the natural 3-D point ordering. We shall do the same but without any access to the 3-D local geometry. In other words, the requirement of using a 3-D scan array is relaxed in this paper. Furthermore, our test environments contain vegetation, which has complex topology.

In addition to geometry, also, LIDAR radiometry can be used for classification purposes. However, the modality of RGB camera data, i.e., 3 channels in 2 dimensions, is generally more favorable to texture characterization than the single intensity channel in 2-D LIDAR data. For example, intensity measurements of a laser scanner enable automatic terrain classification between paved and grass areas, with an accuracy of almost 100% [36]. Previously, this was done with an RGB camera [3]. Fitting a curve on the intensity values of a scan line can be used to determine road width and markings [28]. In addition to using intensity measurements of an active scanner, the road texture and markings may naturally be obtained with a passive sensor such as a line camera. If a line camera is mounted into the same multisensor array than a laser profiler, these textures are obtainable when the data are registered into 3-D coordinates [37]. However, in our paper, the main focus is on classifying the ground per se, instead of a detailed study of the different ground textures or surface materials. Also, our method is independent of radiometry.

II. PROPOSED PRC METHOD

The purpose of the proposed method is to exploit the spatial correlations that exist between the raw range measurements in a point segmentation and classification scheme. Let us first determine the extent of information that is available from the raw scanner data stream. Observations consist of range measurements (or points)

$$r_i \equiv r(l_i, h_i) \quad (1)$$

where r_i is the measured range, i is the (chronological) index number of the scan point p_i , and $h_i \in [1, M]$ is the index of the point within a scan line. All scan lines have a fixed length of M points so that indexing $i = l_i M + h_i$ holds, where l_i is the index of the 2-D scan line into which r_i belongs. Note that the 3-D coordinates of the points and the pose of the scanner are unknown and remain so during the PRC process.

Spatial correlations between the raw range measurements defined in (1) are recovered as follows. A block of adjacent scan lines from the scanner data stream is used as a range image (Section II-A) that is then converted into a binary image via the concept of support (Section II-B). The binary image consists of pixels representing either foreground, i.e., supported r_i , or background, i.e., noise. An extension to the binary image background is made (Section II-C) to provide robustness in the following segmentation (Section II-D). After segmentation, areas may be oversegmented. Hence, dilation

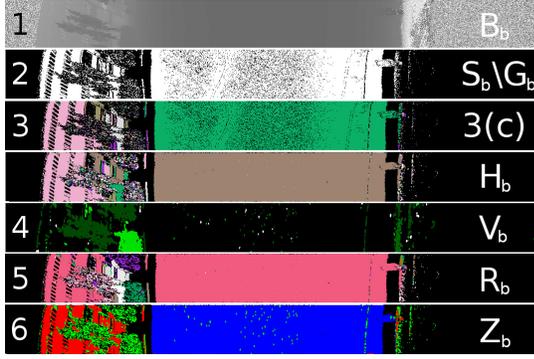


Fig. 1. Visualization of the Algorithm 1 steps from 1 to 6 (step 3 involves two images, Fig. 3(c) and H_b). Step 1 shows a block B_b of raw range data from the Espoonlahti urban test site, $k = 99$ scan lines. For the following steps, points representing noise $p_i \in F_b$ and the extended background $p_i \in G_b$ are shown in black. After connected components, see image in Fig. 3(c) and those below it, separate areas are shown with different colors. In H_b , nonsmooth surfaces are represented by numerous small segments which are gathered into larger entities in step V_b . The final segmentation in R_b is obtained as a joint outcome of H_b and V_b . The final image with label Z_b shows the classification into noise (black), ground (blue), buildings (red), and vegetation (green). A windowed building, a tree, some bushes, the road, and one car can be visually identified.

operations (Section II-E) are used to effectively target this oversegmentation. Finally, each segment is classified either as noise, as a smooth surface (representing, e.g., ground or buildings) or as a nonsmooth surface (representing, e.g., vegetation). The overall process is outlined in Algorithm 1 and visualized in Fig. 1.

A. From Range Image to Binary Image

The data stream consisting of range measurements r_i of (1) is buffered into blocks B_b of k scan lines, where $b = 1, 2, 3, \dots$. The block B_b can then be represented as an $k \times M$ (2-D) image with intensity values corresponding to range values, as shown on the top of Fig. 1. In other words, B_b is a range image. The image size should be chosen so that it is not too large for real-time processing but sufficiently large to enable the distinction of different surfaces. We choose $k = 99$ ($M = 2154$) for the urban case to show that even a quite limited cache may suffice for the method to function as this is of relevance for future distributed solutions and a larger $k = 255$ ($M = 4270$) for the forest case since the objects are closer to the scanner. Our algorithm is real-time capable (see Section IV-D).

The range image B_b is mapped to a binary image by simply dividing the range measurements of (1) into supported ones S_b , defined as

$$S_b = B_b \setminus F_b \quad (2)$$

and nonsupported ones F_b . The supported measurements r_i that belong in B_b , are marked as $r_i \in S_b$, and form the foreground (=ones) of the image. Similarly, the background (=zeros) is formed by F_b . A detailed explanation on the support S_b follows.

B. Supported Measurements

We define that a measurement r_i of (1) is supported by spatial correlations, $r_i \in S_b$, if the cover of supporting

Algorithm 1 Proposed PRC Method Takes the Stream of r_i of (1) as Input, and Yields a Label for Each r_i (Urban: $p_i \in \{\text{Noise, Ground, Vegetation, Buildings}\}$, Forest: $p_i \in \{\text{Noise, Ground, Tree Trunk}\}$)

- 1) Form B_b from $k \times M$ first points of the r_i stream, $b \in 1, 2, 3, \dots$
- 2) Define supported points, $S_b = B_b \setminus F_b$ (see Section II-B).
- 3) Compute labeled image H_b .
 - a) Create a binary image of $k \times M$ pixels with S_b as ones and F_b as zeros.
 - b) Extend the image background (zeros) with high gradient points G_b (see Section II-C).
 - c) Run connected components (see Section II-D).
 - d) Return each point $p_i \in G_b$ to foreground, giving the most common label of neighborhood.
 - e) Re-index segmentation labels to continue labeling from R_{b-1} .
- 4) Compute labeled image V_b .
 - a) From H_b , initialize a binary image, where labeled areas larger than A are also defined as background.
 - b) Morphologically dilate the binary image d times (see Section II-E).
 - c) Run connected components.
- 5) Combine labels $R_b = R_b(H_b, V_b)$
- 6) Classify each $p_i \in F_b$ as noise. Set $p_i \in H_b$ as smooth surface, except if $p_i \in V_b$ set it as nonsmooth surface (for ground labeling, see Section II-F).
- 7) Goto 1 to process the next data block B_{b+1} , if more data is available.

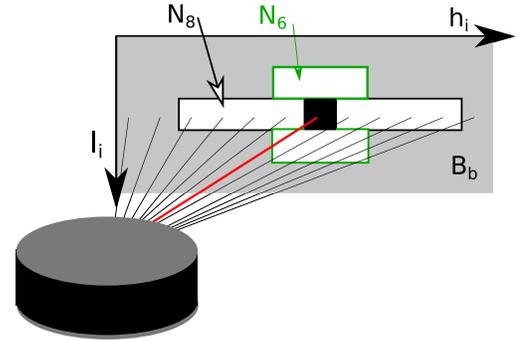


Fig. 2. Illustration of the mapping of a scan line from a 2-D laser scanner into a row of pixels of image B_b . As the time advances, new scan lines of length M are added increasing the index l_i . Note that only a part of B_b (in gray) is shown, as $M = 2154$ (urban) and $M = 4270$ (forest) in this paper. Spatial correlations are searched within the neighborhood $N(i)$ of the point i represented with a black rectangle. The neighborhood is shown in white and is separated into two areas encircled with black N_8 and green N_6 borders (see text for details).

measurements C in its neighborhood $N(i)$ satisfies the following condition:

$$C \geq \rho_c |N(i)| \quad (3)$$

where we choose $\rho_c = 1/3$ to define a minimum support cover density (and motivate this choice in the following). See Fig. 2 for illustration. We do not, however, want the supporting

points to form a line, so we split the neighborhood $N(i)$ of (3) into two. This is important as we want to examine surface smoothness for classification purposes later on. Hence, (3) is redefined as

$$r_i \in S_b \quad \text{if} \quad \begin{cases} C_{\text{in-slice},i} \geq \rho_c |N_{\text{in-slice}}| \text{ and} \\ C_{\text{adj-slice},i} \geq \rho_c |N_{\text{adj-slice}}|. \end{cases} \quad (4)$$

For (4), we then have both in-slice and adjacent-slice range difference conditions

$$C_{\text{adj-slice},i} = \sum_{j \in N_{\text{adj-slice}}(i)} \delta_{ij}, \quad \delta_{ij} = \begin{cases} 1, & \text{if } |r_i - r_j| < \zeta_s \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

and

$$C_{\text{in-slice},i} = \sum_{j \in N_{\text{in-slice}}(i)} \delta_{ij}, \quad \delta_{ij} = \begin{cases} 1, & \text{if } |r_i - r_j| < \zeta_p \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

where the two separate neighborhood areas are (see Fig. 2)

$$N_{\text{adj-slice}}(i) \equiv \{r_{i-M-1}, r_{i-M}, r_{i-M+1}, r_{i+M-1}, r_{i+M}, r_{i+M+1}\} \quad (7)$$

$$N_{\text{in-slice}}(i) \equiv \{r_{i-4}, r_{i-3}, r_{i-2}, r_{i-1}, r_{i+1}, r_{i+2}, r_{i+3}, r_{i+4}\}. \quad (8)$$

In words, $N_{\text{adj-slice}}(i)$ of (7) includes three range measurements around the position h_i from the previous and next scan lines,² and $N_{\text{in-slice}}(i)$ of (8) includes the eight closest range measurements of i within the same scan line. For the chosen $\rho_c = 1/3$ and the neighborhoods of (7) and (8), the support condition (4) employs two constants, $\rho_c |N_{\text{in-slice}}| = 2$ and $\rho_c |N_{\text{adj-slice}}| \approx 3$. This is a generalized formalization and improvement of our preliminary previous work that also accounts for noisy scanners [5], [6], while Zhang and Singh [7] have developed their method for a low-noise scanner.

