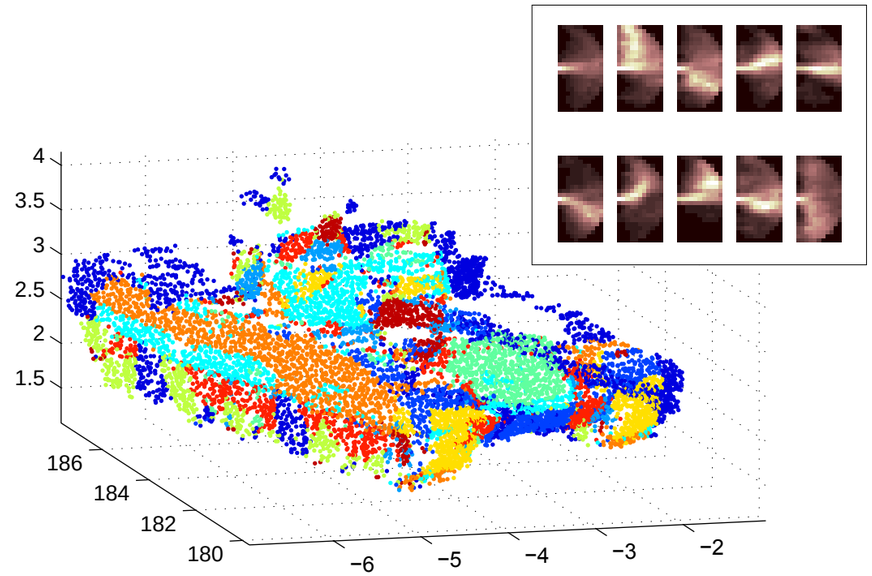
Methods for ground target detection and recognition in 3-D laser data

TOMAS CHEVALIER, PIERRE ANDERSSON, CHRISTINA GRÖNWALL, GUSTAV TOLT



FOI
Defence Research Agency
Sensor Technology
P.O. Box 1165
SE-581 11 Linköping

Phone: +46 13 37 80 00
Fax: +46 13 37 81 00

www.foi.se

FOI-R--2150--SE
ISSN 1650-1942

Scientific report
December 2006

Sensor Technology

Tomas Chevalier, Pierre Andersson, Christina Grönwall, Gustav Tolt

Methods for ground target detection and recognition in 3-D laser data

FOI-R--2150--SE
ISSN 1650-1942

Scientific report
December 2006

Sensor Technology

Issuing organization	Report number, ISRN	Report type
FOI – Swedish Defence Research Agency Sensor Technology P.O. Box 1165 SE-581 11 Linköping	FOI-R--2150--SE	Scientific report
	Research area code 4. C4ISTAR	
	Month year December 2006	**Project no.** E3080
	Sub area code 42 Above water Surveillance, Target acquistion and Reconnaissance	
	Sub area code 2	

Author/s (editor/s)	Project manager
Tomas Chevalier Pierre Andersson Christina Grönwall Gustav Tolt	Tomas Chevalier
	Approved by Lena Klasén, Head of Sensor Technology
	Sponsoring agency Swedish Armed Forces
	Scientifically and technically responsible Jörgen Ahlberg

Report title

Methods for ground target detection and recognition in 3-D laser data

Abstract

The signal processing of three-dimensional laser data is a relatively new field of research, especially the high resolution applications. The laser based 3-D sensors give contrast where passive 2-D sensors in general have small or no contrast, as for instance in darkness. They give the possibility to separate objects at different distances. As long as you have some light passing through a covering material, such as for most cloth, vegetation, smoke, and Venetian blinds, it is possible to get a geometric view of the hidden objects. This geometric object data can be used for absolute measurements and is therefore also a strong basis for recognition.

This report shows some of the studied methods to detect, segment and recognize objects in 3-D laser data and gives some examples. Only sensor systems placed on ground or close to ground in a fairly low angle of incidence are considered.

Keywords

3-D laser radar, point cloud, signal processing, recognition, detection

Further bibliographic information	Language English
ISSN 1650-1942	**Pages** 37 p.
	Price acc. to pricelist

Utgivare FOI - Totalförsvarets forskningsinstitut Sensorteknik Box 1165 581 11 Linköping	**Rapportnummer, ISRN** FOI-R--2150--SE	**Klassificering** Vetenskaplig rapport
	Forskningsområde 4. Ledning, informationsteknik och sensorer	
	Månad, år December 2006	**Projektnummer** E3080
	Delområde 42 Spaningssensorer	
	Delområde 2	
Författare/redaktör Tomas Chevalier Pierre Andersson Christina Grönwall Gustav Tolt	**Projektledare** Tomas Chevalier	
	Godkänd av Lena Klasén, Chef Sensorteknik	
	Uppdragsgivare/kundbeteckning FM	
	Tekniskt och/eller vetenskapligt ansvarig Jörgen Ahlberg	

Rapportens titel

Metoder för markmålsdetektion ach igenkänning i 3D-laserdata

Sammanfattning

Signalbehandling av tre-dimensionella laserdata är ett förhållandevis nytt forskningsområde, särskilt de högupplösta tillämpningarna. De laserbaserade 3D-sensorerna ger kontrast där passiva 2D-sensorer generellt har liten eller helt saknar kontrast, såsom i mörker. De ger möjlighet att separera objekt på olika avstånd. Så länge lite ljus passerar ett skylande material, som för de flesta tyger, vegetation, rök och persienner, ges möjlighet att få en geometrisk avbild av objektet. Denna geometriska avbild kan sedan användas för absoluta längdmätningar och ger därför ett bra underlag för igenkänning.

