
Results on Range Image Segmentation for Service Robots

Stefan Gächter
Laboratoire de Systèmes Autonomes (LSA)
Ecole Polytechnique Fédérale de Lausanne (EPFL)
1015 Lausanne, Switzerland
stefan.gaechter@epfl.ch
http://asl.epfl.ch
2005-09-01
EPFL-LSA-2005-01 — Version 2.1.1

Abstract

This report presents an experimental evaluation of a plane extraction method using various line extraction algorithms. Four different algorithms are chosen, which are well known in mobile robotics and computer vision. Experiments are performed on two sets of 25 range images either obtained by simulation or acquired by a proprietary 3D laser scanner. The segmentation outcome of the simulated range images is measured in terms of an average segment classification ratio. Moreover, the speed of the method is measured to conclude on the suitability for service robot applications.

Contents

1	Introduction	2
2	Problem Definition	2
3	Selected Algorithms and Related Work	3
3.1	Line Extraction	4
3.2	Plane Extraction	5
4	Experimental Comparison	7
4.1	Experimental Setup	7
4.2	Validation of Simulated Range Images	9
4.3	Result of Simulated Range Image Segmentation	11
5	Conclusion	14
6	Version History	16

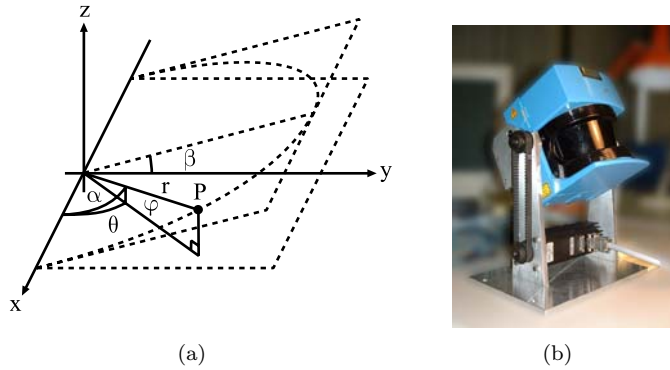


Figure 1: *Range Image Sensor and its Coordinate System.* On the left side is depicted the coordinate system of the range image in 3D. On the right side is depicted the 3D range image sensor - a pivoting 2D laser scanner. A measurement point P of the range image is specified in spherical coordinate system (r, θ, φ) . The range image is the composition of range scans at different elevation angles β . The range scans in 2D are specified in the polar coordinate system (r, α) .

1 Introduction

Context understanding is a key element when developing service robots capable to assist humans in every day life. Its potential and quality is enhanced when using range image sensors that provide the spatial perception as a unique modality. A range image explicitly represents the surface geometry of a given scene. Thus, range image segmentation is a precondition for context understanding.

Range image segmentation is a long standing issue - seminal work addressing this problem within the scope of context understanding has been done by Besl (1988). However, ready made solutions for range image segmentation are not available as it is the case for intensity images. One reason may be the diversity in range image acquisition and format making it difficult to develop a generic algorithm. This report approaches the problem based on a proprietary 3D laser scanner.

Even though integrated range image vision system for service robots has been proposed earlier, see Natonek (1998) and others, thorough performance analysis has been little addressed for segmentation of range images acquired in real indoor environments. This report may contribute to understand better the problems at issue when using current technologies in mobile robotics, where robustness, speed, and scalable algorithm are important aspects.

2 Problem Definition

A 3D range image describes the distance measurements from the sensor to surface points on objects in a scene. In the present case, the points of the range image are specified in spherical coordinate system (r, θ, φ) as depicted in Figure 1(a). It is common to assume that the noise on range image measurement r follows a Gaussian distribution with zero mean, variance σ_r^2 , and negligible angular uncertainty in θ and φ . In case of line extraction from 2D range scans, more sophisticated error models are discussed in Diosi and Kleeman (2003). The latter also addresses sys-

tematical errors. Even though more sophisticated error models have been used, the performance gain by their application remains still uncertain.

The problem is, given a noisy range image of unknown objects in an indoor scene, to segment the measured points into planar surface primitives that will be useful for context understanding in mobile robotic applications. Various algorithms attempt to solve this problem. One algorithm - scan line grouping - is discussed in the following sections, firstly, in a general context and, secondly, in the context of range images composed of multiple 2D range scans taken at different elevation angles. The algorithm partitions the 2D range scans into straight line segments and merges these into 3D planar surface segments. The straight line segments are described by the line equation in polar form:

$$x \cos(\alpha) + y \sin(\alpha) = r,$$

where $r > 0$ is the perpendicular distance from the origin to the line and $-\pi < \alpha \leq \pi$ is the angle between the x axis and r . The planar surface segments are described by the plane equation in spherical form:

$$x \cos(\varphi) \cos(\theta) + y \cos(\varphi) \sin(\theta) + z \sin(\varphi) = r,$$

where $r > 0$ is the perpendicular distance from the origin to the plane, $-\frac{\pi}{2} \leq \varphi \leq \frac{\pi}{2}$ is the angle between the xy plane and r , and $-\pi < \theta \leq \pi$ is the angle between the x axis and the projection of r onto the xy plane. It is possible to associate with each equation a covariance matrix specifying the uncertainties in the parameters.