The control parameters of this support scheme are the adjacent- and in-slice range thresholds, ζ_s of (5) and ζ_p of (6), respectively. It is advisable to choose the control parameters so that $\zeta_s \geq \zeta_p$, since ζ_s matched points (adjacent scan lines) are likely to be physically farther apart than ζ_p matched points (same scan line), and so that they correspond to the characteristic length scales of the environment and the objects therein. For instance, in forest, $\zeta_s = 1.0$ m roughly describes the closest tree-to-tree distance, and $\zeta_p = 0.1$ m roughly describes the characteristic length of the stem diameter of a grown tree. For an urban environment, these values may be a bit larger. A detailed calibration of these parameters is done in Section IV-A. Note that it is easy to see how the proposed method behaves at the theoretical limits of ζ_s and ζ_p . If ζ_s and ζ_p both approach zero, all points are labeled as noise, $r_i \in F_b \forall i$, and if they are both extended ad infinitum, no point get labeled as noise, $r_i \in S_b \forall i$.

²A jump of M points is needed to advance exactly the length of one scan line.

C. Extension of Background

The key to successful image segmentation is to define the background so that separate objects appear separately on the foreground. S_b contains all the measurements r_i that are not determined as noise but may contain instances where separate objects appear as one, only because they are adjacent. In particular, if two objects are visually adjacent but present at different ranges from the scanner, we want to separate them. In addition, we want to separate smooth and nonsmooth surfaces from each other.

In general, one can think that the measurement $r_i \in S_b$ belongs either to a smooth or nonsmooth surface or an intersection of two smooth surfaces. Here, we attempt to specifically detect these intersections and temporarily redefine them as background for the segmentation purposes. The intersections are found by looking at local range gradients with a three-point finite differences formula as follows. Point i is relabeled to background, $r_i \in G_b$ where $G_b \subset S_b$, if the condition

$$\alpha < \frac{r_i - r_{i-1}}{\max(r_{i+1} - r_{i-1}, d_c)} < 1 - \alpha \quad (9)$$

does not hold. Here, the tolerance ratio $\alpha = 0.3$ and the minimal characteristic length of physical objects $d_c = 0.05$ m, and r_i is from (1). Due to the linear nature of (9), the outcome is not very sensitive to alterations of these parameters. The extended background $F_b \cup G_b$ is thus obtained as a union of the noise and the points relabeled via (9).

D. Connected Components in 2-D

A 2-D connected components algorithm is run separately on each block B_b along the data stream, where b stands for the index of the block. The input is a binary image, where the pixels belong either to the background, $F_b \cup G_b$, with value 0, or to the foreground, $S_b \setminus G_b$, with value 1 (see Fig. 1). Each pixel has four neighbors, and the image size is $k \times M$. For output, we obtain an image with labeled foreground, see step 3 in Fig. 1, where Fig. 3(c) shows the corresponding step in Algorithm 1. [Also, see Fig. 8(b) in Section IV.]

1) *Hole Filling*: After running connected components, each point $p_i \in G_b$ put to background in Section II-C is restored to foreground, and given the nonzero label most common in their 8-pixel neighborhood. This yields the image H_b , see step 3 in Fig. 1. Note that it has less ‘‘holes.’’

2) *Boundary Condition Between Blocks*: In Fig. 1, the reason behind different segment colors (and labels) between images H_b and Fig. 3(c) is that for H_b the labels of the last scan line of R_{b-1} are mapped over the block boundary to the first scan line of H_b . Each boundary point on R_{b-1} has a label and votes for that label to be also used by its respective point on H_b . Label changes are introduced if three or more points within the same segment (i.e., points with an identical label) vote for the change. Only changes from a nonzero label to another nonzero label are allowed. This way, a smooth continuous segment can retain its label even if it is split, or forked, between two consecutive blocks. Due to this one-directional boundary condition, however, if two segments would be joined they still remain separate segments



Fig. 3. MLS platforms used to gather the data. (Left) Vehicle mount for built environment. (Right) Backpack mount for forest.

in the preceding block. This is a tradeoff for not having to include recursive loops. After relabeling, and a slight abuse of notation, the final form of set H_b is obtained and it consists of nonoverlapping segments

$$H_b = \bigcup_s H_b^s \quad (10)$$

where s indexes separate segments. As visible in image H_b in Fig. 1, smooth surfaces such as the road and buildings are rather well segmented.

E. Smooth and Nonsmooth Surfaces

The support conditions of (5) and (6) are likely to result in small disjoint sets for nonsmooth surfaces. To connect these for the purpose of obtaining larger and more coherent segments, we employ a dilation operation from image morphology [38]. The set H_b is taken as input. First, each segment H_b^s of (10) having an area size of A pixels or more is labeled as background, i.e., 0, and all foreground is labeled as 1. This yields a binary image, where all smooth surfaces are omitted. Formally, we have defined that all segments of size A or larger are smooth surfaces, and that the smaller surfaces are nonsmooth and equal to the set

$$\tilde{V}_b(A) = \bigcup_{s \in \{|H_b^s| < A\}} H_b^s \quad (11)$$

Next, the foreground \tilde{V}_b is dilated in γ iterations using a four-neighbor connectedness. This is likely to connect the pixels belonging, e.g., to the same bush or tree. A 2-D connected component algorithm is rerun on this new foreground, yielding a set V_b that represents the segmentation of nonsmooth surfaces, see image V_b in Fig. 1. Hence, we show that the vegetation patches can be joined, e.g., to represent separate urban trees if the measurement geometry is favorable. Patches that are smaller than A in H_b and that are not a part of a large entity in V_b could be treated further; however, the patches are left here as a part of V_b [these are later visible as green “noise” in front of the windows of the red building in Fig. 6(a) and (b)]. The final *segmentation* set R_b is obtained from the combination of H_b and V_b . Each nonbackground point $p_i \in R_b$ retains its label from H_b except if it has a

nonbackground label in V_b . In that case, the label is obtained from V_b .

The process of separation into smooth and nonsmooth surfaces is thus controlled with the area size parameter A and the parameter γ that effectively controls the dilation length. The choice of the (area) parameter A is dependent on the block size k and the scan line length M , as these define the size of the image, $k \times M$. To maintain a similar level of algorithmic effect, the ratio of the areas, $A/(k \times M)$, should remain constant. Also, the dilation length should be restricted by keeping $\gamma \ll \sqrt{A}$ so that areas separated by the previous segmentation would not be prone to get reconnected. For example, with the parameters $k = 99$ and $M = 2154$, we use $A = 30$ and $\gamma = 2$.

F. Classification

A classification into four classes is attempted in the built environment. First, each $p_i \in F_b$ is defined as noise as in Section II-B, by looking at the point support. Second, the largest smooth patch, $p_i \in H_b^{s_{max}}$, which is connected over the blocks, is labeled as ground. Third, other smooth areas, $p_i \in H_b^{s \neq s_{max}}$, are labeled as buildings. Fourth, nonsmooth patches, $p_i \in V_b$, are labeled as vegetation. The whole above-mentioned process is managed with two support scheme control parameters, ζ_s and ζ_p , defining the characteristic length of the environment (see Section II-B), and two separation parameters A and γ , defining the division into smooth and nonsmooth surfaces (see Section II-E). See image Z_b in Fig. 1 for an example of a classification result.

In forest, a classification into noise, ground, tree stems, and other vegetation is attempted. A slightly different ground classification method is proposed due to a more obstructed line of sight and the less smooth ground surface. First, $p_i \in F_b$ is defined as noise as previously. Second, points in smooth areas $p_i \in H_b$ are labeled as tree stems. Third, points in nonsmooth patches, $p_i \in V_b$, are labeled as other vegetation. Fourth, we calculate smoothness densities

$$\rho_i = \frac{1}{k} \sum_{j=1}^k \delta_{ij}, \quad \delta_{ij} = \begin{cases} 1 & \text{if } p_i \in H_b \setminus V_b \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

along the block columns, using a larger block size $k = 255$ than for the urban environment. If $\rho_i > 1/3$, i.e., one-third of the points belong to any smooth patch, then the block column is declared as ground (the previous label is overwritten). This condition follows from the measurement geometry, as most slant downward observations are—in fact—from ground. To gain computational efficiency, ρ_i is calculated only once in every ten columns (e.g., 427 times for $M = 4270$), starting from column $h_i = 0$. The other nine columns that follow in index order are treated the same than the one from which ρ_i was calculated.