Den här rapporten visar några metoder att detektera, segmentera och känna igen objekt i 3D-laserradardata, samt ger några exempel. Vi använder oss av data från mätsystem som är placerade på marken eller nära marken, med andra ord en förhållandevis låg infallsvinkel.

Nyckelord

3D-laserradar, punktmoln, signalbehandling, igenkänning, detektion

Övriga bibliografiska uppgifter	**Språk** Engelska
ISSN 1650-1942	**Antal sidor:** 37 s.
Distribution enligt missiv	**Pris: Enligt prislista**

Contents

1 Introduction

The signal processing of three-dimensional laser data is a relatively new field of research, especially the high resolution applications. Laser based 3-D sensors give contrast where passive 2-D sensors in general have small or no contrast, as for instance in darkness. They give the possibility to:

- See at night, since the system uses its own light source (the laser)
- See far, as the light source can have high power
- Still be invisible/unrevealed, by using a non-visual wavelength
- Penetrate many sparse materials: vegetation, camouflage netting, Venetian blinds, dark windows, curtains etc.
- Visualize the measured object in 3-D, to support an operator
- Measure the shape and absolute size of an object, to support an automatic recognition method
- Discover camouflaged objects since the shape is harder to hide than the color

This report shows some of the studied methods to detect, segment, and recognize objects in 3-D laser data, and examples are given. Only sensor systems placed on ground or close to ground in a fairly low angle of incidence are considered.

The work reported here has been conducted at FOI, the Swedish Defence Research Agency, and has been mainly sponsored by the Swedish Armed Forces.

1.1 Purpose

The project “Target recognition with high resolution 3-D imaging laser radar” aims towards assessment of 3-D imaging laser radars for military applications [1]. As a necessary part of the project, an extensive study and evaluation of different methods for signal processing is being performed. The aim of the signal processing has been to exploit the third dimension and cover most parts of the chain from data acquisition and detection, via segmentation and classification to recognition. With this approach the goal has been to test and evaluate methods and the demands on real-time execution is excluded.

1.2 System aspects

One can think of different operation modes, where one mode would be to use the laser as a *robust night-capable surveillance* equipment, searching for anomalies in low resolution mode and sending alarms to other investigative systems. This would motivate the research and testing of registration methods as well as detection methods. The *registration* is used to merge datasets acquired at different times or with different field-of-views (with a bit of overlap). The *detection* methods are used to find anomalies in the geometry e.g. flat surfaces in the vegetation or high obstacles on the ground. A good way of finding abnormal objects is *change detection* where you search for local differences between two (or more) data sets covering to same scene. In some applications object *segmentation* are interesting, for instance to visualize the presence of different objects, or to extract the ground behind or below the trees.

Another mode of operation is to use the 3-D imaging laser radar with all of its *high resolution capability*, to investigate a local area if, for instance, a passive sensor generated an indication of something suspicious. In this mode, we have a small ROI (region-of-interest) and are more interested in segmentation, classification and recognition in this area, to either verify or deny the suspicion. Segmentation is used to identify data points originating from specific objects or materials, e.g. ground, trees, rocks, or vehicles. *Segmentation* is normally performed on a complete scene, and each data point belongs to a segment after the segmentation. Determining a type of class for only a sub group of data points is referred to as *classification* of those data points. The classes can be a tank, an APC, a rock etc. The next step would be to perform object *recognition* on a sub group of data points, where the result would be a T-80b tank, an MT-LB etc.

1.3 Summary

This report shows some of the studied methods to detect, segment and recognize objects in 3-D laser data and gives some examples. The environments being studied during this work are mainly natural environments with vegetation and trees. Urban environment will be considered, but are much more challenging. The local features used to separate targets from background in natural environments, for instance flat surfaces and corners, are very frequent in urban terrain, and new features must be considered. In this report we are only considering sensor systems placed on ground or close to the ground in a fairly low angle of incidence. One laser scanner system is presented in Section 2.

In Section 3 we show a method to register data (to align data sets geometrically), and iteratively improve the ***registration***. In large scale this is important for mapping, when the goal is to create 3-D maps of an area, by merging several data sets. In small scale, the purpose would rather be to merge several data sets of a single object to improve the chance of accurate recognition, or to present a more obvious visualization. The iterative method shows to be computationally demanding, but rather precise and robust, at least with a good initial estimate. A promising method for initialization is also shown.

The registration section is followed by Section 4, that investigates ***detection***. If the registration is used to align two data sets, acquired at different times or from different positions, there are methods to detect changes between the acquisitions. One method is described and illustrated and the result shows clearly where changes have occurred.