3 Selected Algorithms and Related Work

According to Sagerer and Niemann (1997), a segmented object in the context of computer vision is some geometrical object defined by its parts, attributes, and relations. The parts are the main result of decomposing an object into simpler constituents. The attributes describe some physical or geometrical properties of the parts. The relations are not attached to the parts, but describe geometrical configurations, which the parts could constitute. Hence, the segmentation of the object into parts is of utter importance in context understanding. In case of range images, segmentation is a data-driven process using no application specific knowledge, but using generally applicable knowledge about surfaces. The here discussed algorithm, originally presented in Jiang and Bunke (1994), proceeds in two segmentation steps, firstly, points on a range scan are partitioned into straight line segments and, secondly, these line segments are merged into planar surfaces. Therefore, segmentation for line and plane extraction is defined as follows.

Definition 1 *The segmentation is the partitioning of a set into homogeneous subsets of maximum size. A subset is called a segment.*

The outcome of the segmentation depends on the homogeneity criterion, which can be in an implicit form or in a parametric form. The homogeneity criteria in implicit form are continuity (local criterion), curvature, and orientation, which are differential properties of the distance measurements. The homogeneity criteria in parametric form are algebraic equations (global criterion). The characteristic of the criterion -

Table 1: *Overview of Segmentation Methods.* The table states different segmentation methods classified by the homogeneity criterion, its property and application.

Method	Property	Application	Homogeneity Criterion
Edge Detection	local	local	continuity
Thresholding	local	local	curvature
Clustering	local	global	orientation, curvature, algebraic equation (local)
Region Growing	local or global	global	orientation, curvature, algebraic equation
Split-and-Merge	local or global	global	orientation, curvature, algebraic equation

local or global - is the constraint on the variety of the resulting segments. The segmentation method follows from the homogeneity criterion. Thus, possible methods are:

Edge Detection: The method is based on a local dissimilarity measure with local application. The set components for which the dissimilarity measure exceeds a certain threshold are regarded as edges.

Thresholding: The method is based on a local similarity measure with local application. One or several thresholds are applied on the similarity measure partitioning the set into distinct subsets.

Clustering: The method is based on a local similarity measure with global application. The set components are grouped according to the similarity measure.

Region Growing: The method is based on a local or global similarity measure with global application. Initially, the set components are searched for small homogeneous sets - seed regions. The subsets are created by merging the seed regions with the neighboring components having the same homogeneous properties.

Split and Merge: The method is based on a local or global similarity measure with global application. The components are split recursively into homogeneous subsets. The final subsets are created by merging subsets having the same homogeneous properties.

A more detailed analysis of different segmentation methods can be found in Maître (1994). A summary is given here in Table 1. Based on the presented segmentation methods, common line and plane extraction algorithms are discussed in the following sections.

3.1 Line Extraction

This section briefly presents four line extraction algorithms on 2D range scans. They belong to the split-and-merge, clustering, and edge detection segmentation methods. A more detailed discussion on line extraction algorithms and their implementation can be found in Nguyen et al. (2005). The selection is based on the popularity in both, mobile robotics and computer vision.

Recursive-Line-Fitting (RLF)	The algorithm is the first part of a split-and-merge segmentation method. Initially, a set s_0 consists of N_0 measurement points. A line is fitted to the set. The point P_s with maximum distance to the line is detected. If the distance exceeds an inlier threshold, then the set is split up at the point P_s into two subsets s_1 and s_2 of size N_1 and N_2 respectively. The splitting is repeated for each set s_i until the maximum distance is less than the threshold for all sets. Ambiguous edge points can be specially treated.
Iterative-End-Point-Fit (IEPF)	The algorithm is the first part of a split-and-merge segmentation method. The procedure is the same as the recursive-line-fitting algorithm, except that the fitted line is constructed simply by connecting the end points in each set. Ambiguous edge points can be specially treated.
Hough-Transform (HT)	The algorithm is a clustering segmentation method. Each of the N_0 measurement points in the initial set s_0 is transformed to the line parameter space. Measurement points that belong to the maximum crossing point in the parameter space which exceeds an accumulation threshold and that are smaller than an inlier threshold form the subset s_i . The subset s_i is removed from the initial set and the procedure is repeated until the maximum crossing point is less than the accumulation threshold.
Incremental Algorithm (IA)	The algorithm is an edge detection segmentation method. A subset s_i is created out of the initial set s_0 consisting of two measurement points in series. The line parameters for the subset are computed. The next point in series of the initial set is added to the subset, if the recomputed line parameters satisfy the line conditions. Otherwise, the procedure is repeated until all points are assigned to a subset. The incremental process can be speeded up by adding few points in a series instead of one.

