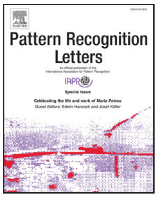
Pattern Recognition Letters 102 (2018) 58–66

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec



Acquiring qualified samples for RANSAC using geometrical constraints

Van-Hung Le[a,b,*], Hai Vu[a], Thuy Thi Nguyen[c], Thi-Lan Le[a], Thanh-Hai Tran[a]

[a] *International Research Institute MICA, HUST - CNRS/UMI-2954 -GRENOBLE INP, Hanoi University of Science and Technology, Vietnam*
[b] *Tan Trao University, Vietnam*
[c] *Faculty of Information Technology, VietNam National University of Agriculture, Vietnam*

ARTICLE INFO

Article history:
Received 10 May 2017
Available online 12 December 2017

Keywords:
RANSAC
RANSAC family
Sampling method
Geometrical analysis

ABSTRACT

Estimating parameters of a geometrical model from 3-D point cloud data is an important problem in computer vision. Random sample consensus (RANSAC) and its variations have been proposed for the estimation of the models parameters. However, RANSAC is computationally expensive and the problem is challenging when the measured 3-D data contain noise and outliers. This paper presents an efficient sampling technique for RANSAC, in which geometrical constraints are utilized for selecting good samples for a robust estimation. The constraints are based on two predefined criteria. First, the samples must ensure being consistent with the estimated model; second, the selected samples must satisfy explicit geometrical constraints of the interested objects. The proposed approach is wrapped as a robust estimator, named GCSAC (Geometrical Constraint SAmple Consensus), for estimating a cylindrical object from a 3-D point cloud. Extensive experiments on various data sets show that our method outperforms other robust estimators (e.g. MLESAC) tested in term of both precision of the estimated model and computational time. The implementations and evaluation datasets used in this paper are made publicly available.

1. Introduction

Estimating parameters of a primitive geographic shape such as plane, sphere, cylinder, cone, from 3-D point cloud data is a fundamental research topic in the fields of computer vision and robotics. The geometrical model of an interested object can be estimated using from two to seven geometrical parameters [17]. A Random Sample Consensus (RANSAC) [6] and its paradigm attempt to extract as good as possible shape parameters which are objected either heavy noise in the data or processing time constraints. For being more accurate, faster and more robust, the RANSAC family focuses on either a better hypothesis from random samples or higher accuracy of data satisfying the estimated model. In this paper, we propose to exploit geometrical constraints to obtain a qualified minimal sample set (MSS), i.e, good samples. This sample set can be used to generate better hypotheses, and as a result an estimated model is achievable. This study also demonstrates a real case where the proposed method is deployed for fitting cylindrical objects.

Originally, a RANSAC paradigm randomly draws 3-D points from an input data set without any prior assumption on the data. Theoretically, RANSAC must run a relatively large number of iterations to find an optimal solution before a stopping criterion is met. Chum et al. (in [3]) argued that “not all all-inlier samples are good”. They proposed the LO-RANSAC algorithm which is inspired by an idea that using an uncontaminated minimal sample is almost sufficiently or near perfect agreement with theoretical performance. Therefore, the most crucial question is how to select the uncontaminated or so-called qualified or *good samples* from a set of data points.

Motivated by this argument, we attempt to search for the qualified samples that could be better selected when geometrical constraints of the interested object(s) are used within a RANSAC-based algorithm. In particular, at each hypothesis in a framework of a RANSAC-based algorithm, a searching process aimed at finding *good samples* based on the constraints of an estimated model is implemented. To perform searches for good samples, we define two criteria: (1) The selected samples must ensure being consistent with the estimated model via a roughly inlier ratio evaluation; (2) The samples must satisfy explicit geometrical constraints of the interested objects (e.g., cylindrical constraints). The strategy of the proposed method first samples a minimal sample set that achieves a given inlier ratio among the samples and then adds samples that meet geometrical constraints derived from the primitive in concern. The seed generation is iterated until the convergence criterion is satisfied. In the other words, the key idea is guiding minimal sample set search using normal constraints of the geometric models (e.g., cylindrical objects). The constraints come from

[*] Corresponding author.
E-mail address: van-hung.le@mica.edu.vn (V.-H. Le).

https://doi.org/10.1016/j.patrec.2017.12.012
0167-8655/© 2017 Elsevier B.V. All rights reserved.

geometrical properties of a cylindrical model. We tackle that at each iteration, thanks to the good samples, thus, an optimal model (with the maximal inlier ratio) is highly expected. Consequently, the number of iterations can be adaptively updated according to the certain inlier rate, results in an earlier termination. The proposed method is different from LO-RANSAC because only minimal sample set is utilized for the model estimation. It is also separated from the USAC (A Universal Framework for Random Sample Consensus) [15], where the sample check stage and/or model check are implemented after each random sampling procedure. Our search for good samples is actively burned at the start of each iteration. To evaluate a hypothesis, we utilize the maximum likelihood as MLESAC algorithm [18]. For a termination condition, we adopt adaptive RANSAC [8] in which the number of iterations is adjusted whenever a better model is generated. Finally, the effectiveness of the proposed method is confirmed by a real case of fitting results of cylindrical objects in several datasets. The proposed algorithm obtains good performances in term of both precision of the estimated model and the processing time even on data with a low inlier ratio.

The main contributions of the paper are: (1) A new technique using geometrical constraints to search for good samples within a RANSAC-based algorithm. The proposed technique allows one to increase the quality of the sampled points and therefore the estimated model; (2) A demonstration of success of applying the proposed technique for a robust estimator for estimating a cylindrical object from a point cloud data. The implementations of the proposed method, method for evaluation and the collected data sets are made publicly available.

2. Related work

For a general introduction and performances of RANSAC family, readers can refer to good surveys in [2,16]. In the context of this research, we briefly survey related works which are categorized into two topics.