If only one data set is available, we instead have to use methods to detect abnormal objects in the terrain, i.e., *anomalies*, or features known to be frequent at the typical objects. One presented method, the spin-image representation, is used to detect features known to correspond to targets. We also investigate some methods to detect abnormal features in the terrain, such as flat surfaces, solid structures, and regions that both are higher than ground and has a deviant reflectance. Some of the methods show to work well in natural or clean environment. In cluttered environment, as urban terrain, further development is needed.

Supportive signal processing methods are presented in Section 5, ***segmentation and classification***. A method to estimate the ground level in a data set is presented, and found to produce quite accurate ground models. This can be used to estimate the potential drive route of a vehicle, and in many other ways as input to other methods described in this report. For instance we can analyze the shape of objects standing on the ground. Tall cylindrical objects are more likely natural objects than low square

objects. If the ground level and the zenith reference are known we can “gate” all data points between, for instance, 0.3 and 3 meters above the ground, and also remove the data points from trees. The rest of the hits are more easily examined. Finally a supportive method is used to classify spin-images in a more sophisticated way than ordinary classifiers.

The last processing method section, 6, contains some work concerning ***recognition*** of an object when its extents are known and the corresponding data points are separated from the background, i.e. the segmentation results are available. The methods cover rectangle fitting for automated target measurement, and recognition of complex targets divided into for instance body, tower, and barrel. A method to compare an object to a CAD-model library is also presented, together with some mathematical extensions to enhance the performance.

The methods in this report show a potential to fill the chain of needed signal processing operation from raw data to recognition. Since the goal has not been to develop a real-time system, effort has not been put on those aspects, but rather on performance issues. There are many parts in this work that need to be more thoroughly examined, for instance more objective performance estimations of some of the methods.

2 Sensor data

The sensor itself is not in focus in this report but sets the limits in some cases and has to be specified. Most data, except those used in Section 6.1, comes from the Optech ILRIS-3D laser scanner, described in Section 2.1. The other sensor has been a helicopter-mounted laser scanner, by TopEye AB [2], measuring vertically in a zigzag-pattern during flight.

2.1 Optech ILRIS-3D

The Optech ILRIS-3D is a 3-D imaging laser scanner acquiring absolute geometric coordinates as X-, Y-, and Z-coordinates as well as reflected intensity at the laser wavelength for each measurement point. The scanner pattern follows horizontally scanning lines filling the view from down and up. The coordinates exported from the acquisition system is locally defined with the Y-coordinate as range, the X-coordinate as the horizontal coordinate, and the Z-coordinate as the vertical coordinate. This is manually converted into a global coordinate system where *z* is elevation (according to the zenith reference) and the *x*-*y* plane is parallel to the ground. The typical density of the points for the measurements used in this report is 20-50 mm between each point. A typical dataset consists of 0.1 to 2.5 million points. The scanner has been mounted on a tripod standing on the ground as well as mounted in a sky lift 5-50 m above the ground. More information concerning the Optech ILRIS-3D laser scanner can be found in Table 1 and in [3].

Table 1. System parameters for the laser scanner Optech ILRIS-3D.

Maximum range	350 m (4 % reflectance) 800 m (20 % reflectance)
Measurement accuracy	Spatial values (X-Z) at 50 m ±7 mm Spatial values (X-Z) at 100 m ±10 mm Range values (Y) at 50 m ±10 mm Range values (Y) at 100 m ±10 mm
Measurement resolution	At 50 m: Spot size 15 mm, Spot distance <10 mm At 100 m: Spot size 20 mm, Spot distance <20 mm
Pulse repetitions frequency	2000 points per second
Laser wavelength	1.5 µm (NIR)
Beam divergence	0,2 mrad
Field of view	40° (±20°)
Eye safety	Laser Class 1, eyesafe

3 Registration

The more samples available from the target, the more information can be extracted. For example, increasing the number of samples could allow for target identification, rather than only detection. Sample limits for detection and identification have been studied in [4] and [5]. Objects in natural environments (e.g. forests) are often subject to occlusion from various structures (e.g. tree stems, leaves, and bushes), which significantly reduce the amount of samples on the target compared to what would be the case if the same target was placed in open terrain. This brings us to the idea of merging data acquired from different positions, thereby increasing the number of samples from occluded objects. However, a fundamental problem of using data acquired from different positions is that each data set is generally associated with its own coordinate system, related to the specific sensor that was used for collecting the data. Thus, in order to be able to merge several data sets, we have to express every point in each data set in a *global* coordinate system. This is commonly known as the *registration* problem. It is typically done by selecting the coordinate system of one data set as reference and then determining the rigid transformation of the other data set that best aligns the two sets.

In this section, tools for registration of laser point scatters are presented. It should be pointed out that in order to benefit from getting more samples, the registration process should not introduce large errors. However, a quantitative analysis of the registration performance is beyond the scope of this initial study.