The common total-least-square method is used for line fitting. Further, a simple clustering algorithm is used for filtering largely noisy points and coarsely dividing a range scan into contiguous groups. The range scan is split up where big jumps in radial differences of consecutive points occur. Moreover, groups with too few numbers of points are removed.

3.2 Plane Extraction

A large number of works are devoted to the range image segmentation problem, but unlike in the previous section, only one plane extraction algorithm is presented here. A thorough review of the latest algorithms is missing in the literature. However, the evaluation done in Hoover et al. (1996) and Jiang et al. (2000a) is still of actuality. Furthermore, a detailed description of various range image segmentation methods can be found in Jiang and Bunke (1997).

The algorithm under consideration is based on region growing where the primitives are straight line segments instead of individual measurement points. This algorithm, presented in Jiang and Bunke (1994), gained some popularity in the mobile robotics field Natonek (1998), Leger (1999) due to its simplicity and speed. Similar algorithms for planar surfaces have been developed Haindl and Žid (1997) or have been extended to curved surfaces Jiang et al. (2000b), Khalifa et al. (2003), Haindl and Žid (1998).

Scan-Line-Grouping (SGL)	The algorithm assumes that all measurement points on a straight 3D line segment belong to the same planar surface. Therefore, each range scan is
-----------------------------	--

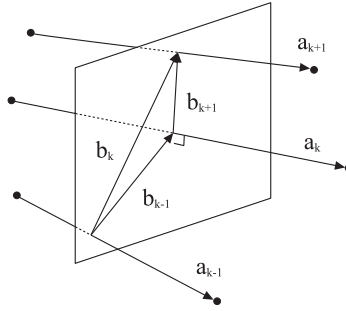


Figure 2: *Directional and Normal Vectors*. The directional and normal vectors, a_i and b_i respectively, for a triplet of line segments s_i used to compute the optimality measure.

divided into straight line segments and their neighborhood relationship is established. Out of these segments, potential seed regions consisting of three neighboring line segments are created and valued by an optimality criterion in the range $[0, 1]$, where 0 indicates the worst and 1 the best possible seed region. A seed region is an initial subset that originates a plane. The plane parameters for a subset are computed by least-square fit. A neighboring line segment is added to the subset, if the distances between its two end points and the plane are within a threshold. The procedure is repeated until no more neighboring line segments can be added, at which time a new subset is started using the next best available seed region. The region growing is repeated until no seed region remains.

In the current implementation, the algorithm is modified to take into account the characteristics of the range sensor. The differences are briefly discussed in the following.

Neighborhood Relationship: The neighborhood is defined by the scan angle α and the elevation angle β . A line segment s_k at elevation angle β_k is delimited by the end points (r_k^1, α_k^1) and (r_k^2, α_k^2) . Each line segment has one left neighbor (except the first segment) and one right neighbor (except the last segment). A line segment s_{k+1} at elevation angle β_{k+1} is a neighbor of s_k , if $[\alpha_k^1, \alpha_k^2] \cap [\alpha_{k+1}^1, \alpha_{k+1}^2] \neq \emptyset$ holds.

Optimality Measure: The seed region consists of three neighboring line segments s_i , $i = \{k-1, k, k+1\}$ of minimal length. The optimality criterion for each seed region is computed using the directional and perpendicular vectors, a_i and b_i respectively. The directional vectors of the line segments are the difference between the end points. The perpendicular vectors are the differences between the mid-point of segment s_k and the intersection points of the segments s_{k-1} and s_{k+1} with the plane defined by the normal a_k , see Figure 2. Then, the optimality measure is defined as

$$J = \frac{1}{6} \left(\sum_{i \neq j} \frac{|a_i \cdot a_j|}{|a_i||a_j|} + \sum_{i \neq j} \frac{|b_i \cdot b_j|}{|b_i||b_j|} \right),$$

which falls into the interval $[0, 1]$.

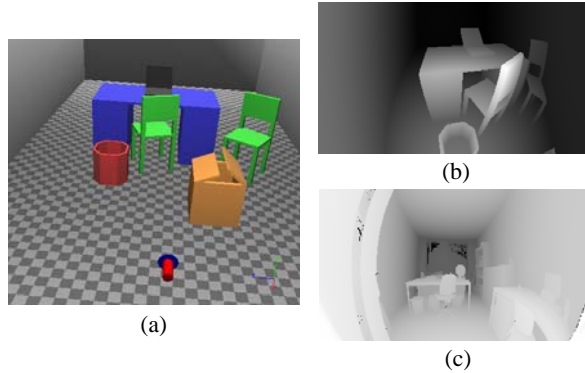


Figure 3: *Simulated and Real Range Images*. In (a) is depicted the virtual office scene from where the simulated range images have been taken. The small cylinder in the forefront indicates the sensor. The real office scene from where the real range images have been taken is not depicted here. The measurement points in spherical coordinate system are projected on a 2D view plane, where the range values are represented proportionally by gray levels. The projections of the simulated and real range image are depicted in (b) and (c) respectively. The field of view for both images is $90^\circ \times 180^\circ$, or 201×361 points, in vertical and horizontal direction.