The first topic is efficient schemes on the selection of minimal subset of samples for RANSAC-based robust estimators, because the original RANSAC is very general with a straightforward implementation, it always requires considerable computational time. Many RANSAC variants have been proposed with further optimization for a minimal sample set (MSS) selection. Progressive Sample Consensus or PROSAC [3] orders quality of samples through a similarity function of two corresponding points in the context of finding good matching features between a pair of images. LO-RANSAC [4] and its fixed version LO$^+$-RANSAC [13] add local optimization steps within RANSAC to improve accuracy. To speed up the computation, adaptive RANSAC [8] probes the data via the consensus sets in order to adaptively determine the number of selected samples. The algorithm is immediately terminated when a smaller number of iterations has been obtained. With the proposed method, the *good samples* are expected to generate the best model as fast as possible. Therefore, the termination condition of the adaptive RANSAC [8] should be explored. Recently, USAC [15] introduces a new frame-work for a robust estimator. In the USAC frame-work, some strategies such as the sample check (Stage 1b) or the model check (Stage 2b), before and after model estimation, respectively, are similar to our ideas in this work. However, USAC does not really deploy an estimator for primitive shape(s) from a point cloud. A recent work [10] proposes to use geometric verification within a RANSAC frame-work. The authors deployed several check procedures such as sample relative configuration check based on the epipolar geometry. Rather than the “check” procedures, our strategies anticipate achieving the best model as soon as possible. Therefore, the number of iterations is significantly reduced thanks to the results of the search for good sample process.

For the second topic is cylindrical object estimation (or more general, fitting primitive shapes) from 3-D point clouds, readers can refer to a survey on feature-based techniques [1]. Some fitting techniques, for instance, multiscale super-quadric fitting in [5], Hough transform in [14], are commonly used. However, the robust estimators (e.g., RANSAC family [2]) are always preferred techniques. Original RANSAC [6] demonstrates itself robust performances in estimating cylinders from range data. In [19], normal vectors and curvature information are used for parameters' estimation and extraction of cylinders. The cylindrical objects are also interested in the analytic geometrical techniques. The authors in [7] formulate a cylinder using three parameters such as radius r, height h, and the axis the cylinder γ. [17] defines a cylinder through two samples and their normal vectors. The height of the cylinder normally is calculated in an additional step, e.g., determined by the maximal distance between two projected points in [7]. In this study, geometrical analysis of a cylinder in [17] is adopted for defining criteria of the qualified samples as well as for estimating parameters of the interested model from a 3-D point cloud.

3. Algorithms

3.1. General proposed frame-work

To estimate parameters of a 3D primitive shape, an original RANSAC paradigm, as shown in the top panel of Fig. 1, selects randomly an MSS from a point cloud and then model parameters are estimated and validated. The algorithm is often computationally infeasible and it is unnecessary to try every possible sample. Our proposed method (GCSAC - in the bottom panel of Fig. 1) is based on an original version of RANSAC, however it is different in three major aspects: (1) At each iteration, the minimal sample set is conducted when the random sampling procedure is performed, so that probing the consensus data is easily achievable. In other words, a low pre-defined inlier threshold can be deployed as a weak condition of the consistency. Then after only (few) random sampling iterations, the candidates of good samples could be achieved. (2) The minimal sample sets consist of qualified samples which ensure geometrical constraints of the interested object. (3) Termination condition of the adaptive RANSAC algorithm [8] is adopted so that the algorithm terminates as soon as the minimal sample set is found for which the number of iterations of current estimation is less than that which has already been achieved.

For evaluating and updating the best estimated model, different from an original RANSAC-based algorithm whose inlier ratio is the criteria for updating the best model, GCSAC adopts the Negative Log-Likelihood criteria of MLESAC algorithm [18]. The best estimated model is chosen when the minimum value of the log-likelihood (denoted as $-L$) is found. At each iteration, $-L$ is calculated by:

$$-L = -log\Big(\gamma_r\Big(\frac{1}{\sqrt{2\pi\sigma}}\Big)^n exp\big(-\frac{e^2}{2\sigma^2}\big) + \Big(1-\gamma_r\Big)\frac{1}{\nu}\Big) \qquad (1)$$

where σ is the standard deviation of the error; v is the parameter space which outliers are expected to fall in; γ_r is the mixing parameter that is estimated based on Expectation Maximization (EM) from a set of indicator variables η_i with $(i = 1, 2, \ldots, N)$.

To determine the termination criterion for the estimation algorithm, a well-known calculation for determining a number of sample selection K is:

$$K = \frac{log(1-p)}{log(1-w^s)} \qquad (2)$$

where p is the probability to find a model describing the data, s is the minimal number of samples needed to estimate a model, w is percentage of inliers in the point cloud.

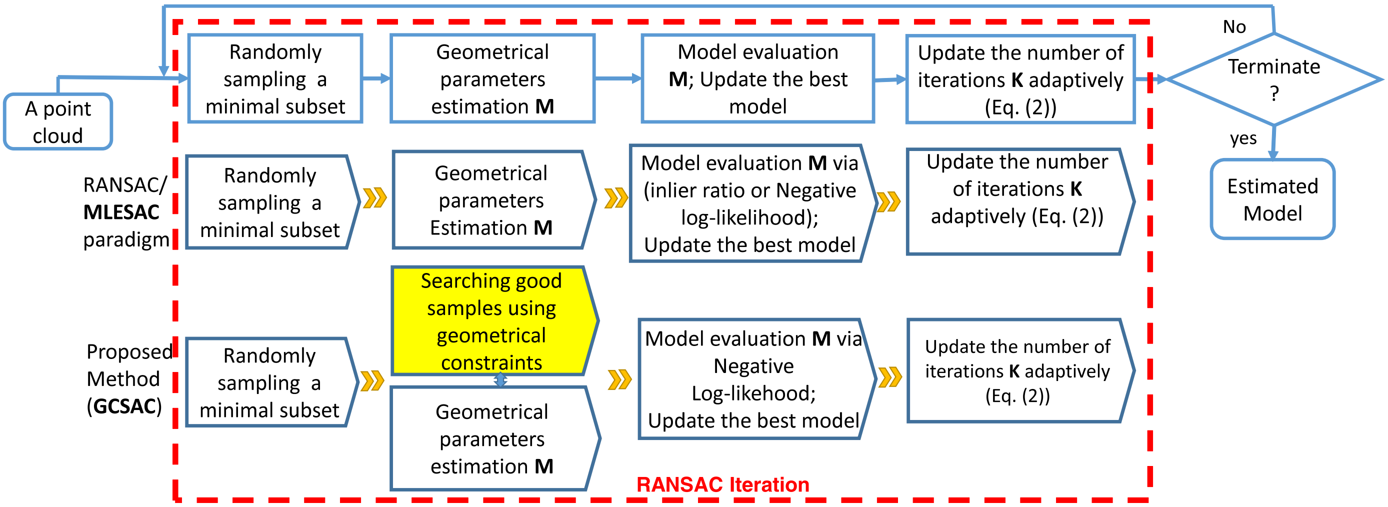

A point cloud
Randomly sampling a minimal subset
Geometrical parameters estimation **M**
Model evaluation **M**; Update the best model
Update the number of iterations **K** adaptively (Eq. (2))
Terminate ?
No
yes
Estimated Model
RANSAC/ **MLESAC** paradigm
Randomly sampling a minimal subset
Geometrical parameters Estimation **M**
Model evaluation **M** via (inlier ratio or Negative log-likelihood); Update the best model
Update the number of iterations **K** adaptively (Eq. (2))
Proposed Method (**GCSAC**)
Randomly sampling a minimal subset
Searching good samples using geometrical constraints
Geometrical parameters estimation **M**
Model evaluation **M** via Negative Log-likehood; Update the best model
Update the number of iterations **K** adaptively (Eq. (2))
RANSAC Iteration