3.1 Coarse registration by top-view 2-D histogram

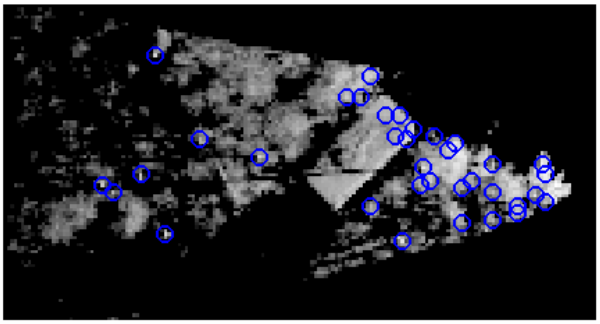
Most, if not all, techniques for accurate registration rely on a good initial pose estimate which is then typically iteratively refined, until the best alignment has been found. However, if the initial estimate is not good enough, the refinement step may actually degrade the result. We will refer to the process of obtaining an initial pose estimate as *coarse registration*. Generally, the key to successful registration is to determine features that are likely to appear in both data sets and that can hence be use for matching and to estimate the most likely transformation. A general and completely scene- and sensor-independent feature extraction and matching technique would indeed be appreciated, but we still expect that the problem could be solved by making use of some realistic assumptions about the scene and *a priori* information about the data acquisition process:

1. *The aspect angle of the sensor is roughly known, e.g. obtained with an inclinometer.*
   This makes it possible to pre-process the data by rotating the point sets so that the *z* axis points (approximately) upwards.
2. *There are enough hits from the ground to enable ground (bare earth) extraction.*
   This assumption is not limiting the applicability of the system significantly – if the ground is not visible, the targets, assumingly placed on the ground, are probably not either.

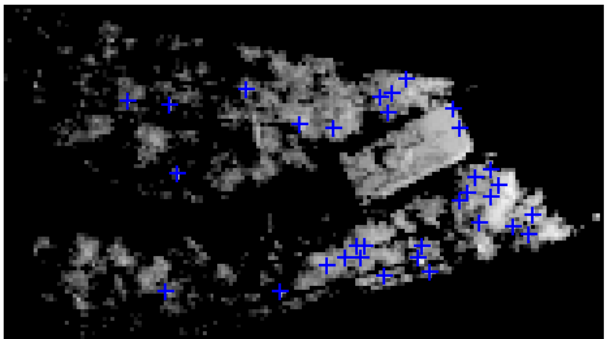
3. *There are approximately vertical structures placed on the ground.*
   This is a realistic assumption, for two reasons. First, it is vertical structures (e.g. trees) that are likely to occlude the target, which is one of the main motivations for studying this problem in the first place. Second, registration of a ground surface without vertical structures would be extremely uncertain (imagine matching two regions containing only perfectly flat ground).

For detecting vertical structures, a technique for detecting trees (Section 0) was used, with a slight modification – a predefined number objects were extracted, to avoid the problems associated with finding suitable thresholds.

(a)



(b)



(c)

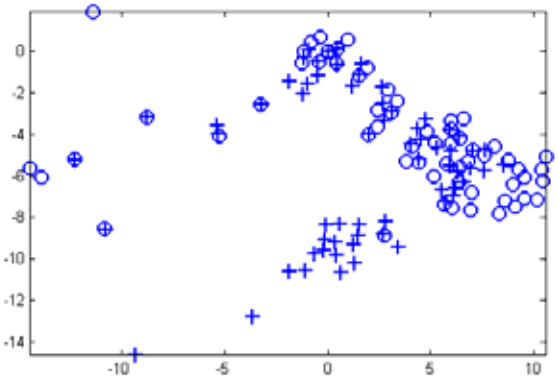


Figure 1. (a),(b) The histograms of two data sets with detected vertical structures. The histograms are displayed in a logarithmic scale, for visualization purposes. Note the abundance of features having no correspondence in the other data sets. (c) Result after point pattern matching, where the symbols corresponds to (a) and (b). Axes in meters.

Since object features are computed with data acquired from a certain position, they will vary from one point set to another. Thus, we cannot match the objects individually based on the features based on PCA (principal component analysis). Instead, we pursue a point pattern matching strategy, which determines the best 2-D transformation that best aligns the patterns [6]. We can generally tolerate a large number of patterns that do not have correspondences in the other dataset, e.g. from different fields of view or occlusion. A reason for this is that we use the actual 3-D coordinates and do not have to considerer scaling effects which could create a large number of possible poses.

Currently, the point pattern matching is performed by searching the space of possible translations and rotations. This space is reduced to a manageable size by always forcing two objects to be exactly overlaid. In order to express the degree to which two points r_i and r_j are close to each other, we use the "fuzzy" similarity, or proximity, function

$$s(r_i, r_j) = \max(1 - \frac{| r_i - r_j |}{T_d}, 0) ,$$

where T_d is the limit beyond which two points are considered to be false matches. The total degree of fit for a particular pose is defined as the sum of similarities between all points in the data set to be transformed and their respective closest neighbor in the reference point set. Note that this matching criterion is asymmetric, i.e. may produce different solutions depending on which point set one chooses as the reference set. However, since this criterion was adequate in our experiments, the task of finding a better matching criterion is left as a future issue. One method, evaluated for iteratively aligning of 3-D data to 2-D data from a hyperspectral camera, is based on entropy minimization and reported in [7].