Moreover, the current implementation is without pre- and post-treatment.

4 Experimental Comparison

4.1 Experimental Setup

The performance of the range image segmentation algorithm is evaluated as described in Hoover et al. (1996). However, different sets of range images are used to account for the particularity of the range sensor. The range images used by Hoover et al. (1996) and others are acquired by sensors with a small field-of-view. The images have high density and are almost uniformly sampled on a grid. In contrast, the present sensor has a large field-of-view. The images have low density and are uniformly sampled in the angle, see Figure 3(b) and (c). The sets of used range images are based on simulation and real experiments. The ground truth was created for the simulated range images.

Range Image Sensor: The 3D range image sensor, see Figure 1(a), is a custom setup based on a 2D laser scanner, SICK LMS200. In the current configuration, the sensor has a maximum measurement range of 8m, a range resolution of 10mm, a systematic error range of $\pm 15\text{mm}$, and a statistical error standard deviation of 5mm. These values have been validated by Ye and Borenstein (2002) for most measurement conditions of varying reflectivity and incidence angle. The sensor has a scan angle of 180° with angular resolution of $\Delta\alpha = 0.5^\circ$. The 2D laser scanner is mounted on a pivoting support. The support is driven by a step motor via a belt transmission. A similar design has been used by Surmann et al. (2001) and others. In the current configuration, the sensor has a elevation angle of 90° with angular resolution of $\Delta\beta = 0.45^\circ$. The minimum elevation angle is $\beta_{min} = -45^\circ$. Hence, the sensor has a field of view $90^\circ \times 180^\circ$ in vertical and horizontal direction and a complete 3D range image consists of $201 \times 361 = 72'561$ measurement points.

The first test set consists of 25 range images of a typical office environment.¹ A pair of images with a different viewpoint each have been taken in 12 different rooms. The rooms are highly structured, but also exhibit large planar surfaces where unobstructed walls and floor are present. The reflectance of the objects varies from opaque to transparent.

The second test set consists of 25 range images of a virtual office environment. Each image has been taken from a different viewpoint. The environment consists of a table, two chairs, a notebook, a dust bin, and a box, see Figure 3(a). The sensor model has the specification as stated above and has been implemented in Webots, a mobile robot simulation software developed by Cyberbotics Ltd..

The ground truth was created for each image in the second test set. The segmentation was done in a semi-automatic manner. A triangulated model of the virtual office environment was designed. Each measurement point in a range image was labeled according the circumscribing triangle. The labeled points were merged manually for triangles constituting the same planar surface, thereby object occlusion has been taken into account and surfaces have been broken up whenever necessary.

Performance Metrics: The performance is measured by comparing the segmentation outcome for simulated range images with the ground truth as described in Hoover et al. (1996). Five types of region classification are considered: correct detection, over-segmentation, under-segmentation, missed, and noise. Over-segmentation results in multiple detections of a single surface. Under-segmentation results in insufficient separation of multiple surfaces. A missed classification is used when the segmentation algorithm fails to find a surface, which appears in the image (false negative). A noise classification is used when the segmentation algorithms find a surface, which does not appear in the image (false positive). The formulas for deciding classification are based upon the classification threshold T , where $50\% < T \leq 100\%$. The classification threshold measures the congruency between segmented surface and ground truth. The metrics defining each classification are given in Hoover et al. (1996). An additional metric describing the accuracy of the recovered geometry is computed, the mean and standard deviation of all the dihedral angles between correct detected regions of the segmented range image and ground truth.

The parameter values for the segmentation algorithm are chosen according to the sensor and environment. The parameters are divided into two types: common parameters and algorithm specific parameters. Common parameters are those shared by all algorithms and for all test sets. The values are:

- Minimum number of points per line segment: 9.
- Minimum physical length of a line segment: 10cm.
- Standard deviation of range measurement: 1.0cm.
- Maximum distance from a point to line that the point is considered inlier to the line - inlier threshold: 1.0cm.
- Maximum distance from a end point of a line segment to the approximated plane that the line segment is considered as part of plane - merging threshold: 1.5cm.

¹ The test sets are available upon request from the authors.