Fig. 1. Overviews of the proposed method (GCSAC) and the conventional RANSACs and a fitting scheme. Top panel: The original RANSAC/MLESAC paradigms; Bottom panel: Our proposed method, GCSAC.

At each iteration, a sample point is specified as an inlier whose distance to the estimated model is smaller than a threshold *T*. In real datasets, this distance threshold *T* is usually chosen empirically. As explained in [8], when distribution of the data is a Gaussian with zero mean and standard deviation δ, the threshold distance *T* can be estimated by $T^2 = 5.99\delta^2$ for a probability of $\alpha = 0.95$ that the sample is an inlier (in case of the number of minimal sample set = 2).

While *p* is often set to a fixed value (e.g., $p = 0.99$ as a conservative probability), *K* therefore depends on *w* and *s*. At each iteration *k*, if the estimated model is better (more accurate), then all of the w can be estimated more confidently. As result of using the good samples, the proposed method obviously tends to estimate an optimal model at each iteration, instead of randomly finding out the best one as the original RANSAC. Therefore, at each iteration *k*, *w* is confidently estimated and *K* is updated by Eq. (2). It may occur an immediate termination when at a certain iteration, *w* determines a *K*, which is smaller than that already been performed.

3.2. Fitting 3-D cylindrical objects

For geometrical analyzing a cylinder object, we utilize the method proposed by [17]. A cylinder is determined by four parameters: A center point on the cylinder axis, denoted as $I(x_0, y_0, z_0)$; A vector of the main direction, denoted as γ_c; Radius R_a of the cylinder; Height of the cylinder, set to 1 by default.

The geometrical relationships of above parameters are shown in Fig. 2(a). A cylinder can be estimated from two points (p_1, p_2) (two blue-squared points) and their corresponding normal vectors ($\mathbf{n}_1, \mathbf{n}_2$) (marked by green and yellow line).

The normal vectors are computed using techniques proposed in [9]. At each point p_i, k-nearest neighbors kn of p_i are determined within a radius *r*. The computation of the vector of p_i is therefore reduced to the analysis of eigenvectors and eigenvalues of the covariance matrix *C* created from the kn of p_i, as given by:

$$C = \frac{1}{kn}\sum_{i=1}^{kn}(\mathbf{p}_i - \mathbf{p}_{av})(\mathbf{p}_i - \mathbf{p}_{av})^T, \quad C\mathbf{v}_j = \lambda_j\mathbf{v}_j, \quad j \in \{0, 1, 2\}; \tag{3}$$

where $\mathbf{p}_{av} = \frac{1}{kn}\sum_{i=1}^{kn} p_i$ represents the 3-D centroid of the nearest neighbors. λ_j is the j^{th} eigenvalue of the covariance matrix, and V_j is the j^{th} eigenvector found by Eq. (3).

Let γ_c be the main axis of the cylinder (red line) which is estimated by:

$$\gamma_c = \mathbf{n}_1 \times \mathbf{n}_2 \tag{4}$$

To specify a centroid point *I*, we project the two parametric lines $L1 = p_1 + t\mathbf{n}_1$ and $L2 = p_2 + t\mathbf{n}_2$ onto a plane specified by *PlaneY* (see Fig. 2(b)). The normal vector of this plane is estimated by a cross product of γ_c and $\mathbf{n}_1$ vectors ($\gamma_c \times \mathbf{n}_1$). The centroid point *I* is the intersection of *L*1 and *L*2 (see Fig. 2(c)). The radius R_a is set by the distance between *I* and p_1 in *PlaneY*. A result of the estimated cylinder from a point cloud is illustrated in Fig. 2(f). The height of the estimated cylinder is normalized to 1.

3.3. Cylindrical object fitting using geometrical constraints for quality of samples

The proposed frame-work is deployed for fitting a cylindrical object from a point cloud data. The corresponding normal vectors are already prepared in advance. The fitting procedure as shown in Fig. 3 consists of the corresponding steps of the proposed frame-work. The algorithm starts by roughly selecting initial good samples. We adopt the idea of adaptive RANSAC [8] to probe initial samples. At each iteration, we assume that the worst case estimate of w_t determines an initial sample. In the worse case (let's assume $w_t = 0.1$), a consensus set containing more than 10% of the data is found, that is at least the proportion of inliers. The initial inlier ratio w_t is set to 0.1 (or 10% inlier rate) is to easily find out a candidate of the estimated model. If a large value of w_t is set, a larger size of the consensus set will be found. As a consequent, more computational time before searching good samples is required. We tackle that the good samples selection helps to estimate the better model even with an initial probing the data via a small consensus set. Once the stack U_n^* of initial samples is conducted, we then search for good samples which satisfy the geometrical principles constraints of a 3-D cylindrical object as following. A calculating in the cost/score functions (e.g., our GCSAC adopted a Maximal Log-Likelihood function in MLESAC or counting the number of inlier points in original RANSAC algorithms) for the estimated cylindrical object (as specified in Section 3.2) is suffered from two free parameters. While the first parameter, threshold distance *T*, it is a fixed and pre-determined value, the second one, an angle restricts the deviation of a points normal from that of the estimated shape [17]. These conditions are investigated as following analyses. The