After the alignment procedure described above, we still have to estimate the translation along the *z*-axis. This is estimated by computing the median of the differences between the two ground surfaces in the region where the point sets overlap.

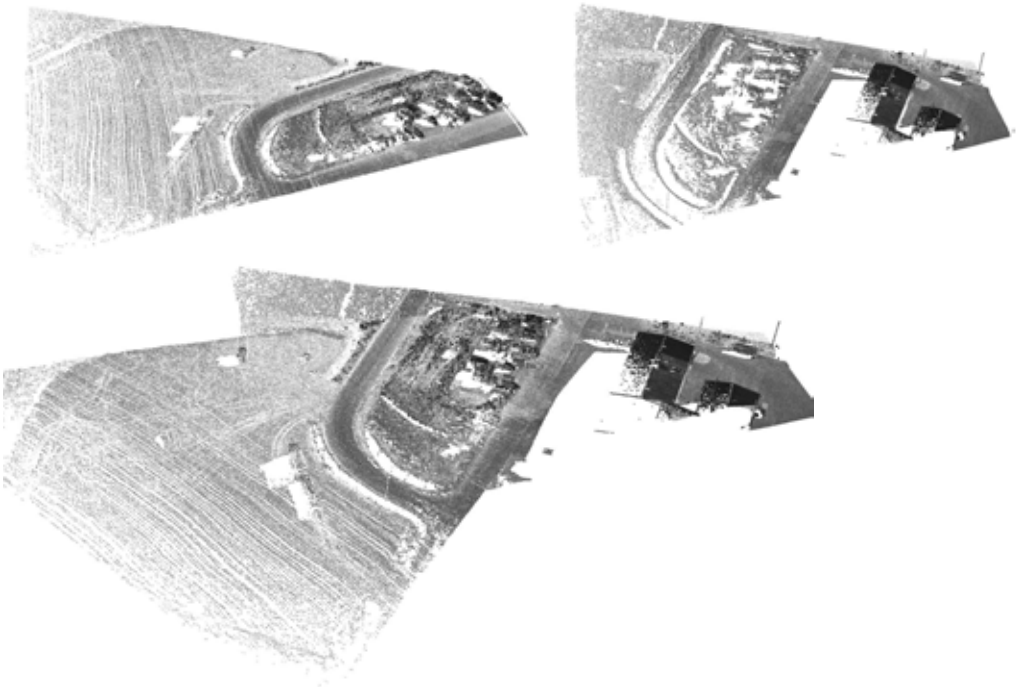


Figure 2. The figure illustrates that the 2-D coarse registration technique is capable of registration of data sets acquired from quite different positions.

3.2 *Iterative Closest Point with keypoint detection*

The Iterative Closest Point (ICP) algorithm, proposed in [8], has been widely used for fine 3-D data registration. Since then a number of improvements and variants of the original ICP algorithm have been suggested, see e.g. [9]. Common for most ICP-related work presented in the literature is that the algorithms are applied on close range data from static scenes containing rigid objects. It is thus interesting to study whether this technique can be used for natural scenes.

The first step is to pair each point in a point set ***P*** with its closest point in another set ***Q***. Let ***P'*** and ***Q'*** be two sets of corresponding points, such that q'_i are the point in ***Q*** closest to p'_i. The assumption is then that that the sought-after transformation [***R***,***T***] is the one that minimizes the least square error function

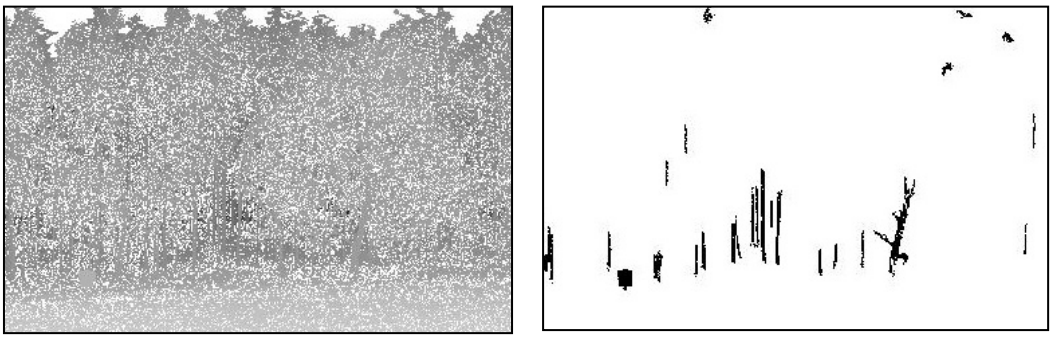
$$\Sigma^2 = \sum_i \| q'_i - Rp'_i - T \|^2, \qquad (1)$$

where ***R*** is the rotation matrix and ***T*** is the translational vector. The estimated transformation $[\hat{R}, \hat{T}]$ determined in the *motion estimation* step, i.e. by minimizing (1), is then applied to **P**. Another search for closest points is performed, after which a refined motion is estimated using the new points correspondences, and so on. This iterative procedure continues until the solution converges or until some other criterion is met.