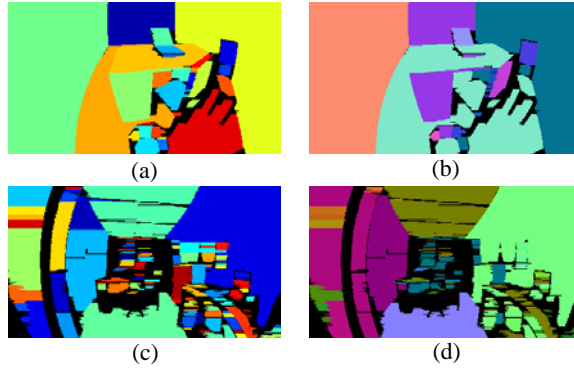


Figure 4: *Segmented Range Images*. In the upper row, images (a) and (b), are depicted the segmentation result of the simulated range image given in Figure 3(b). In the lower row, images (c) and (d), are depicted the segmentation result of the real range image given in Figure 3(c). The images in the left column show the segmented regions, where the images in the right column show the corresponding orientations.

The values of minimum number of points and length are chosen with respect to the maximum distance present in the environment, narrow offices with a surface in general not exceeding 25m^2 . The statistical error of the range measurement is chosen greater than the value given by the sensor manufacturer SICK AG to take into account imperfections of the real planar surfaces. The inlier threshold has the same value as the statistical range error resulting in an over-segmentation of the extracted lines. The merging threshold is chosen higher. Thus, the expected range image segmentation results should have a tendency for over-segmentation and reveal sufficiently details of the structured scene. The algorithm specific parameters are based on the results in Nguyen et al. (2005).

The simulated range images are corrupted with noise following a Gaussian distribution with zero mean and standard deviation $\sigma_r = 1.0\text{cm}$. Moreover, 25% of jump edges in the range scans are considered as mixed measurement points; neighboring measurement points with range difference greater than 50cm are replaced by their mean value.

The algorithms for line extraction are implemented in C and that for plane extraction in MATLAB. The experiments are performed on a notebook with PentiumM-1.8GHz and 1GB of memory. The values of computation time are measured with the MATLAB profiler.

In the following sections, firstly, the segmentation results for simulated and real range images are compared and, secondly, the segmentation results for simulated images are compared with the ground truth.

4.2 Validation of Simulated Range Images

Because no ground truth is available for the real images, the segmentation outcome for simulated range images is compared with the outcome for real range images based on the measures *number of seed regions*, *size of the segmented regions*, and the *distribution of the surface normals*. This gives a basic idea of how well the simulated images validate the real images.

The range images to illustrate the segmentation algorithm are depicted in Figure 3(b) and (c). The algorithm for the line extraction is the

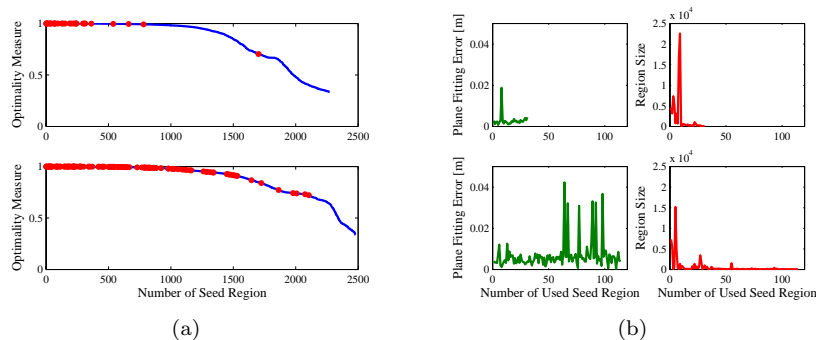


Figure 5: *Optimality Measure and Plane Fitting Error*. On the left side is depicted the optimality measure in function of the seed regions, where the blue line indicates the measure for all possible values and the red dots only the ones of the used seed regions. On the right side is depicted the fitting error for the final regions and their size in function of the used seed regions. The upper row corresponds to the range image depicted in Figure 3(b). The total number of seed regions is 2272, the number of used is 31. The lower row corresponds to the range image depicted in Figure 3(c). The total number of seed regions is 2480, the number of used is 114.

recursive-line-fitting method. This algorithm has the best performance as it is shown later. The outcome is depicted in Figure 4(a) and (c). The orientations of each segment are depicted color coded in Figure 4(b) and (d). The optimality measure of the corresponding seed regions and the plane fitting error of the final segments are depicted in Figure 5. As it can be seen, the number of initial seed regions for the simulated and real range image, 2272 and 2480 respectively, are similar, however, the number of final segments, 31 and 114 respectively, is considerably higher for the real image, because the real images is more structured than the simulated one.

Moreover, the plane fitting error is slightly correlated with region size as long as the optimality measure remains high. If the optimality measures drops, the plane fitting error increases even for small regions. In Figure 6 are depicted the region size histograms of the two sets. The region size is the number of measurement points in a segment, and the surface normal histogram for the two sets of 25 simulated and real range images. The average ratio of outliers, discarded measurement points, is 4.5% and 41.6% for the simulated and real set respectively, which correlates with the total number of regions of 634 and 3083 respectively. The simulated set tends to results in larger segments, while the real set in smaller ones, and the simulated set tends to results in segments oriented upward, where the real set in segments oriented downward and forward. It is obvious that the two sets have rather different characteristics for large and small segments. However, the region size histogram has a similar distribution between about 80 and 150 measurement points. It is assumed that at least for this band the correlation between simulated and real image set is strong enough to make a conclusion on the comparison with the ground truth in the following section.