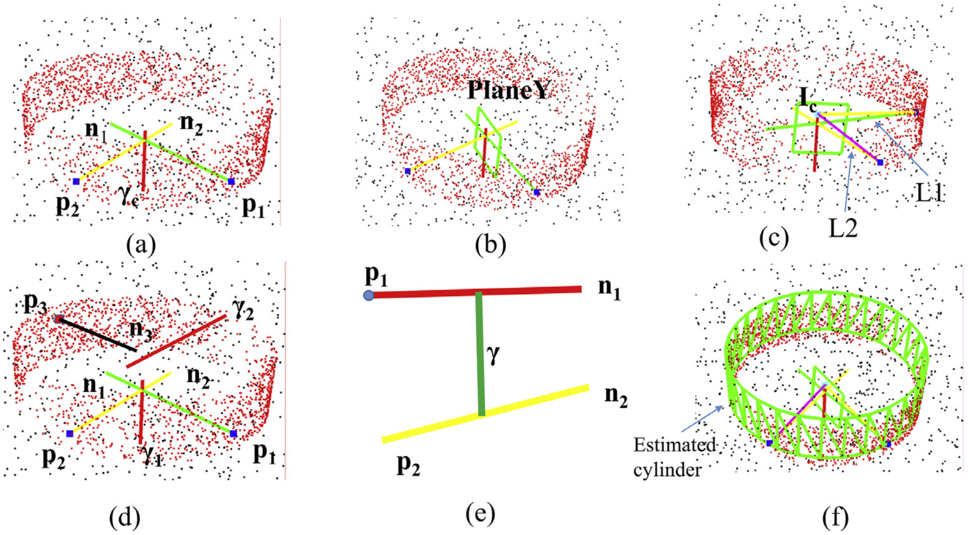


Fig. 2. Geometrical parameters of a cylindrical object. (a)–(c) Explanation of the geometrical analysis to estimate a cylindrical object. (d)–(e) Illustration of the geometrical constraints applied in GCSAC. (f) Result of the estimated cylinder from a point cloud. Blue points are outliers, red points are inliers. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

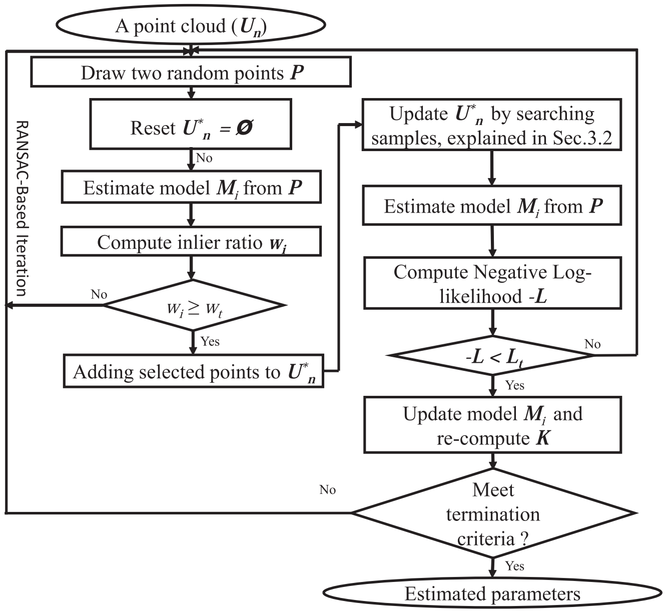

A point cloud (U_n)
Draw two random points P
Reset $U^*_n = \varnothing$
No
Estimate model M_i from P
Compute inlier ratio w_i
No
$w_i \geq w_t$
Yes
Adding selected points to U^*_n
RANSAC-Based Iteration
Update U^*_n by searching samples, explained in Sec.3.2
Estimate model M_i from P
Compute Negative Log-likelihood $-L$
$-L < L_t$
No
Yes
Update model M_i and re-compute K
No
Meet termination criteria ?
Yes
Estimated parameters

Fig. 3. A diagram of the cylindrical object estimation using GCSAC.

first condition is illustrated in Fig. 4. In this example, a pair of inlier p_1 and an outlier p_2 is selected. This is a common case when at a certain iteration, two samples are randomly drawn. As shown in Fig. 4(a), γ_c - the main axis of the cylindrical object, is incorrectly estimated. Consequently, the estimated cylinder is a wrong estimation (as shown in Fig. 4(b)). This case can be eliminated using the first condition when a threshold distance T specifying inliers is applied. For instance, a standard cost function of RANSAC counts the number of inliers. As a consequent, yielding is a very low in this case.

For the second parameter, let's denote random inlier samples as p_1, p_2 and p_3, as shown in Fig. 2(d). In case of drawing two random points p_1, p_3, obviously, the first criteria is quickly satisfied because both of these samples are inliers (w_i is larger than $w_t = 0.1$). However, as shown in Fig. 2(d), the direction of the axis γ_2 is totally different from ground-truth. Our second criterion (or search good samples) aims to update the initial samples (for example, p_3 should be updated by p_2). To obtain this, a centroid point I is a point on the main axis of the cylinder that the radius is the Euclidean distance from p_1 to I. We first built a plane π that is perpendicular to the plane $PlaneY$ and consists of n_1. Therefore its normal vector is $n_\pi = (n_{PlaneY} \times n_1)$ where n_{PlaneY} is the normal vector of $PlaneY$, as shown in Fig. 4(a). In the other words, n_1 is nearly perpendicular with n^*_2 where n^*_2 is the projection of n_2 onto the plane π. This observation leads to the criterion below:

$$c_p = \arg \min_{p_2 \in \{U_n \setminus p_1\}} \{\mathbf{n}_1 \cdot \mathbf{n}^*_2\} \tag{5}$$

If c_p is close to 0 then $\mathbf{n}_1$ and $\mathbf{n}^*_2$ are orthogonal (perpendicular) vectors. It is noticed that in the example as shown in Fig. 2(e), the projection of $\mathbf{n}_3$ onto plane π should be parallel with $\mathbf{n}_1$. Therefore the dot product $\mathbf{n}_1 \cdot \mathbf{n}^*_3$ is a large scalar value.

Observation is that the geometrical constraints can be applied as a pre-processing step. As results, a list of selected samples which ensure the constraints are extracted. We then apply a linear fitting algorithm for estimating a model. This scheme is quite similar to the åsimple strategy in [4]. However, they observed that this scheme does not lead to correct estimates. In our practical experiments, when the residual distribution is biased and the radius of the cylinder is large, i.e. outlier points, the estimated model parameters are no longer correct. The FILSAM works well only when the outliers do not have signification influence, that means the majority of inliers have bigger impacts on the least square's results. Theoretically, given N samples, maximal the number of selections is $\binom{N}{2} = N(N-1)/2$ for evaluating quality of whole samples. Therefore, total cost of FILSAM is significantly large.