The ICP algorithm is associated with some typical limitations. First, it has to be supplied with an initial pose estimate in order not to end up in a local minimum far from the correct solution. Still, there is no guarantee that the global minimum is reached. In fact, the presence of many points with no correspondence in the other point set may lead to very poor pose estimates.

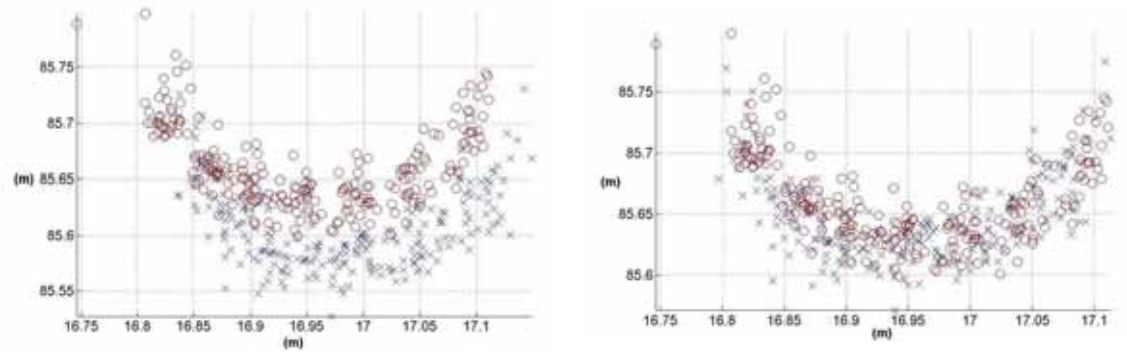
3.2.1 Keypoint detection

In natural, dynamic environments, e.g. scenes containing trees moving in the wind, we expect a great number of false correspondances if the entire point sets are used. Hence, it would be desirable to exclude points that are likely to have less correspondence in the other point set prior to the pose estimation. We assume that the selected points are those who lie on surfaces. In addition to increase the chances that the algorithm eventually converges to a good solution, the computational cost is dramatically reduced, when these surface points have been identified. Different keypoint extraction techniques, e.g. based on range variance filtering, filtering based on the normal direction and a connected component analysis technique, have been proposed and investigated [10]. See Figure 3 and Figure 4 for illustrations.



(a) (b)

Figure 3. (a) Laser intensity image of a forest scene. (b) Detected keypoints.



(a) (b)

Figure 4. An illustration of the typical effect of using keypoints in natural environments. Close-up of a tree trunk in the forest scene (Fig. 1) after ICP matching. (a) *Without* keypoints. (b) *With* keypoints. Note the better correspondence between the two data sets, due to the extraction of more reliable point sets through surface extraction.

4 Detection

This section describes methods to detect abnormal features in natural scenes as well as detecting features that are known to belong to certain objects, from e.g. a model library or data from real targets acquired in advance.

4.1 Local flatness estimation

In a natural forest environment, vehicle targets will typically represent the largest structured objects in laser radar data. Moreover, a structured object can be recognized by having surfaces that can be considered flat in local neighborhoods of sufficient size. We exploit this property for detecting targets, using PCA of data partitions.

Each data point set is partitioned into a number of non-overlapping cube shaped subvolumes (hereafter referred to as *cells*) of a suitable size chosen to match the size of approximately planar surface segments expected to be found on the target (e.g. 0.5 x 0.5 x 0.5 m^3). Each cell is then checked for flatness by measuring the degree to which data is distributed along a plane of arbitrary orientation.

The partitioning algorithm outputs a data structure where each element contains indices pointing to the affiliated data points of that cell, and also reference indices pointing to its neighboring cells, specifying the connectivity of partitions. A separate list of labels referencing the affiliation of each data point is also provided. The result is used for more efficient handling of large data sets in for example surface normal estimation and spin-image generation (see Section 4.3 for further explanation of these concepts). Here, it is instead used as a direct component of the surface estimation algorithm.

For a given cell C_k, we study two features: the point density feature, d_k, and the degree to which the points within the cell form a planar surface, s_k. These measures are defined as follows:

$$d_k = N_k \cdot R_k^2,$$

$$s_k = d_k \cdot \frac{\lambda_{k,2}}{\lambda_{k,3} + \tilde{\lambda}_3},$$

where N_k is the number of samples in C_k, R_k is the distance from the origin of the sensor to C_k, $\lambda_{k,i}$ is eigenvalue i given by PCA of the point coordinates in C_k with $\lambda_{k,1} \geq \lambda_{k,2} \geq \lambda_{k,3}$ and $\tilde{\lambda}_3$ is the median of $\{\lambda_{k,3}\}_k$, to take the statistics of the given data set into account.

In order to prevent cells with a low point count from obtaining a high degree of "surfaceness" by chance, the point density feature d_k in s_k adds weight to cells where a high point count is present to support a high surface score. A compensation factor R_k^2 corrects for the decline in expected point density that comes with increasing distance from the sensor. For the same reasons we also limit the analysis to be valid only in partitions containing more than five points.