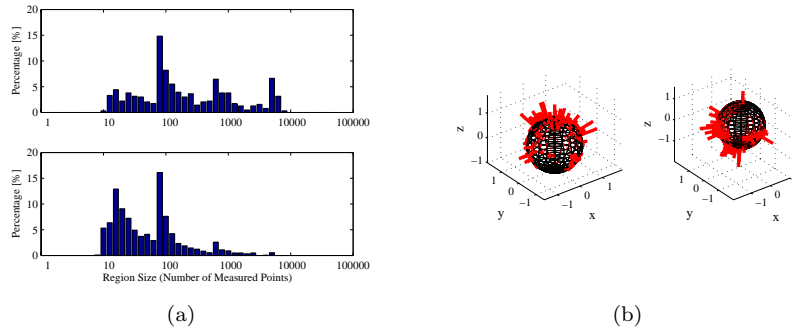


Figure 6: *Region Size and Surface Normal Histogram*. On the left side are depicted the region size histograms. The region size is the number of measurement points in a segment. On the right side is depicted the surface normal histogram in function of the orientation. The upper respectively left graph corresponds to the set of 25 simulated range images. The average ratio of outliers, discarded measurement points, is 4.5% and the total number of regions is 634. The lower respectively right graph corresponds to the set of 25 real range images. The average ratio of outliers, discarded measurement points, is 41.6% and the total number of regions is 3083. The region size histogram is normalized by the total number of regions, where the surface normal histogram is normalized by the maximum bin.

4.3 Result of Simulated Range Image Segmentation

The average classification rate of correct detected, over- and under-segmented, missed, and noise instances in function of the classification threshold for the simulated set are depicted in Figure 7 and 8. The average classification rate is the mean of the number of classified instances divided by the total number of instances in the ground truth. Thus, a perfect range image segmentation would result in a correct detection classification rate of 100%. The classification threshold measures the congruency. A threshold of 100% demands perfect congruence between regions in the ground truth and the corresponding ones in the range images. This is virtually impossible to achieve and, therefore, the number of missed regions increase to 100% with increasing threshold as can be seen in Figure 8(b).

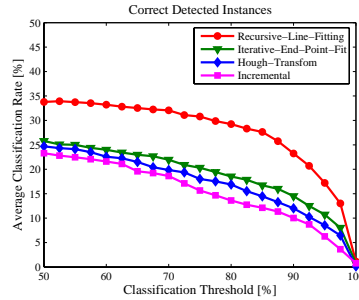


Figure 7: *Correct Detected Instances*. Depicted are the average classification rates of corrected detected instances in function of the classification threshold for four different line extraction algorithms: recursive-line-fitting (circle), iterative-end-point-fit (triangle), Hough-transform (diamond), and incremental algorithm (square). The average is taken over a set of 25 simulated range images.

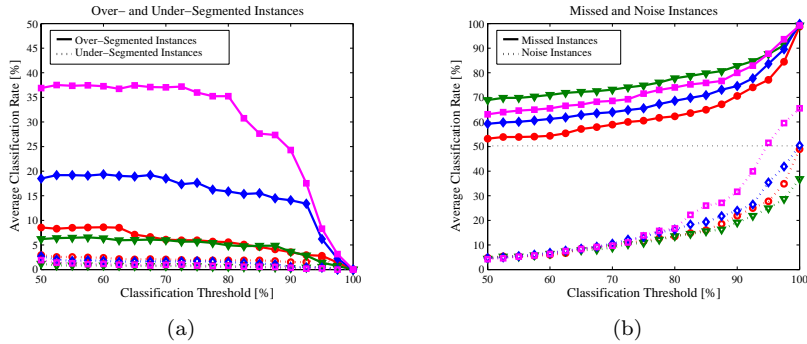


Figure 8: *Over-Segmented, Under-Segmented, Missed, and Noise Instances*. On the left side are depicted the average classification rates of over-segmented (full line) and under-segmented (dashed line) instances and on the right side are depicted the average classification rates of missed (full line) and noise (dashed line) instances in function of the classification threshold for four different line extraction algorithms: recursive-line-fitting (circle), iterative-end-point-fit (triangle), Hough-transform (diamond), and incremental algorithm (square). The average is taken over a set of 25 simulated range images.