III. TEST SITES AND REFERENCE DATA

The proposed method is evaluated using two large data sets gathered from different environments (see Fig. 3). In both cases, the scanning plane is in a slant, close to a vertical orientation to enable a favorable scanning geometry in detecting vertical objects. We use raw data that have not

been preprocessed but that may be connected onto an existing classification, i.e., a reference. The data from EuroSDR Espoonlahti built environment test site are captured using the FGI ROAMER platform [39], mounting a Faro Photon 80 scanner in a 2-D helical mode with 48-Hz frequency and $M = 2154$ points per each scan line. The used scanner emits a continuous wave, which is prone to cause more noise than a pulsed time-of-flight scanner. This sets more challenge for the proposed method and offers a decent test case considering the single-photon LIDARs that are vulnerable to background illumination. The data from a CAR method that we use for comparison, i.e., as our reference, are published in [13] and span a distance of 900 m containing 13 million classified points. The vehicle driving speed was 30 km/h and the average scan line spacing was 17 cm. In detail, PRC noise, ground, and building results are matched against the noise, ground, and building results in [13], respectively. PRC vegetation class corresponds to the reference tree class, and the PRC other class contains the rest of the reference classes, including cars, road signs, and traffic lights.

The forest data of 85 million points are captured with Akhka backpack [20] (see Fig. 3), mounting a Faro Focus 3-D 120 scanner in a 2-D helical mode with 97-Hz frequency and $M = 4270$ points per each scan line. The dead angle of the scanner, which is pointing downward, is 55° . The forest test area is about 2000 m^2 ($40 \text{ m} \times 50 \text{ m}$) rectangular mature forest plot in an area behind FGI in Masala (60.15°N , 24.53°E), southern Finland. Tree reference locations and widths are obtained from field measurements, and the ensuing reference tree trunk classification is explained in detail in [20], where Liang *et al.* used this reference to evaluate the results of their object-based CAR method. Briefly, tree locations were measured from a fixed location inside the plot using a Trimble 5602 DR 200+ total station. The total station was oriented to the same coordinate system as backpack data using virtual reference station-GNSS measurements. The stem perimeter of each tree was measured by a steel tape to the nearest millimeter at breast height (1.3 m above the ground level), from which diameter at breast heights (DBHs) were calculated. A reference classification is obtained for each point by placing a 3-D cylinder onto each tree location obtained by the previous field measurements. Cylinder z -coordinate direction is obtained from the earth-centered, earth-fixed coordinate system (coordinate transform is done using GRS80 ellipsoidal model). Radius for each cylinder is defined as two times the field measured DBH to compensate for inclined tree stems and backpack registration errors in the data. The height of each cylinder is 4.0 m starting from 1.0 m below measured DBH height. Points residing outside the cylinders and under 0.5 m from the nearest reference tree DBH altitude are defined as reference ground, which (nonideally) includes some low vegetation. Noise reference contains all points that are not labeled as reference tree stem or reference ground. In comparison, points outside the test area are omitted.

IV. RESULTS

We obtain a stream of data from a 2-D laser scanner that sees ground but otherwise has an unknown pose. The stream

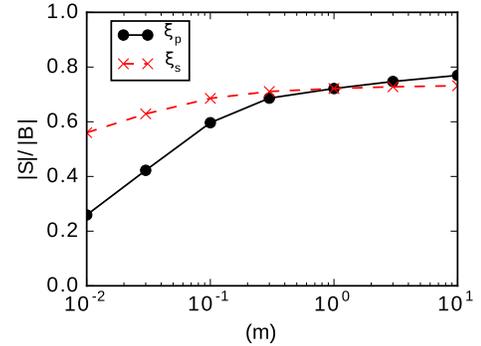


Fig. 4. Ratio of supported points to total points $|S|/|B| = \sum_b |S_b|/|B_b|$ as a function of ζ_p (ζ_s) while keeping $\zeta_s = 1.0$ m ($\zeta_p = 1.0$ m) fixed.

consists of unregistered points $p_i(r_i)$, with r_i of (1). Our goal is to separate noise, ground, buildings, and vegetation points from each other in the urban environment and the tree stem points and ground points from the other points in the forest. To evaluate the outcome, the following classification measures are used. Recall is the amount of true positives divided by the sum of true positives and false negatives. Accuracy is the number of correct predictions divided by the total number of predictions. Precision is the amount of true positives divided by all positives.

A. Choice of Control Parameters

The stability of the proposed method with respect to the control parameters ζ_p and ζ_s in Section II-B is examined with the help of Fig. 4. Therein, the ratio of supported points against total points $\sum_b |S_b|/|B_b|$ as a function of ζ_p while keeping $\zeta_s = 1.0$ m fixed, and vice versa, is shown. The ratio is weakly dependent on the control parameters, which means that the method produces similar results regardless of the exact choice for the numerical value of these parameters. Specifically, the value for ζ_s should be in the range of $[0.01, 10]$ m, and the value for ζ_p in the range of $[0.3, 10]$ m. We choose the parameters $\zeta_s = 1.0$ m and $\zeta_p = 1.0$ m and $\zeta_s = 1.0$ m and $\zeta_p = 0.1$ for a more elaborate evaluation of the built environment results.

The spatial correlations should be statistically significant so that the points do not get supported by random. For the support condition of (4), $\zeta_s = 1.0$ m, and $\zeta_p = 1.0$ m, the probability of uniformly distributed random range measurements, $x_i \in [0, 120]$ m, supporting each other is 0.004 for N_6 and 0.007 for N_8 . This yields a very low joint probability of 3×10^{-5} . In other words, the proposed method is not likely to support measurements at random, and when ζ_s and ζ_p get smaller, this probability gets even smaller.

B. Urban Environment

Confusion matrices normalized with the amount of reference points are given in Table I and the overview of the data is shown in Fig. 5. With $\zeta_s = 1.0$ and $\zeta_p = 1.0$, a ground recall is at 96.4% (see Table I). This is an excellent result, because—to the best of our knowledge—this is the first