The surface measure is calculated in an algorithm context containing ground estimation by watershed segmentation (see Section 5.1), and all data points below a ground threshold level are ruled out as targets, receiving a surface score of zero. In this way ground points that otherwise might receive a high surface score and constitute a source of false alarms are excluded from the analysis.

The algorithm was initially tested on ten data sets collected with the ILRIS-3D scanner; see Table 2 for a brief overview.

Table 2. Overview of analyzed data sets

ID	Scene	Targets	Description	Sensor	# points
1	Grass field and forest	MT-LB/pbv401 (2)	Targets in forest	Ground	869258
2	Tree line facing sensor 175-200 m	T-72 (1) Volvo V70 (1)	Targets in tree line	Ground	845698
3	away		Targets on field	Ground	2578576
4	Forest with road	M60 (2) T-72 (1)	Targets on road (1) and in forest (2)	Tower	1533136
5	Forest	Camouflage painted container	Spring season	Hilltop	1135561
6			Spring season		1136509
7			Spring season		1136980
8	(Reference panel on		Spring season		1136094
9	tripod was removed		Spring season		1135983
10	manually from data)		Summer season		976149

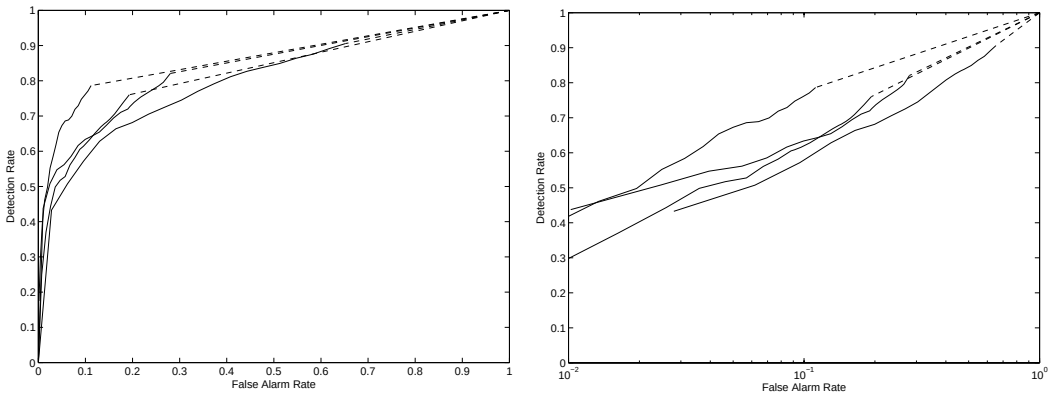


Figure 5. ROC curves for data sets 1-4 showing the detection rate plotted against the false alarm rate in linear scale (left) and logarithmic scale (right). The solid lines are interpolated between real data, while dashed lines are extrapolated between the last value and the unavoidable (1,1) - coordinate.

The partition size is set to 0.5 m, corresponding to the expected size of approximately flat surfaces on the targets. The ground threshold is set to 0.5 m. This will exclude most of the actual ground points while not falsely rejecting significant numbers of target parts. The amount of actual target points rejected by the ground estimator averaged approximately 16 % for data sets 1-4 and 9 % for data sets 5-10. In these early results in Figure 5, we present ROC curves based on individual point scores. We see that a fair amount of target points are still detected at surface measure levels where the false alarm rate is only a few percent, indicating some degree of partial separation between the two classes. Figure 6 shows ROC curves for data sets 5-10.

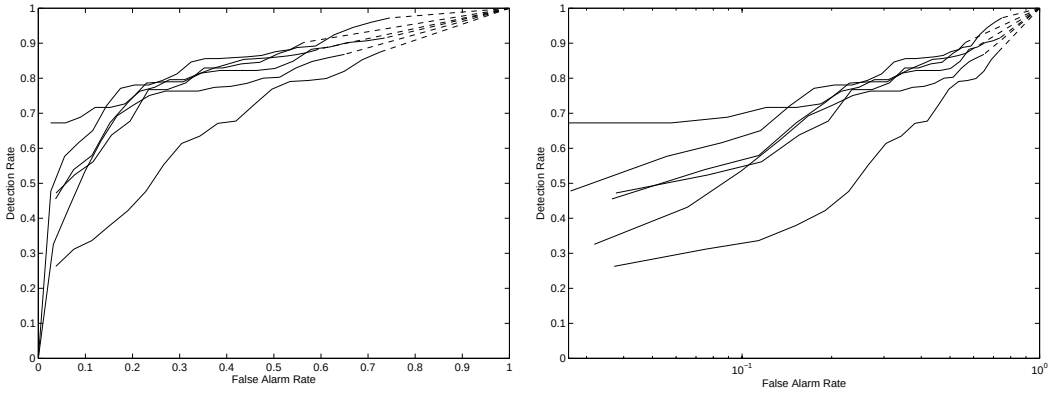


Figure 6. ROC curves for data sets 5-10 showing the detection rate plotted against the false alarm rate in linear scale (left) and logarithmic scale (right).