4. Experimental results

The proposed methods are warped by C++ programs using a PCL 1.7 library on a PC with Core i5 processor and 8G RAM. The program runs sequentially as a single thread. The performances of the proposed algorithm are evaluated in three experiments for cylinder estimation.

4.1. Descriptions of the datasets for evaluation

The first (1st) dataset is purely artificial data which consists of six different subsets, denoted from dC_1 to dC_6. For each subset dC_i,

Table 1
The characteristics of the first dataset (synthesized data).

Data set	Characteristics of the generalized data			
	Height/Radius	Direction of main axis	Spatial distribution of inliers	Spread of outliers
dC_1, dC_4	1 /2	parallel with the z-axis	Around of a cylinder	$[-3, 3]$, $[-4, 4]$
dC_2, dC_5	1 /2	parallel with the y-axis	Around of a cylinder	$[-3, 3]$, $[-4, 4]$
dC_3, dC_6	1 /2	parallel with the y-axis	one half of a cylinder	$[-3, 3]$, $[-4, 4]$

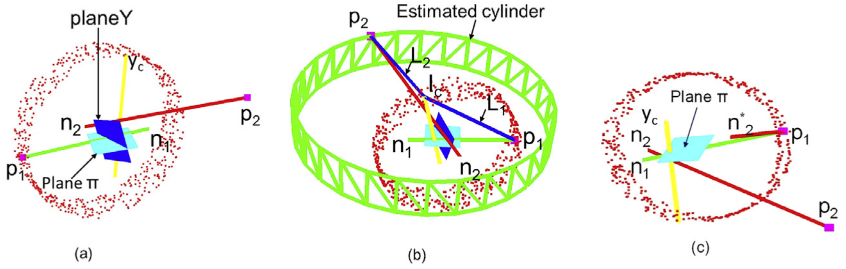


Fig. 4. (a) Setting geometrical parameters for estimating a cylindrical object from a point cloud as described in Section 3.2. (b) The estimated cylinder (green one) from an inlier p_1 and an outlier p_2. As shown, it is an incorrect estimation. (c) Normal vectors n_1 and n_2^* on the plane π are specified. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

inlier ratio is increased by a step of 5% from 10% to 80%. Therefore, there are fifteen point clouds. They are denoted from dS_1 to dS_{15}. A point cloud dS_i consists of three thousand points. An inlier data point (x_i, y_i, z_i) of dS_i lying on cylinders curved-surface is generated as follows: $x_i = \cos(\theta_i), z_i = \sin(\theta_i), y_i$ is randomly selected in [0, 1], θ_i is randomly selected from [0, 2π]. Outliers are generated randomly as a normal distribution. Characteristics of each set dC_i are described in Table 1. The major differences between a dC_i to dC_j could be the main axis's orientation, σ of the normal distribution for generating outlier/inlier data; or what is spatial distribution of inliers. Fig. 5 illustrates the synthesized data of dC_1, dC_2, dC_3 whose inlier ratio equals to 50%.

The second (2nd) dataset is collected from public available data sets which contains 300 objects belonging to 51 categories [11]. In this study, we collect only videos consisting of the cylindrical objects. Totally, this dataset consists of 8 coffee mugs, 14 food cans, 5 food cups, 6 soda cans.

The third (3rd) dataset is collected in real experiments in our lab-based environment where cylinders (e.g., a coffee cup) are located on the table. The main purpose of this setup is to deploy an aided-service which supports grasping queried objects for visually impaired people. In this set up, a Kinect is mounted on the chest of a person and he is moving around a table. This setup aims to detect a cylindrical object in real scenarios (e.g., to find a coffee cup in the kitchen). This dataset consists of four types of cups with different radii (3.75 cm, 3.5 cm, 3.25 cm, 3.0 cm). The cup is made of porcelain. Details of the experimental setup and so forth dataset are described in [12].

4.2. Evaluation measurements

Let notate a ground-truth cylinder $C_t(x_t, y_t, z_t, r_t, a_t)$ and the estimated one $C_e(x_e, y_e, z_e, r_e, a_e)$ where (x_t, y_t, z_t), (x_e, y_e, z_e) are the coordinates of the center points, r_t, r_e are the radii; a_t, a_e are the angles between the main axis and the normal vector of the table plane of C_t and C_e, respectively. To evaluate the performance of the proposed method, we use following measurements:

- The inlier ratio w of the best estimation is defined by:

$$w = \frac{\#\text{inliers}}{\#\text{number of samples}} \tag{6}$$

- Let denote the relative error of the estimated inlier ratio to be E_w. The smaller E_w is, the better the algorithm is.

$$E_w = \frac{|w - w_{gt}|}{w_{gt}} \times 100\% \tag{7}$$

where w_gt is the calculated by Eq. (6) using ground-truth inliers. This index is applied to the synthesized data (the first dataset) only because ground-truth inliers of real datasets are not precisely specified from original scenes.

- The total distance errors S_d is calculated by a summation of distances from any point p_j to the estimated cylinder C_e. S_d is defined by:

$$S_d = \sum_{j=1}^{N} d(p_j, C_e) \tag{8}$$

- The processing time t_p is measured in milliseconds (ms). The smaller t_p is the faster the algorithm is.

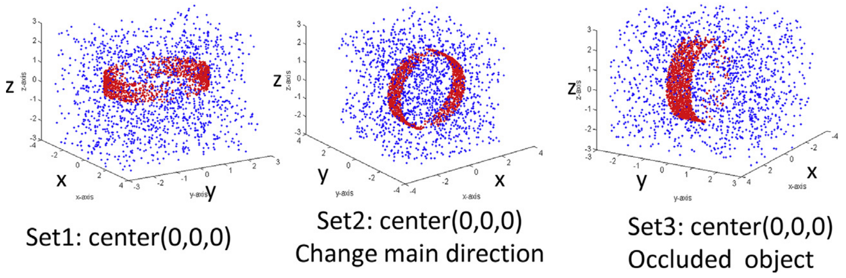


Fig. 5. Illustrations of dC_1, dC_2, dC_3 point clouds of the first dataset in case of 50% inlier ratio. The red/blue points are inliers/outliers, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Average results of the algorithms on synthesized dataset. The experiment on this data was repeated 50 times for statistical evaluation.