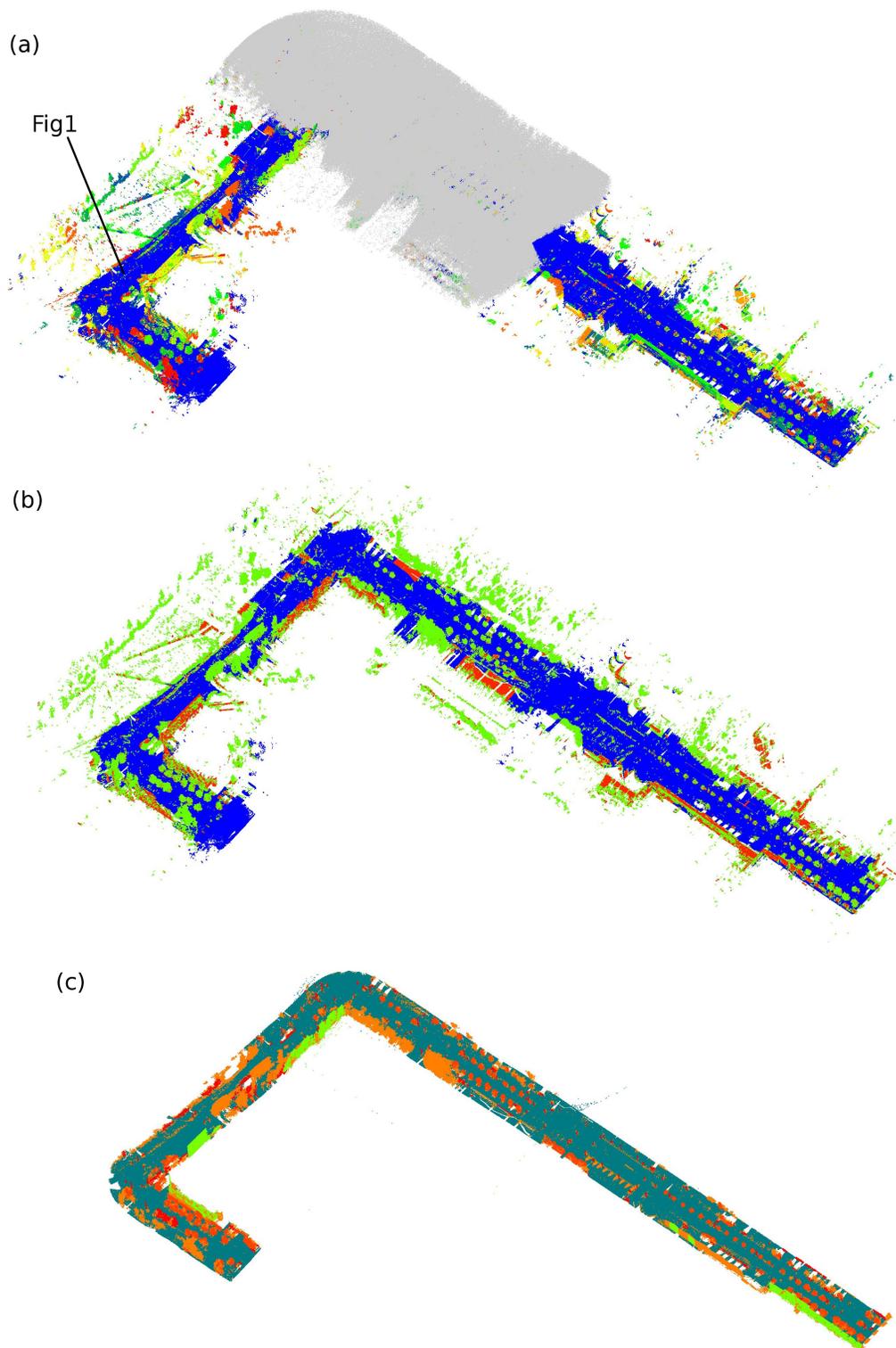


Fig. 5. Results of the proposed PRC method from Espoonlahti test site on top of the registered reference data. (a) After segmentation. Gray points: data from one part of the trajectory has purposefully not been noise filtered to visualize the amount of noise in the data. Black line: approximate location to where the block data shown in Fig. 1 is registered. (b) Classification is done to ground (blue), buildings (red), and vegetation (green). (c) Our reference, CAR data from [13], employs more specific labels, and thus is shown in different colors for visualization purposes (see Section III).

time when the urban ground classification is done with a 2-D scanner with an unknown pose. However, note that noise close to the ground results in false positives by PRC, also visible in Table I. Approximately, 29% of noise points are presumed

to be ground. To see whether this is because the noise filtering is not sufficiently aggressive, we alter the parameters. With $\xi_p = 0.1$, the noise recall yields a slightly better result, but there is a significant drop in true ground, true vegetation, and

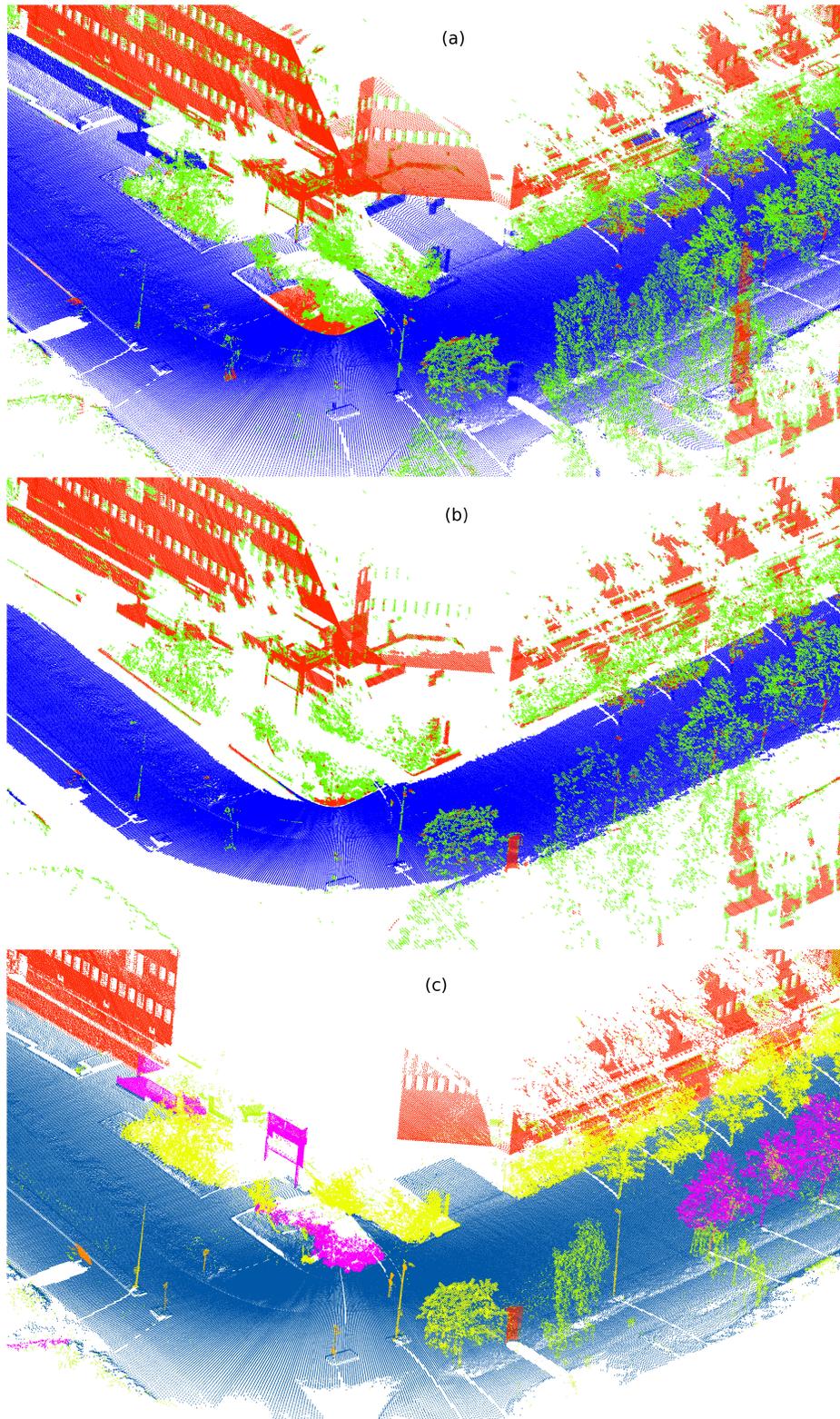


Fig. 6. Close up view on the classification results shown on top of the registered reference data. (a) Less and (b) more aggressive PRC outcomes with $\zeta_s = 1.0$ m and $\zeta_p = 1.0$ m, and $\zeta_s = 1.0$ m and $\zeta_p = 0.1$ m, respectively, are compared against (c) reference CAR data from [13]. Ground is blue. Buildings are red. Vegetation is shown in green in (a) and (b), and in green (vegetation) and in yellow (trees) in (c). In the reference (c), magenta points belong to the “other” class. One section of the building wall is shown for the PRC but not for the reference.

true buildings (see Table I). Ground and noise labeling are affected only by the two support scheme parameters ζ_s and ζ_p , with true noise labeling being quite robust with respect

to them. Lowering the value of these reduces the detection percent of true ground, as points increasingly get labeled as noise (see Fig. 6). Note especially the detected width of the

TABLE I

CONFUSION MATRIX USING PARAMETERS $\xi_s = 1.0$ AND $\xi_p = 1.0$ (TOP) AND $\xi_s = 1.0$ AND $\xi_p = 0.1$ (BOTTOM). FOR BOTH $A = 30$ AND $\gamma = 2$. RECALL VALUES ARE SHOWN, AND THOSE OVER 0.20 AND 0.50 ARE SHOWN IN YELLOW AND GREEN, RESPECTIVELY, FOR READER'S CONVENIENCE. CLASSIFICATION RESULTS ARE COMPARED AGAINST OUR REFERENCE, CAR DATA FROM [13]

Reference	Predicted			
	Noise	Ground	Vegetation	Building
$\xi_p = 1.0$				
Noise	0.541	0.286	0.095	0.077
Ground	0.003	0.964	0.010	0.023
Vegetation	0.113	0.084	0.665	0.137
Building	0.042	0.222	0.125	0.611
Other	0.165	0.592	0.120	0.123
$\xi_p = 0.1$				
Noise	0.614	0.263	0.058	0.065
Ground	0.159	0.804	0.014	0.023
Vegetation	0.373	0.024	0.467	0.136
Building	0.169	0.025	0.119	0.687
Other	0.222	0.340	0.153	0.285

road. The aggressiveness of the PRC filter, i.e., how wide is the detected road, is controllable with these support scheme control parameters.

The connected component parameters A and γ in Section II-E affect the separation between smooth and non-smooth surfaces. With a block size of 99 scan lines, best results are obtained around $A = 30$ and $\gamma = 2$, see Table I with $\xi_p = 1.0$. Building and vegetation point classification recalls are 61.1% and 66.5%, respectively. If γ is set to zero, false and true positives in vegetation decrease, while false and true positives in building increase. If γ is set higher than 2, the opposite happens. At $\gamma = 20$, the amount of building positives is negligible. Vegetation areas are then increased too aggressively, which reduces correct labeling.

Altering the area parameter A results in minor changes in classification results. With a lower value $A = 10$, building positives are slightly increased at the cost of declining vegetation positives and with a higher $A = 100$ vice versa. Changes are within 10 percentile units. Furthermore, the impact of A decreases as γ is increased.

For reader's convenience, the approximate location of the data processed through steps shown in Fig. 1 is shown in Fig. 5(a). The same approximate location is used also for Figs. 6 and 7. Note the same car that is present in each figure. The car shape is distinctly different in the registered form in Figs. 6 and 7 than in the raw data visualization in Fig. 1, because of the tilted scanner orientation and the way the range image is constructed.