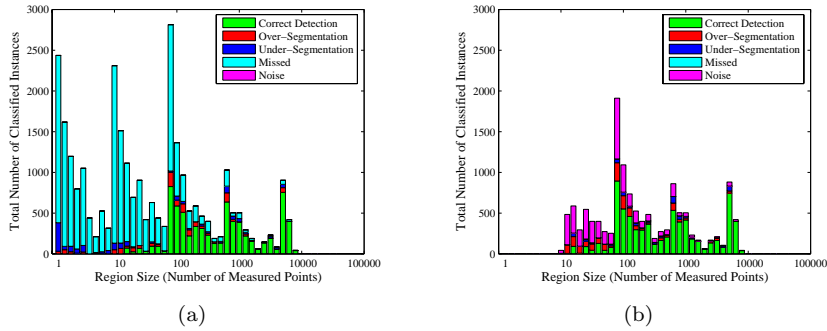


Figure 9: *Region Size Histograms of Ground Truth and Segmented Range Images.* On the left side is depicted the region size histogram for the planar surfaces in the ground truth and on the right side is depicted the region size histogram of the planar surfaces in the segmented range images. The region size is the number of measurement points in a segment. The size distribution for the different classified instances is given for correct detection (green), over-segmentation (red), under-segmentation (blue), missed (cyan), and noise (magenta) instances. The classification is based on a set of 25 simulated range images when using the recursive-line-fitting algorithm for line extraction.

As shown in the Figure 7 and 8, the segmentation method based on the recursive-line-fitting algorithm performs best. Generally, this algorithm has better performance in correct detection and low over- and under-segmentation over the whole classification threshold range. When the iterative-end-point-fit algorithm is used, the correct detection is poorer and the initial rate drops by 10%. This algorithm tends to shorten the line segments. The average outlier ratio is 11.1% compared with 4.3% for the previous algorithm, see Table 2. Thus, the influence of false edge detection is more unlikely, which is an advantage when merging the line segments and a slightly better performance in over- and under-segmentation results. In contrast, the algorithm misses most planar surfaces, because short line segments are discarded.

In general, the average classification ratio of missed instances is high given that a minimum segmentation length is imposed, which has similar or larger size than the smallest structure in the ground truth. It is the same for the minimum number of measurement points per line segment. This is clearly visible in the region size histogram of classified instances depicted in Figure 9, where most of the missed regions are of small size. The histograms are based on the result for the recursive-line-fitting method.

The performance when using the other algorithms differs mainly in the outcome of the over-segmentation, which is considerably higher. The line extraction based on Hough-transform can result in ambiguous line segments, i.e. line segments from the same range scan that intersect or overlap. In case of the incremental method, false edge detection is the reason. The edges in the range scan are in general not uniformly sampled in distance, which is a precondition for a good performance of the incremental algorithm. Therefore, edge points are added to the wrong line segment and alter its true pose. The result for both methods is over-segmentation.

The geometry accuracy measure is similar for all four cases. The mean and standard deviation values are about 2° at classification threshold of

Table 2: *Computation Time per Range Image*. The table states the computation time per range image at each processing step for different line extraction methods. Moreover, the average number of extracted segments and outlier ratio is given. The average is taken over a set of 25 simulated range images with size of $201 \times 361 = 72'561$ measurement points.

	RLF	IEPF	HT	IA
Average Outlier Ratio	4.3%	11.1%	10.5%	13.0%
Average Number of Extracted Segments	25.4	18.8	23.8	31.6
Line Extraction	0.62s	0.79s	22.04s	0.85s
Neighborhood Relation	0.23s	0.18s	0.20s	0.17s
Seed Regions	0.44s	0.26s	0.40s	0.22s
Plane Merging	3.86s	3.27s	3.77s	2.17s
Total	5.16s	4.50s	26.42s	3.41s

80%.

In terms of computation time, the segmentation method using the recursive-line-fitting algorithm performs best, see Table 2. The table states the computation time per range image for the main processing steps: line extraction, neighborhood relation compilation, seed region computation, and plane merging. The plane merging step is the most demanding. The average computation time for line extraction may vary - the Hough-transform based algorithm is by far the slowest - the average time for the other processing steps is similar among the different methods. The total average computation time for the best case is about 3.41s for the incremental based method. Over-segmentation and high outlier ratio may have a positive impact. The computation time, when using the set of real range images, roughly doubles for the line extraction step and roughly halves in the plane merging steps. The computation for the other steps remains similar. Thus, the total average computation time for real images does not change significantly, from 5.16s to 4.01s, when using the recursive-line-fitting algorithm.

5 Conclusion

This report has presented an experimental evaluation of a plane extraction method using four different line extraction algorithms. Overall, the range image segmentation based on the recursive-line-fitting algorithm has best performance. The range image segmentation based on the iterative-end-point-fit algorithm may perform better with real images, because the line segments depend less on the accuracy of the detected end points. In general, the quality of range image segmentation is strongly related to the performance of the line extraction method. The right choice may differ along with the application and implementation details.

The presented method - scan line grouping - takes advantage of the given data structure and provides reasonable results in short computation time. However, the method has also a weakness. The scanning of the environment with a large field of view together with uniform sampling in angle results in range scans not uniformly sampled in distance. Even if a clear edge is present in the environment, it may be not registered because of the flat incidence angle of the scanning plane. In such a case, all line extraction algorithms break down and the outcome of the range image

segmentation is poor. The method may be improved on the cost of computation time.