Dataset/ Measure	Method	E_w (%)	S_d	t_p (ms)	E_a ($deg.$)	E_r (%)
1st Dataset	**MLESAC**	42.3	1736.04	768.63	15.8	14.15
	FILSAM	36.54	1733.78	386.22	14.35	12.7
	GCSAC	**20.01**	**1702.09**	**58.71**	**8.76**	**3.83**

- The relative error of the estimated radius E_r is the difference of the estimated radius r_e and the ground truth one r_t.

$$E_r = \frac{|r_e - r_t|}{r_t} \times 100\% \tag{9}$$

- Let denote E_a infer a difference between the estimated angle a_e and truth one a_t. E_a is calculated by:

$$E_a = |a_e - a_t| \tag{10}$$

For comparative evaluations, the proposed method (named GCSAC) is compared with two different schemes, as shown in Fig. 1. One is original implementations of the MLESAC algorithm; and the second one is named FILSAM. FILSAM utilizes the geometrical constraints to filter out the *good samples*, before applying a linear fitting algorithm. For each evaluation, we denote the corresponding result of each method as following format: $name\_of\_measurement^{name\_of\_method}$. For instance, E_w^{MLESAC} is the inlier ratio error of the MLESAC algorithm. In the evaluations for all datasets, the smaller indexes (e.g., E_w, E_r, E_a) are, the better models are estimated.

For setting the parameters, we fixed thresholds of the estimators with $T = 0.05$ (or 5 cm), $w_t = 0.1$. T is a distance threshold to set a data point to be an inlier or outlier. For fair evaluations, T is set equally for all three fitting methods. Choosing an optimal T value is outside of the scope of this research. In case of the first dataset, the measurement error is known as a Gaussian with zero mean and standard deviation. According to [8] (Page 119), once the distribution of noise data is known in advance (e.g., the spread of the outliers is specified in Table 1), the threshold distance T is set by $T^2 = 5.99\delta^2$. Therefore, in the experimental evaluation, threshold distance T is equal 0.05. It ensures that an inlier will only be incorrectly rejected 5% of the time. For the real datasets (the second and third datasets), the threshold distance T is chosen empirically ($T = 0.05$). We observe in the real datasets that the cylindrical objects have common shapes and sizes. The threshold T is set 0.05 that is to appropriate with the size of the objects (e.g., the average radius of a coffee mug is about 3.75 cm). With threshold $T = 0.05$, it equals only 13% of a common radius.

As discussed in Section 3.3, w is a small value because this threshold is easier probing inliers from original data. In common cases, inlier ratios are usually higher than 10% (e.g., often from 50% to 60%). To filter out the good samples in the pre-processing step of the FILSAM scheme, we collect the good samples from a number of iterations. In practice, we set 10% from the maximally available iterations.

4.3. Experimental results

We first examine the performances of GCSAC in terms of the quality of the estimated model and computational time on the first dataset. Because of using the synthesised data, we can control both ground-truth inlier ratios and handle the optimal solution on each point cloud dS_i. As shown in Tables 2 and 3, all error measurements inferring quality of the estimated model using our method (e.g., E_w, E_r, E_a) are lower than that of MLESAC or FILSAM. Fig. 6 investigates deeper the stop terminations (or convergence) and residuals of GCSAC versus MLESAC. As shown in Fig. 6 left panel, comparing to MLESAC, the proposed technique converges faster and the stop criteria is met. Refer to Eq. (2), it may occur that at a certain iteration, w determines a K less than the number iterations for samples selection that have already performed, the number of iterations is reduced. However, the number of iterations of both method is the unequal decreasing with thedecreasing of inlier ratio because the sampling process is random and the estimation algorithm chooses a good sample when the number of iterations is small.

Fig. 7 demonstrates that GCSAC is able to estimate a cylindrical object even with a point cloud with an outlier rate of 90%. Intuitively, the estimated model by GCSAC is much better than that of MLESAC (right-panel, Fig. 7). In terms of quantitative evaluation, Table 2 shows measurements of E_r, E_a which also confirms the performances of GCSAC versus MLESAC and FILSAM. Particularly, in the case of the point clouds with low inlier ratios (e.g. from 10% to 45%), the estimated model using MLESAC is failed because these results are far from the corresponding ground-truth. The angle error is ($E_a = 15.8°$) for MLESAC, whereas GCSAC's error is only 8.76°. Generally, the quality of the estimated models using GCSAC and FILSAM is quite similar (as shown by E_a and E_r). However, the processing time is significantly different between the two methods. While FILSAM requires a large computational time ($t_p = 386$ ms), GCSAC requires only 58.71 ms.

For the second and the third datasets, they consist of natural scenes which are taken at different viewpoints and various types/sizes of the cylindrical objects. The effectiveness of the proposed search for good samples of GCSAC is illustrated by Fig. 8. While random MSS samples generate a failed candidate, as shown in Fig. 8(a) (whose current inlier rate is 0.03). These samples are updated using the searching process. Then a better model is estimated with the inlier rate of 0.19 (refer to the current best model in Fig. 8(c). Fig. 9 illustrates some example results using GCSAC on the second dataset. The evaluation results are reported in Table 3. All of the evaluation results show that GCSAC outperforms the MLESAC and FILSAM methods. Specially, the estimated inlier ratio w, and total distance error S_d confirm that fitting results are fairly good with GCSAC. While FILSAM is the worst case, the MLESAC is quite modest. These observations are also noted in [2,4].

We then deploy GCSAC in a real scenario where a visually impaired person queries a coffee cup in a kitchen [12]. The proposed method therefore can be used as a robust estimator for locating the coffee cup. Fig. 10(top panel) illustrates our experiment setup, and Fig. 10(bottom panel) shows corresponding results, where the coffee cups are marked in the point clouds. General performances of GCSAC versus others are summarized in Table 3. The inlier ratios in these evaluations are quite modest with more than 50% of entire data points. Therefore, the processing time of GCSAC is reduced compared to MLESAC and FILSAM. For each scenario, the computational time of GCSAC, MLESAC, FILSAM is 9.2 ms (or 110 fps), 9.9ms (or 100 fps) and 10.64 ms (or 94 fps), respectively. The accuracy of locating the interested object is verified using a chessboard which is prepared in the setup.

The first dataset (synthesized point clouds, as described in Table 1) for evaluation and the implementations of the proposed GCSAC methods are available at http://mica.edu.vn/perso/Le-Van-Hung/GCSAC/index.html.

Table 3
Experimental results on three datasets 2, 3. The experiments were repeated 20 times, then errors are averaged.