Note that the reference data are not perfect. For example, in Fig. 6, the reference does not include all building points. The shopping mall is missing a large section of its facade close to the center of the image, and on the lower right corner, the partially captured building facades are correctly classified for PRC results although they are not so in the reference. These points correctly detected by the proposed PRC method show as a rather high amount, 7.7%, of "false positives" (see Table I). The same applies to vegetation and ground when PRC filtering is less aggressive than the reference (see Fig. 6). Points labeled

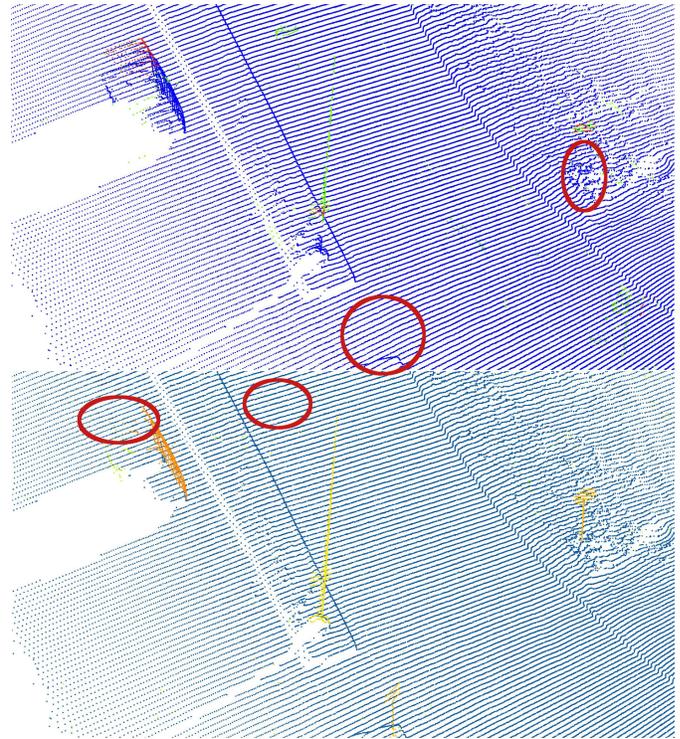


Fig. 7. (Top) Comparison of PRC against (Bottom) reference with respect to filtering. Points representing the street sign poles vanish (red circles, top) as the requirement of supportive measurements is not met, i.e., one line of adjacent points is not enough. On the other hand, some important points are recovered by the PRC that was dismissed in the reference (red circles, bottom). PRC shows the street signs in red indicating a smooth surface.

as "other" for reference do not have a corresponding PRC class, as shown in Table I, but these points are so few that their overall impact on the results is taken as negligible.

The added benefit of spatial correlations is shown in Fig. 7, where on top of the rightmost red circle, smooth surface points of a road sign are detected (points in red). Similarly, the road sign attached to the lighting pole in the center of the image is revealed by red points indicating a smooth surface. In contrast, the lighting pole is classified as a nonsmooth surface. Objects do not need to be continuous for pointwise classification. See the missing lamp distinct from the lighting pole surrounded with a red circle in the reference image shown in Fig. 7. It is restored in the PRC image of the same figure.

The proposed method has limitations with respect to thin objects and scanning geometry. Fig. 7 shows a closeup view on classification results on Espoonlahti test site data. Streetlight poles are not wholly captured due to the support scheme. If the pole is so thin, or captured from so far away, that it consists of a single scan strip, the support filter treats the pole points as outliers. Object points close to the ground are classified as ground since the support algorithm cannot differentiate a strong enough contrast in the range between these points and their background. The distinguishing capacity of PRC may to some extent be regulated by the support scheme parameters, see the advertisement hoarding change color from blue [Fig. 6(a)] to red [Fig. 6(b)] near the center of the image.

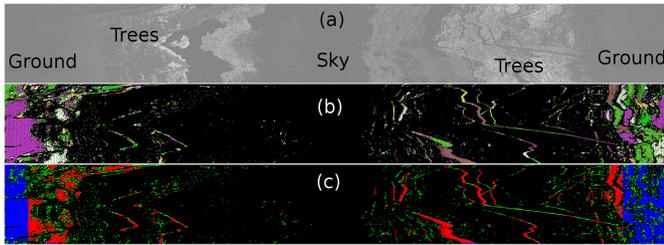


Fig. 8. (a) $k \times M$ block B_b of forest range data, with $k = 255$ and $M = 4270$. (b) PRC segmentation result. (c) PRC classification into noise (gray), ground (blue), tree stems (red), and other vegetation (green). Ground (blue) recall is high only near the trajectory.

The following tradeoff issue on whether to relax or to tighten the support scheme parameters is brought up in Section V.

C. Finding Trees in the Forest

The objective is to classify the points with labels noise, ground, tree stems, and other vegetation. However, since the reference only contains classes noise, ground, and tree stems, we match the predicted “other vegetation” with reference noise.

Finding the tree stem points (5% of total points) is like finding needles in the haystack since the stems may be partially hidden by forest basin, branches, and smaller plants. This means that the separation between the tree stem points and the forest basin points is also hard, even with G_b of Section II-C. Therefore, we employ the subblock smoothness density of (12) to label ground.

“The haystack,” i.e., a block B_b of range data from the forest, is visualized in Fig. 8(a). The scanner shown in Fig. 3 (right) is aligned so that the 55° dead angle points toward the ground, meaning that the mid part of the scan line corresponds to the sky. Fig. 8(b) and (c) shows the following segmentation and classification results. Note how the tree stems shift vertically in the block image, indicating that the scan platform has rotated at that time. Tree stems appear continuous but regardless of this do not form continuous segments. Hence, the classification based on the area size of the segment misinterprets some parts of the stem to the class “other vegetation.”

The range block shown in Fig. 8 is plotted on top of the registered reference data for visualization purposes in Fig. 9. The previously mentioned rotation of the scan platform is visible in which there seems to be a static axis of rotation on the right side of the traveled path. This is due to the fact that the actual rotation takes place while the platform is on the move along the path. Any classification to distances over 20 m becomes difficult since the line of sight is obstructed by forest vegetation, compare Fig. 9(a) and (b). At the limit of long range, points are increasingly labeled as noise.

The accuracy and precision of the tree stem and the ground classes are the most important quantitative factors application-wise. The calibration of the two support parameters is done similarly as for the built environment. The tree stem points are best separated from “the haystack” with $\xi_s = 0.01$ and $\xi_p = 0.01$. With these parameter values, 74.6% of points do

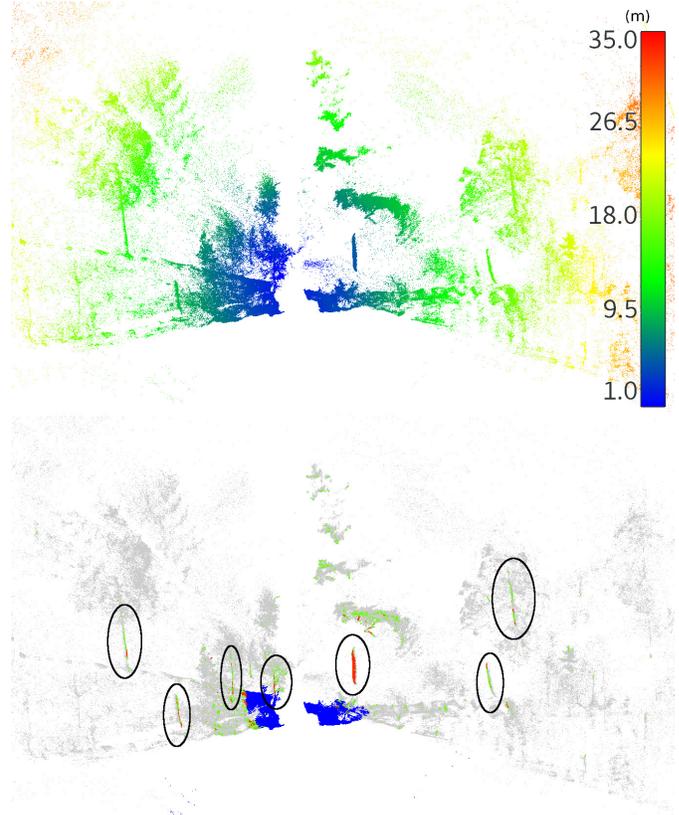


Fig. 9. (a) Block B_b of points after 3-D registration for visual purposes. Coloring is based on the measured range (m). Walking route is in white. (b) Visualization of the PRC classification result of Fig. 8, with noise (gray), ground (blue), tree stems (red), and other vegetation (green). Detected tree stem points are encircled for visual clarity.

not obtain support and are treated as noise. The results are visualized in Fig. 10. Raw scanner data points p_i are classified into three categories, and each point representing either noise, ground, or a tree stem. Confusion matrix results are displayed in Table II. The prediction of the proposed PRC method for tree stems is conservative, yielding little false positives and a 50.0% recall. Tree stem classification accuracy is 96.1% and precision 58.0%. Ground classification accuracy is 60.0% and precision 91.3%, with 31.0% recall. There is a sharp contrast between this low *overall* ground recall rate that is due to trees occluding ground points farther to the trajectory, and the high *local* ground recall rate close to the traveled path [see Figs. 9(b) and 10(a)]. Note that these results are also affected by shortcomings in the elevation precision of the ground reference, as explained in Section III.