The report has also pointed out the difficulty to simulate range images. Even though indoor environments feature mainly planar surfaces, they are highly structured and the surface reflectance varies strongly. The resulting images are much more cluttered and noisy than can be achieved by a simple simulation. It needs better modeling to test the segmentation method soundly. Still, the simulation had the advantage of controllability and it was possible to conclude on the basic performance of each line extraction method.

The range image segmentation, as presented here, is useful for applications where time is critical and the field of view can be limited.

References

- Paul J. Besl. *Surfaces In Range Image Understanding*. Springer-Verlag Inc., New York, 1988. ISBN 0-387-96773-7.
- Cyberbotics Ltd. Switzerland, <http://www.cyberbotics.com/> (28.6.2005).
- Albert Diosi and Lindsay Kleeman. Uncertainty of line segments extracted from static SICK PLS laser. In *Australasian Conference on Robotics and Automation*, December 2003.
- Michal Haindl and Pavel Žid. Fast segmentation of range images. In Alberto Del Bimbo, editor, *Image Analysis and Processing*, Lecture Notes in Computer Science 1310, pages 295–302. Springer-Verlag, Berlin, 1997. ISBN 3-540-63507-6.
- Michal Haindl and Pavel Žid. Range image segmentation by curve grouping. In Anil K. Jain, Sveth Venkatesh, and Brian C. Lovell, editors, *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, volume 2, pages 985–987. IEEE Press, 1998. ISBN 0-8186-8512-3.
- Adam Hoover, Gillian Jean-Baptiste, Xiaoyi Jiang, Patrick J. Flynn, Horst Bunke, Dmitry B. Goldgof, Kevin Bowyer, David W. Eggert, Andrew Fitzgibbon, and Robert B. Fisher. An experimental comparison of range image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):673–689, 1996. ISSN 0162-8828.
- X. Jiang, K. Bowyer, Y. Morioka, S. Hiura, K. Sato, S. Inokuchi, M. Bock, C. Guerra, R.E. Loke, and J.M.H. du Buf. Some further results of experimental comparison of range image segmentation algorithms. In *Proceedings of the 15th International Conference on Pattern Recognition*, September 2000a.
- Xiaoyi Jiang and Horst Bunke. Fast segmentation of range images into planar regions by scan line grouping. *Machine Vision and Applications*, 7(2):115–122, 1994. ISSN 0932-8092.
- Xiaoyi Jiang and Horst Bunke. *Dreidimensionales Computersehen: Gewinnung und Analyse von Tiefenbildern*. Springer-Verlag, Berlin und Heidelberg, 1997. ISBN 3-540-60797-8.
- Xiaoyi Jiang, Horst Bunke, and Urs Meier. High-level feature based range

- image segmentation. *Image and Vision Computing*, 18(10):817–822, July 2000b.
- Inas Khalifa, Medhat Moussa, and Mohamed Kamel. Range image segmentation using local approximation of scan lines with application to CAD model acquisition. *Machine Vision and Applications*, 13(5–6): 263–274, March 2003. ISSN 0932-8092.
- Patrick C. Leger. Fast planar segmentation of range data for mobile robots. United States Patent, 5'978'504, November 1999.
- Gilbert Maître. *Segmentation et traitements préliminaires des images de profondeur*. PhD thesis, Université de Neuchâtel - Institut de Microtechnique, 1994.
- Emérico Natonek. Fast range image segmentation for servicing robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 406–411, May 1998.
- Viet Nguyen, Agostino Martinelli, Nicola Tomatis, and Roland Siegwart. A comparison of line extraction algorithms using 2D laser rangefinder for indoor mobile robotics. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2005.
- Gerhard Sagerer and Heinrich Niemann. *Semantic Networks for Understanding Scenes*, chapter 2, pages 43–75. Plenum Press, New York, 1997. ISBN 0-306-45704-0.
- SICK AG. Germany, <http://www.sick.com/> (1.7.2005).
- Hartmut Surmann, Kai Lingemann, Andreas Nüchter, and Joachim Hertzberg. Aufbau eines 3D-Laserscanners für autonome mobile Roboter. Technical Report GMD-Report 126, GMD - Forschungszentrum Informationstechnik, Germany, 2001.
- Cang Ye and Johann Borenstein. Characterization of a 2-D laser scanner for mobile robot obstacle negotiation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2512–2518, May 2002.

6 Version History

- 2005-07-01 Version 1.0 by Stefan Gächter Draft version of this document.
- 2005-07-08 Version 2.0 by Stefan Gächter Finalized version of this document.
- 2005-07-20 Version 2.1 by Stefan Gächter Reviewd version of this document.
- 2005-09-01 Version 2.1.1 by Stefan Gächter Spelling and diction corrected.