Dataset/Measure	Method	w (%)	S_d	t_p (ms)	E_a $(deg.)$	E_r (%)
2nd **Dataset (Coffee mug)**	**MLESAC**	9.94	3269.77	110.28		9.93
	FILSAM	11.23	3032.21	98.62		24.82
	GCSAC	**13.83**	**2807.40**	**33.44**		**7.00**
2nd **Dataset (Food can)**	**MLESAC**	19.05	1231.16	479.74		19.58
	FILSAM	19.32	1148.07	321.7		25.4
	GCSAC	**21.41**	**1015.38**	**119.46**		**13.48**
2nd **Dataset (Food cup)**	**MLESAC**	15.04	1211.91	101.61		21.89
	FILSAM	16.25	1159.3	201.83		20.14
	GCSAC	**18.8**	**1035.19**	**14.43**		**17.87**
2nd **Dataset (Soda can)**	**MLESAC**	13.54	1238.96	620.62		29.63
	FILSAM	16.25	1190.43	345.82		33.35
	GCSAC	**20.6**	**1004.27**	**16.25**		**27.7**
3rd **Dataset**	**MLESAC**	46.04	63.99	9.87	23.87	20.85
	FILSAM	47.59	54.93	10.64	22.6	17.81
	GCSAC	**48.96**	**49.26**	**9.20**	**20.50**	**15.47**

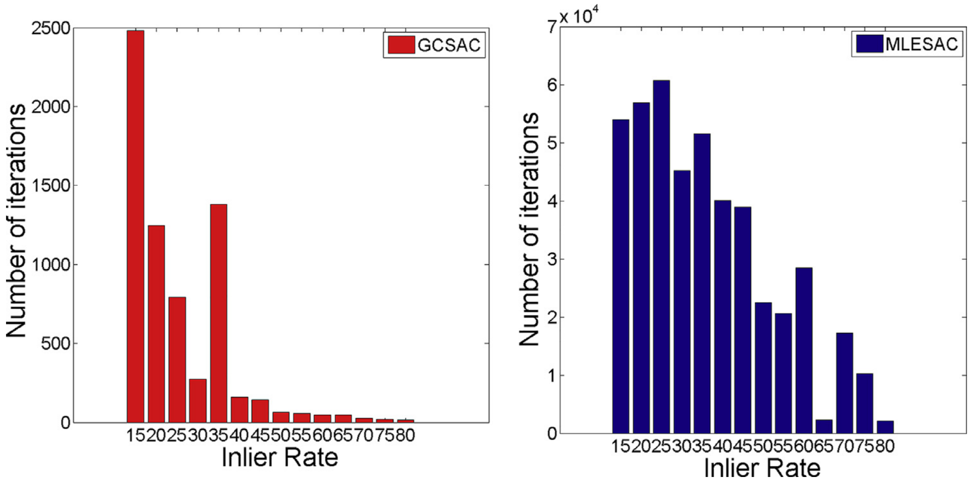


Fig. 6. The number of iterations of GCSAC (left side) and MLESAC (right side) for estimating a cylinder from point clouds in the first dataset. The results are given by averaging 50 run times.

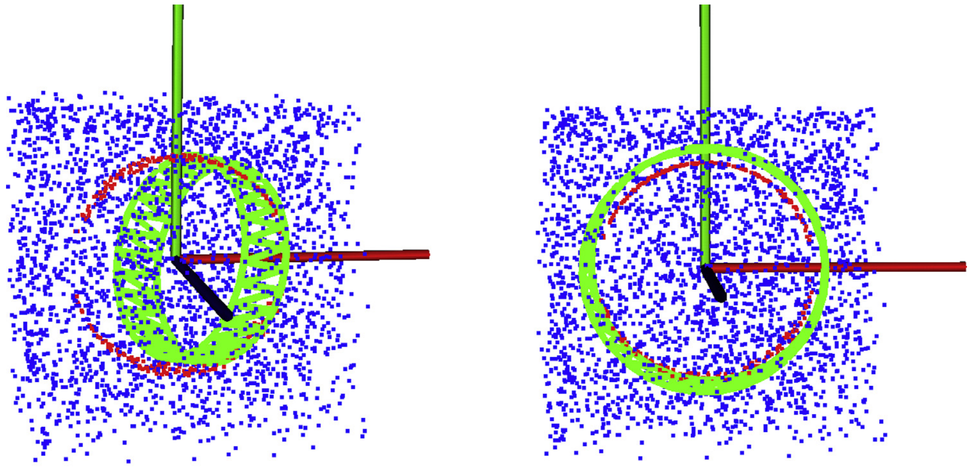


Fig. 7. The estimated cylinders (green points) in a point cloud of the first dataset with only 10% inlier. Right panel: GCSAC results. Left panel: MLESAC result. Red/blue points are inliers and outliers, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

5. Conclusions

In this paper, we have presented a new frame-work of a RANSAC-based algorithm for fitting a primitive geometrical model from a point cloud. The key idea of the proposed method is a new sampling process where good samples are qualified using geometrical constraints. We have shown that by adaptively updating number of sample selection, the proposed algorithm more quickly meets the termination conditions. We then applied the proposed method to fit cylinder objects. It has no limitation to extend the current work to other primitive shapes such as spheres, cones, or boxes. In the current version, we evaluated intensively the pro-

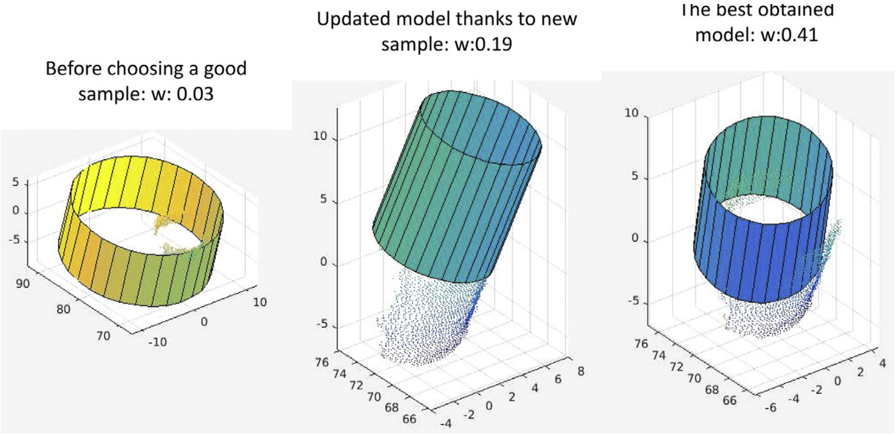


Fig. 8. An illustration of GCSAC's at a *k*th iteration to estimate a coffee mug in the second dataset. Left: the fitting result with a random MSS. Middle: the fitting result where the random samples are updated due to applying the geometrical constrains. Right: the current best model.