For sensitivity analysis purposes, we used also much larger values $\xi_s = 1.0$ m, which roughly describes the closest tree-to-tree distance, and $\xi_p = 0.1$ m, which roughly describes the characteristic length of the stem diameter of a grown tree. With these parameters, 34.1% of all points get labeled as noise. Based on visual inspection (see Fig. 11), the detection of outliers is quite successful even for this relaxed set of parameters. Not only the outliers that are clearly outside the data set, e.g., above the tree level, are removed but also those outliers that reside within the forest and between the trees

TABLE II

CONFUSION MATRICES FOR THE PROPOSED METHOD IN FOREST TEST SITE, RECALL VALUES (LEFT), AND UNNORMED SAMPLES (RIGHT), $\xi_s = 0.01$ AND $\xi_p = 0.01$. RECALL VALUES OVER 0.20 AND 0.50 ARE SHOWN IN YELLOW AND GREEN, RESPECTIVELY, FOR READER'S CONVENIENCE. TREE STEM CLASSIFICATION ACCURACY IS 96.1% AND PRECISION IS 58.0%, EVEN THOUGH TREE STEM POINTS ACCOUNT FOR ONLY 5% OF THE TOTAL POINTS. PREDICTED "OTHER VEGETATION" IS MATCHED AGAINST THE NOISE CLASS IN THE REFERENCE

Reference	Predicted					
	Noise (other)	Ground	Tree stem	Noise (other)	Ground	Tree stem
Noise (other)	0.93	0.02	0.04	15285576	336704	232019
Ground	0.65	0.31	0.03	14741263	6934955	429704
Tree stem	0.28	0.18	0.50	515205	326183	914452

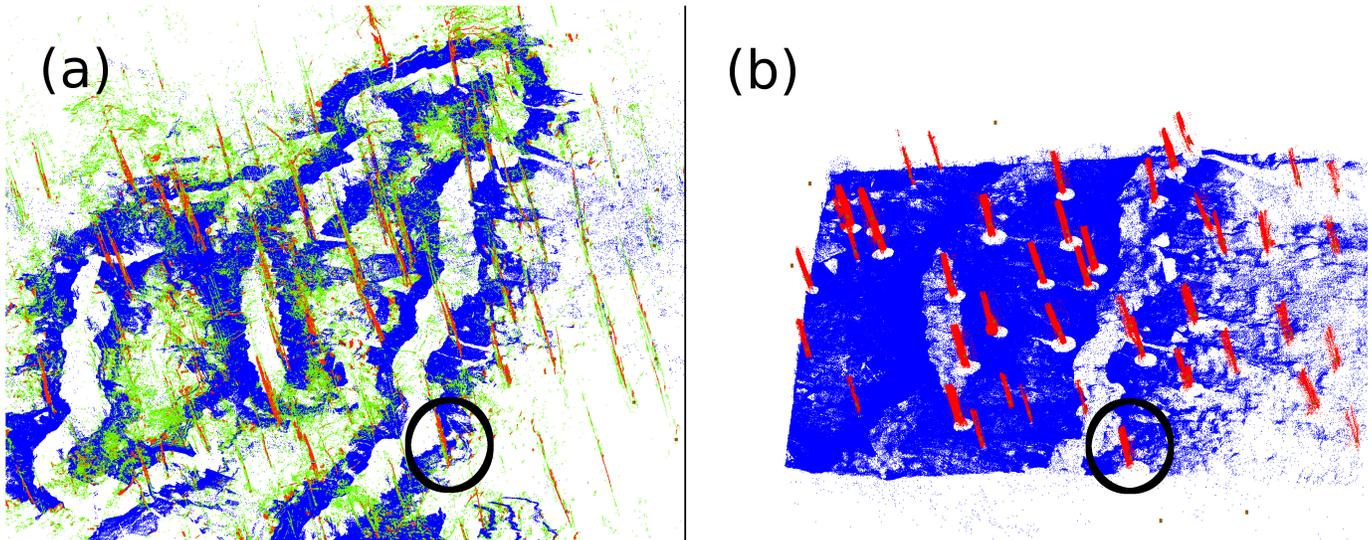


Fig. 10. Visualization of PRC results on top of the 3-D registered reference point cloud. (a) PRC classification result for the forest test site with ground in blue. Smooth (red) and nonsmooth (green) surfaces are taken to represent tree stems and leaves, respectively. The white path is the traveled route. (b) Reference tree locations (DBH) are shown as large brown dots, with points belonging to reference tree stem (red) and ground (blue) classes. Black circle: visual reference and surrounds the same tree in both plots.

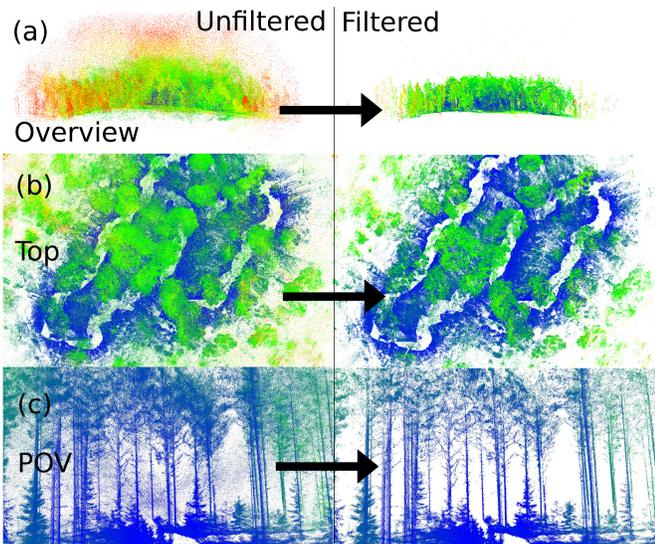


Fig. 11. Local support filter in action, using $\xi_s = 1.0$ m, $\xi_p = 0.1$ m. (a) Overview, (b) top view, and (c) point of view (POV) of the forest data with PRC results on top of 3-D registered reference data. Unfiltered and filtered point clouds are shown on left and right, respectively. Note that filtering is done before point cloud registration, although registered point clouds are shown for visualization purposes. Coloring is based on range.

appear to vanish. Therefore, the noise filtering is robust with respect to the control parameters. However, with $\xi_s = 1.0$ and $\xi_p = 0.1$, PRC yields out a bare 20.4% tree stem recall with 41.3% precision. This only means that using too long length

scale values leads to fewer tree stem points being successfully separated from the occluding foliage (as we are looking for too long needles in the haystack).

D. Real-Time Capability

For the results presented in Section IV, the proposed algorithm was run on previously captured data so that classification results can be compared against a known reference. Here, we discuss the real-time capability of the algorithm. The Espoonlahti test site data contain 12.9 M points and was captured with a 48-Hz frequency in 125 s. In comparison, CPU run times for the proposed method are around 100 s with unoptimized code, although the connected components and dilation routines are from an optimized outside library. The processing rate is 0.13 Mpts/s. With 48-Hz frequency, a block size of $k = 99$ scan lines represents the data captured in 2 s. For a higher update rate, continuous processing is suggested.

For the forest data of 85 million points, the respective run time is 460 s, with a scan line and k both about two times as large leading to a 4–5 times larger block size. The increase in the processing rate from 0.13 to 0.18 Mpts/s, however, mostly follows from tight filtering. With an almost 100-Hz frequency, a block size of $k = 255$ scan lines represents the data captured in 2.5 s. The proposed method is real-time capable and, therefore, very likely applicable for an online algorithm. Also, a more intense data capture rate can likely

be handled since the block operations can be straightforwardly parallelized.

V. DISCUSSION

A. On Results

The reference LIDAR data, urban CAR results in [13], and the tree reference in [20] allowed us to test the multipurpose nature and the shortcomings of the proposed method in two very different environments. The MLS-based tests were carried out in an urban environment using a vehicle and in forest using a human-carried backpack. Compared to the urban environment, the forest basin is a geometrically complex entity consisting of a mix of different plants. Trees growing out of this mix are hard to distinguish. This is especially true close to the scanner when the scanning geometry is unideal for an effective range separation. To avoid this, we evaluated the subblock smoothness density of (12), but scan angle reduction or line [14] or cylinder [27] regression could also be used.

For the chosen parameters, the proposed method performs especially well in the urban environment with ground, vegetation, and building recalls being 96%, 67%, and 61%, respectively. In the forest, tree stem classification resulted in 50.0% recall and in accuracy and precision of 96.1% and 58.0%, respectively. Among other things, these results advocate for a new way to design SLAM algorithms. For example, in contrast to matching all available points among themselves [40], points belonging to the same class can be matched together to calculate the pose of the system. Especially, if there are ways to improve the classification results by smartly reducing the overall aggressiveness of the filter, e.g., with multiple filter layers computed with different sets of support scheme parameters or by redefining the filtering so that individual critical filter values can be computed for each point.