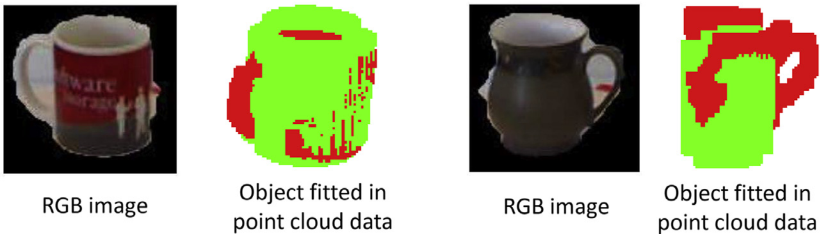

RGB image
Object fitted in point cloud data
RGB image
Object fitted in point cloud data

Fig. 9. Result of coffee mug fittings. Ground-truth objects are marked as red points; estimated ones are marked as green points. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

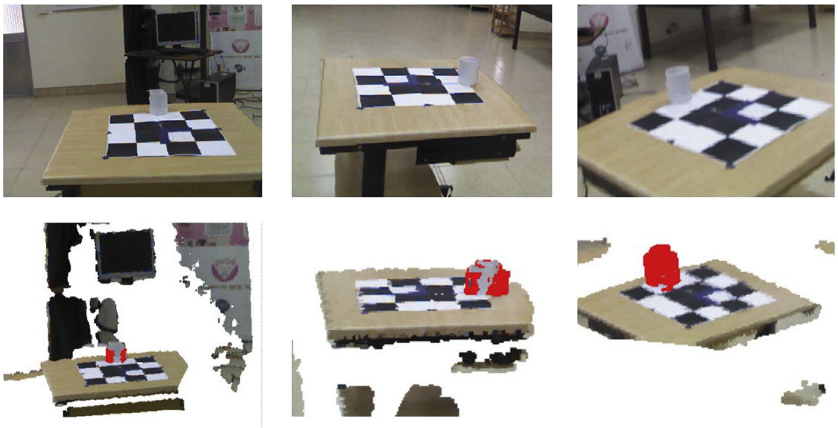


Fig. 10. The cup is detected by fitting a cylinder. Top row: The scenes in Dataset 3 captured at different view-points; Bottom row: The results of locating a coffee cup using GCSAC in corresponding point clouds.

posed method on different datasets. Our proposed method can be used to estimate a cylinder from point clouds which has low inlier ratios. In the case of real data, we also obtained a real-time performance for fitting the cylindrical-like objects such as coffee cup, food can, soda can (22 fps). These results suggest a feasible way to deploy aided-service to support visually impaired people grasping objects which are common requested in their daily life activities. In the future, we continue to validate these improvements on other geometrical structures, especially with a robust estimator that is independent of inlier distribution and error thresholds.

Acknowledgment

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.01-2017.12.

References

[1] K. Alhamzi, M. Elmogy, 3d object recognition based on image features : a survey 03 (03) (2014) 651–660.
[2] S. Choi, T. Kim, W. Yu, Performance evaluation of RANSAC family, in: BMVC, 2009, pp. 1–12.
[3] O. Chum, J. Matas, Matching with prosac progressive sample consensus, in: CVPR, 2005, pp. 220–226.
[4] O. Chum, J. Matas, J. Kittler, Locally optimized ransac., in: DAGM-Symposium, in: LNCS, 2781, Springer, 2003, pp. 236–243.
[5] K. Duncan, S. Sarkar, R. Alqasemi, R. Dubey, Multiscale superquadric fitting for efficient shape and pose recovery of unknown objects, ICRA, 2013.
[6] M.A. Fischler, R. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM 24 (6) (1981) 381–395.
[7] S. Garcia, Fitting primitive shapes to point clouds for robotic grasping, Master Thesis (2009).
[8] R.I. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, Second, Cambridge University Press, ISBN: 0521540518, 2004.

[9] S. Holz Dirk, R.B. Rusu, S. Behnke, Real-Time Plane Segmentation Using RGB-D Cameras, in: LNCS, 2011, pp. 306–317.

[10] M. Kohei, U. Yusuke, S. Shigeyuki, S. Sato, Geometric verification using semi-2d constraints for 3d object retrieval, in: ICPR, 2012., 2016, pp. 2339–2344.

[11] K. Lai, L. Bo, X. Ren, D. Fox, A large-scale hierarchical multi-view RGB-D object dataset, in: ICRA, 2011, pp. 1817–1824.

[12] V.H. Le, T.L. Le, H. Vu, T.T. Nguyen, T.H. Tran, T.C. Dao, H.Q. Nguyen, Geometry-based 3-d object fitting and localization in grasping aid for visually impaired, IEEE-ICCE, 2016.

[13] K. Lebeda, J. Matas, O. Chum, Fixing the locally optimized RANSAC, in: BMVC 2012., 2012, pp. 3–7.

[14] G. Osselman, B. Gorte, G. Sithole, T. Rabbani, Recognising structure in laser scanner point clouds, in: International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 2004, p. 3338.

[15] R. Raguram, O. Chum, M. Pollefeys, J. Matas, J.M. Frahm, Usac: a universal framework for random sample consensus, IEEE TPAMI 35 (8) (2013) 2022–2038.

[16] R. Raguram, J.-M. Frahm, M. Pollefeys, A comparative analysis of RANSAC techniques leading to adaptive real-time random sample consensus, in: ECCV, 2008, pp. 500–513.

[17] R. Schnabel, R. Wahl, R. Klein, Efficient RANSAC for point-cloud shape detection, Comput. Graphics Forum 26 (2) (2007) 214–226.

[18] P.H.S. Torr, A. Zisserman, Mlesac: a new robust estimator with application to estimating image geometry, Comput. Vis. Image Understand. 78 (1) (2000) 138–156.

[19] T. Trung-Thien, C. Van-Toan, L. Denis, Extraction of cylinders and estimation of their parameters from point clouds, Comput. Graphics 46 (2015) 345–357.