The support condition of the proposed PRC method behaves as expected between the theoretical limits, i.e., increasing the length threshold parameters ξ_s and ξ_p decreases the amount of points labeled as noise. Also, as shown in Fig. 4, the support scheme is quite robust with respect to the choice of control parameters ξ_s and ξ_p . There is, however, some tradeoff in whether to relax or tighten the support scheme parameters ξ_s and ξ_p . For example, by relaxing the support scheme parameters, one can conserve thin objects such as lighting poles, however, so that the lower part of them is predicted to be ground. In general, approaching the long length scale limit causes undersegmentation (as discussed in Section IV-C). On the other hand, by tightening the support scheme parameters, the lower part of objects is less likely predicted as ground, but then the thin objects disappear (see the example in Fig. 7). In the light of this tradeoff, we put the PRC method into a real test by trying to classify tree stem points in forest where both the undersegmentation and the loss of the detection capability of thin stems should be avoided. In addition, the ranging precision of the measurement device is likely to become a relevant issue in the tight limit. This issue, however, is beyond the scope of this paper. The scanner used for the forest data has a ranging error of 0.002 m (at 25 m),³ which is a lot smaller than the lowest used computational value of 0.01 m.

³Faro Focus 3-D 120 scanner, specifications from www.faro.com

In addition to the support scheme parameters ξ_s and ξ_p , two separation parameters were introduced. Disjoint patches are merged with the parameter γ that marks the number of morphological dilations done on the vegetation segments. Extension to background in (9) is triggered for two points near each stepwise change in the range measurements within a scan line. Thus, keeping the value $\gamma = 2$ is advisable. If γ is too large, points labeled as building also become defined as vegetation. The optimal value for the minimum smooth area A depends on the size of the block image B . In urban environment, the classification results (of Section IV-B) appeared to be quite robust with respect to the choice of A .

B. On Potential and Shortcomings

The explored PRC scheme may open new horizons in (distributed) LIDAR data processing (see Table III). Notably, data from moving 2-D laser scanners can be preclassified before registration. Since SLAM can be run on a filtered and reduced stream of data [6], preclassifying the raw data allows for SLAM enhancements. Continuing the idea of hierarchy presented in [19], local spatial correlations from PRC and the global 3-D geometry from CAR may be combined to perform enhanced hierarchical classification. Machine learning applications exist, also. On the one hand, raw data are ideal for artificial intelligence (AI) and the manual work in obtaining AI training data could be reduced with PRC methods. On the other hand, AI methods could be used to adjust the PRC scheme, e.g., the neighborhood properties, to provide better results. In addition, PRC provides a good alternative for black-boxed firmware that filters points before the points are extracted from the scanner, especially since it is working with the similar spatial correlations found useful in chip-level data filtering [8]. This should be of interest for researchers. Reflecting to this background, we have shown that the proposed method is a viable classifier, although, as expected, it falls short from CAR methods in classification metrics.

In principle, the PRC can be thought to be a way to compress the information about the spatial correlations between a point and its neighbors. Typical to geoinformation industry, finalized point clouds seldom contain this line-of-sight information that was available when the data were captured. Hence, preregistration classified points may be used to carry these spatial correlations along in a compressed form of a single class label, familiar, e.g., to the ASPRS LAS-format—even if conventional means, e.g., satellite positioning, are used to register points in a postprocessing step.

The proposed method is independent of radiometry, and its only geometrical prerequisite is that the scanner sees the ground (see Section II-F, steps two and four for the built environment and the forest, respectively). However, this makes the method vulnerable to reflecting surfaces. Adding a radiometric component to detect these reflections is to be done as a part of the future work. In addition, future research focuses on improving the method by including machine learning techniques. The neighborhood shapes introduced in Section II-B could then be optimized, e.g., for SLAM [9] and forest industry applications [16]. Also, the thin and elongated road

TABLE III

SUITABILITY OF CLASSIFICATION METHODS FOR MLS DATA: 1) CAR; 2) MAPS; AND 3) PRC TO DIFFERENT USE CASES. CITATIONS OFFER EXAMPLES

Use case	CAR	MAPS	PRC
SLAM	N/A	Partly after SLAM, partly before SLAM	Enhance, enable for noisy scanners [6]
Object-based classification	3D objects [20], [19]	Continuous registration [15]	N/A (distorted 2D profiles)
Point-wise classification	3D adjacency [2]	3D profiles or local occupancy grids [4], [3]	Local 2D adjacency (this work, Figs. 1 and 8)
Minimal requirements	full SLAM or GNSS registration	3D LIDAR or 3D relative registration	2D LIDAR

inventory items such as lighting or road sign poles might be recoverable by optimizing shape properties for the segmented areas after running connected components. A large part of the data may be omitted if only the boundaries of smooth areas are stored, especially if storing is done in vectored forms. Smart sampling as a function of the scan range may also provide efficient means to compress and analyze the data.

VI. CONCLUSION

PRC means segmenting and classifying the mobile-scanned LIDAR data before and independently of data registration. The chronological order and the spatial correlations between the range measurements are employed to perform classification. Hence, PRC significantly differs from the CAR and MAPS methods with respect to the input data. The raw scanner data stream must be completely unfiltered and the range measurements in an emission-based chronological order. Such data are typically not stored, nor are they available after traditional data processing, but we have managed to access the raw data dumps of two previous CAR works so that our results could be compared to theirs.

PRC is a complementary paradigm, in addition to CAR and MAPS, in performing MLS classification of LIDAR points. Where CAR methods excel in classification metrics, PRC has potential for real-time solutions. Where MAPS methods excel for 3-D scanners or in SLAM integration, PRC does so for 2-D scanners or in being independent of SLAM. Importantly, however, PRC may be used with or without CAR (or MAPS) methods to allow for new system designs and improved processing performance. Here, we have demonstrated the computational efficiency of one PRC method and shown its viability with respect to classification metrics.

The proposed PRC algorithm effectively encapsulates the spatial correlations that exist in the raw data stream of a mobile laser scanner into a pointwise classification label. The proposed algorithm is viable in drastically different environments with a change in two essential control parameters defining the characteristic length scales. For instance, forest data differ from the urban environment data in which the objects are both closer to the scanner, but also closer to each other. Therefore, the characteristic length scale of spatial correlations is also smaller. We have shown that by essentially calibrating the two control parameters to match this typical spatial correlation length of the environment, the proposed method manages to classify raw range measurements into

either smooth surfaces such as buildings, tree stems, or ground, or into nonsmooth surfaces such as vegetation or noise. The proposed PRC method is shown to be real-time capable. Finally, the importance of PRC, in general, is likely to increase in the future, as PRC may be especially useful with the upcoming single-photon LIDARs that are vulnerable to background illumination and their data thus contain a lot of noise.

REFERENCES

- [1] G. Vosselman, "Point cloud segmentation for urban scene classification," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. 1, pp. 257–262, Nov. 2013.
- [2] M. Weinmann, B. Jutzi, S. Hinz, and C. Mallet, "Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers," *ISPRS J. Photogramm. Remote Sens.*, vol. 105, pp. 286–304, Jul. 2015.
- [3] B. Douillard, A. Brooks, and F. Ramos, "A 3D laser and vision based classifier," in *Proc. Int. Conf. Intell. Sensors Sensor Netw. Inf. Process. (ISSNIP)*, Dec. 2009, pp. 295–300.
- [4] I. Bogoslavskyi and C. Stachniss, "Efficient online segmentation for sparse 3D laser scans," *PFG–J. Photogramm. Remote Sens. Geoinf. Sci.*, vol. 85, no. 1, pp. 41–52, 2017.
- [5] V. V. Lehtola, J.-P. Virtanen, P. Rönholm, and A. Nüchter, "Localization corrections for mobile laser scanner using local support-based outlier filtering," *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. 3, no. 4, pp. 81–88, 2016.
- [6] V. V. Lehtola, J.-P. Virtanen, M. T. Vaaja, H. Hyypä, and A. Nüchter, "Localization of a mobile laser scanner via dimensional reduction," *ISPRS J. Photogramm. Remote Sens.*, vol. 121, pp. 48–59, Nov. 2016.
- [7] J. Zhang and S. Singh, "LOAM: Lidar odometry and mapping in real-time," in *Proc. Robot. Sci. Syst.*, Berkeley, CA, USA, 2014, p. 9.
- [8] A. Kirmani *et al.*, "First-photon imaging," *Science*, vol. 343, no. 6166, pp. 58–61, 2014.
- [9] A. Kukko, R. Kajaluoto, H. Kaartinen, V. V. Lehtola, A. Jaakkola, and J. Hyypä, "Graph SLAM correction for single scanner MLS forest data under boreal forest canopy," *ISPRS J. Photogramm. Remote Sens.*, vol. 132, pp. 199–209, Oct. 2017.
- [10] T. Luettel, M. Himmelsbach, and H.-J. Wuensche, "Autonomous ground vehicles-concepts and a path to the future," *Proc. IEEE*, vol. 100, pp. 1831–1839, May 2012. doi: 10.1109/JPROC.2012.2189803.
- [11] E. Puttonen, M. Lehtomäki, H. Kaartinen, L. Zhu, A. Kukko, and A. Jaakkola, "Improved sampling for terrestrial and mobile laser scanner point cloud data," *Remote Sens.*, vol. 5, no. 4, pp. 1754–1773, 2013. [Online]. Available: <http://www.mdpi.com/2072-4292/5/4/1754>
- [12] J. Kostamovaara *et al.*, "On laser ranging based on high-speed/energy laser diode pulses and single-photon detection techniques," *IEEE Photon. J.*, vol. 7, no. 2, Apr. 2015, Art. no. 7800215.
- [13] M. Lehtomäki *et al.*, "Object classification and recognition from mobile laser scanning point clouds in a road environment," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 2, pp. 1226–1239, Feb. 2016.
- [14] H. Hyyti and A. Visala, "Feature based modeling and mapping of tree trunks and natural terrain using 3D laser scanner measurement system," *IFAC Proc. Volumes*, vol. 46, no. 10, pp. 248–255, 2013.
- [15] J.-F. Lalonde, N. Vandapel, D. F. Huber, and M. Hebert, "Natural terrain classification using three-dimensional lidar data for ground robot mobility," *J. Field Robot.*, vol. 23, no. 10, pp. 839–861, Oct. 2006.

- [16] H. Hyyti, V. V. Lehtola, and A. Visala, "Forestry crane posture estimation with a two-dimensional laser scanner," *J. Field Robot.*, vol. 35, no. 7, pp. 1025–1049, 2018.
- [17] H. Houshiar and A. Nüchter, "3D point cloud compression using conventional image compression for efficient data transmission," in *Proc. Int. Conf. Inf. Commun. Automat. Technol. (ICAT)*, Oct. 2015, pp. 1–8.
- [18] J. Zhang and S. Singh, "Low-drift and real-time lidar odometry and mapping," *Auton. Robots*, vol. 41, no. 2, pp. 401–416, 2016.
- [19] B. Yang, Z. Dong, G. Zhao, and W. Dai, "Hierarchical extraction of urban objects from mobile laser scanning data," *ISPRS J. Photogram. Remote Sens.*, vol. 99, pp. 45–57, Jan. 2015.
- [20] X. Liang, A. Kukko, H. Kaartinen, J. Hyyppä, X. Yu, A. Jaakkola, and Y. Wang, "Possibilities of a personal laser scanning system for forest mapping and ecosystem services," *Sensors*, vol. 14, no. 1, pp. 1228–1248, 2014. [Online]. Available: <http://www.mdpi.com/1424-8220/14/1/1228>
- [21] R. Martin, H. Bernhard, O. E. Sander, and V. George, "Feasibility of facade footprint extraction from mobile laser scanning data," *Photogramm.-Fernerkundung-Geoinf.*, vol. 2011, no. 3, pp. 97–107, Jun. 2011.
- [22] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: A survey," *Mach. Vis. Appl.*, vol. 25, no. 3, pp. 727–745, 2014.
- [23] C. Cabo, A. Kukko, S. García-Cortés, H. Kaartinen, J. Hyyppä, and C. Ordoñez, "An algorithm for automatic road asphalt edge delineation from mobile laser scanner data using the line clouds concept," *Remote Sens.*, vol. 8, no. 9, p. 740, 2016.
- [24] D. Craciun, J.-E. Deschaud, and F. Goulette, "Automatic ground surface reconstruction from mobile laser systems for driving simulation engines," *Simulation*, vol. 93, pp. 201–211, Mar. 2017.
- [25] X. Liang, J. Hyyppä, A. Kukko, H. Kaartinen, A. Jaakkola, and X. Yu, "The use of a mobile laser scanning system for mapping large forest plots," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 9, pp. 1504–1508, Sep. 2014.
- [26] E. Ayrey *et al.*, "Layer stacking: A novel algorithm for individual forest tree segmentation from LiDAR point clouds," *Can. J. Remote Sens.*, vol. 43, no. 1, pp. 16–27, 2017.
- [27] E. Puttonen, A. Krooks, H. Kaartinen, and S. Kaasalainen, "Ground level determination in forested environment with utilization of a scanner-centered terrestrial laser scanning configuration," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 3, pp. 616–620, Mar. 2015.
- [28] A. Jaakkola, J. Hyyppä, H. Hyyppä, and A. Kukko, "Retrieval algorithms for road surface modelling using laser-based mobile mapping," *Sensors*, vol. 8, no. 9, pp. 5238–5249, 2008.
- [29] Q. Li, L. Chen, M. Li, S.-L. Shaw, and A. Nüchter, "A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 540–555, Feb. 2014.
- [30] V. Nguyen, A. Martinelli, N. Tomatis, and R. Siegwart, "A comparison of line extraction algorithms using 2D laser rangefinder for indoor mobile robotics," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Aug. 2005, pp. 1929–1934.
- [31] S.-Y. An, L.-K. Lee, and S.-Y. Oh, "Fast incremental 3D plane extraction from a collection of 2D line segments for 3D mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 4530–4537.
- [32] A. Nguyen and B. Le, "3D point cloud segmentation: A survey," in *Proc. 6th IEEE Conf. Robot. Automat. Mechatron. (RAM)*, Nov. 2013, pp. 225–230.
- [33] I. Chauhan, C. Brenner, R. D. Garg, and M. Parida, "A new approach to 3D dense LiDAR data classification in urban environment," *J. Indian Soc. Remote Sens.*, vol. 42, no. 3, pp. 673–678, 2014. doi: [10.1007/s12524-013-0354-4](https://doi.org/10.1007/s12524-013-0354-4).
- [34] A. J. Trevor, S. Gedikli, R. B. Rusu, and H. I. Christensen, "Efficient organized point cloud segmentation with connected components," in *Proc. Semantic Perception Mapping Explor. (SPME)*, 2013.
- [35] J. McCormac, A. Handa, A. Davison, and S. Leutenegger. (2016). "Semanticfusion: Dense 3D semantic mapping with convolutional neural networks." [Online]. Available: <https://arxiv.org/abs/1609.05130>
- [36] K. M. Wurm, H. Kretschmar, R. Kümmerle, C. Stachniss, and W. Burgard, "Identifying vegetation from laser data in structured outdoor environments," *Robot. Auton. Syst.*, vol. 62, no. 5, pp. 675–684, 2014.
- [37] H. Zhao and R. Shibasaki, "Reconstructing a textured CAD model of an urban environment using vehicle-borne laser range scanners and line cameras," *Mach. Vis. Appl.*, vol. 14, no. 1, pp. 35–41, 2003.
- [38] J. Serra and P. Soille, *Mathematical Morphology and Its Applications to Image Processing*, vol. 2. Fontainebleau, France: Springer, 2012.
- [39] H. Kaartinen, J. Hyyppä, A. Kukko, A. Jaakkola, and H. Hyyppä, "Benchmarking the performance of mobile laser scanning systems using a permanent test field," *Sensors*, vol. 12, no. 9, pp. 12814–12835, 2012. [Online]. Available: <http://www.mdpi.com/1424-8220/12/9/12814>
- [40] D. Borrmann, J. Elseberg, K. Lingemann, A. Nüchte, and J. Hertzberg, "The efficient extension of globally consistent scan matching to 6 DoF," in *Proc. 4th Int. Symp. 3D Data Process. Vis. Transmiss.*, Atlanta, GA, USA, Jun. 2008, pp. 29–36.



Ville V. Lehtola received the Ph.D. degree in computational statistical physics from Aalto University, Espoo, Finland, in 2010.

He was a Senior Researcher with the Finnish Geospatial Research Institute (FGI), Helsinki, Finland. He is currently an Assistant Professor of mobile and indoor mapping with the ITC Faculty, University of Twente, Enschede, The Netherlands. He has a background in algorithms, high-performance computing, and superclusters. His research interests include lasers and vision, robotics and automation, and positioning.



Matti Lehtomäki is currently pursuing the Ph.D. degree with Aalto University, Espoo, Finland.

He is currently a Research Scientist with Finnish Geospatial Research Institute, Helsinki, Finland. His research interests include computational methods, algorithm development, laser scanning, and computer vision.



Heikki Hyyti is currently pursuing the Ph.D. degree with Aalto University, Espoo, Finland.

He is currently a Research Scientist with Finnish Geospatial Research Institute, Helsinki, Finland. His research interests include sensor fusion methods in forestry and perception for autonomous forest machines.



Risto Kaijaluoto is currently pursuing the Ph.D. degree with Aalto University, Espoo, Finland.

He is currently a Research Scientist with the Finnish Geospatial Research Institute, Helsinki, Finland. His research interests include robotics, automatic point cloud processing, and deep learning.



Antero Kukko is currently a Research Professor with the Centre of Excellence in Laser Scanning Research, Finnish Geospatial Research Institute, Helsinki, Finland, and also an Adjunct Professor with the Department of Built Environment, Aalto University, Espoo, Finland. His research interests include the development and use of mobile laser scanning systems in multiple applications, from impact cratering and fluvial processes to precision forestry, urban mapping, and road conditions and assets.



Harri Kaartinen is currently a Research Professor with the Finnish Geospatial Research Institute, Helsinki, Finland, and a Professor of practice with the Department of Geography and Geology, Turku University, Turku, Finland. His background is in geodesy and remote sensing. His research interests include laser scanning systems, and their application and performance.



Juha Hyypä is currently a Distinguished Research Professor, the Director of the Centre of Excellence in Laser Scanning Research (Centre grant by Academy of Finland), and the Director of the Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute, Helsinki, Finland. His research interests include laser scanning systems, their performance and new applications, especially related to mobile and ubiquitous laser scanning systems including autonomous driving, and point cloud processing.