Selection of an appropriate threshold for target segmentation is made based on the expected distribution of target and environment surface scores. Figure 7 shows the distribution of surface scores for an entire scene, as well as for the target points separately. A threshold level equaling the sample mean plus one or two sample standard deviations is a reasonable compromise.

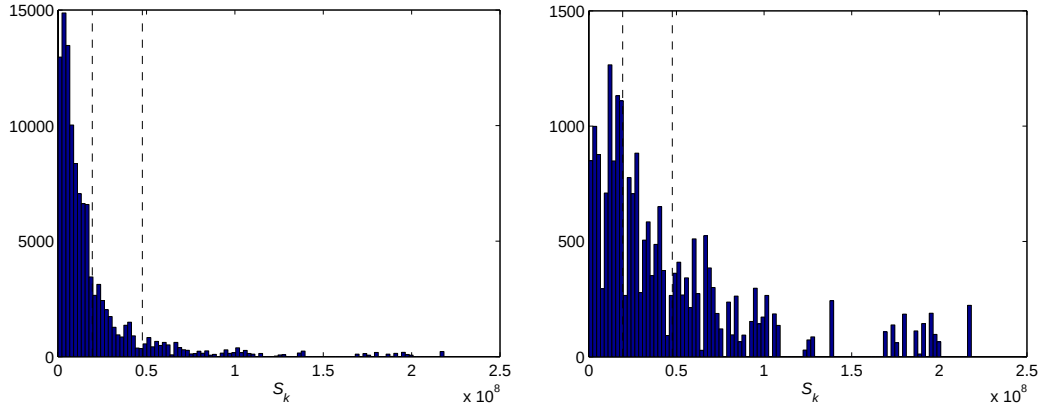


Figure 7. Normalized surface feature histograms showing the distribution of scores for data set 2 above. Left: Target and environment data points. Right: Target data points only. The dashed lines represent the sample mean and the mean plus one standard deviation in the combined data. Data points that receive a surface score of zero have been omitted.

Segmentation performance is improved using a two step hysteresis threshold. Initially data points are selected based on a high threshold, T_1, and neighboring data points are kept if their score exceeds a new, lower, threshold T_2. The algorithm chooses T_1 and T_2 based on statistics of the analyzed data point set, according to the pattern below:

$$T_1 = \mu_S + n_\sigma \cdot \sigma_S$$

$$T_2 = \mu_S ,$$

where μ_S is the sample mean and σ_S is the sample standard deviation of the calculated surface scores of the scene. The parameter n_σ sets the span between the high and low thresholds, in terms of the sample standard deviation, and values in the range $1 < n_\sigma < 3$ have been tested. Results using $n_\sigma = 2$ are summarized in Table 3. Some of the high false alarm rates (> 10 %) seen for data sets 5-10 are due to an environment surface feature present in the scene at very close range, thus containing a significant portion of the total number of data points.

The effects of using different T_1 levels are studied and some results are presented in Figure 8. In Figure 9 we see how segmentation performance can be improved by using a hysteresis threshold compared with using a single threshold. One example, presented in Figure 10, shows detection of vehicle targets using local flatness estimation and a two-step hysteresis threshold with $T_1 = \mu_S + 2\sigma$.

Table 3. Detection and false alarms using a hysteresis threshold with $T_1 = \mu_S + 2\sigma_S$ and $T_2 = \mu_S$.

	Detections		False alarms	
Data set ID	#	Rate	#	Rate
1	4695	0.3882	8100	0.0094
2	8152	0.2899	1733	0.0021
3	16828	0.5744	25040	0.0098
4	13940	0.5628	80878	0.0536
5	6428	0.7712	81142	0.0720
6	6877	0.7231	127661	0.1133
7	6744	0.7200	40798	0.0362
8	1799	0.3353	152571	0.1349
9	4334	0.7119	128777	0.1140
10	2764	0.7438	83592	0.0860

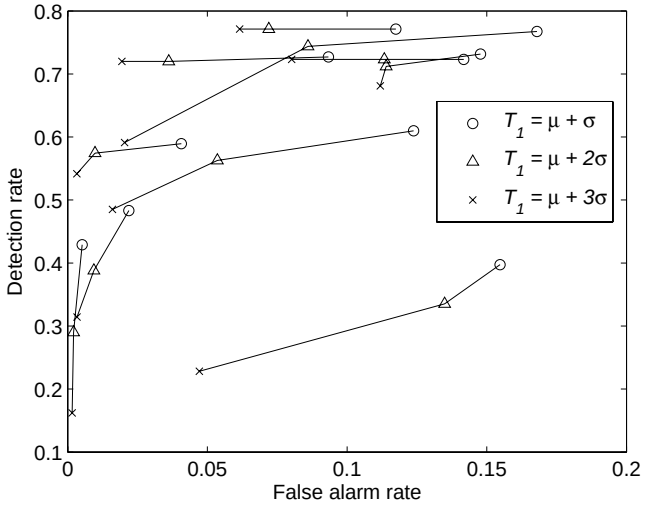


Figure 8. The detection rate and false alarm rate change as the high hysteresis threshold T_1 is raised, when analyzing data sets 1-10.

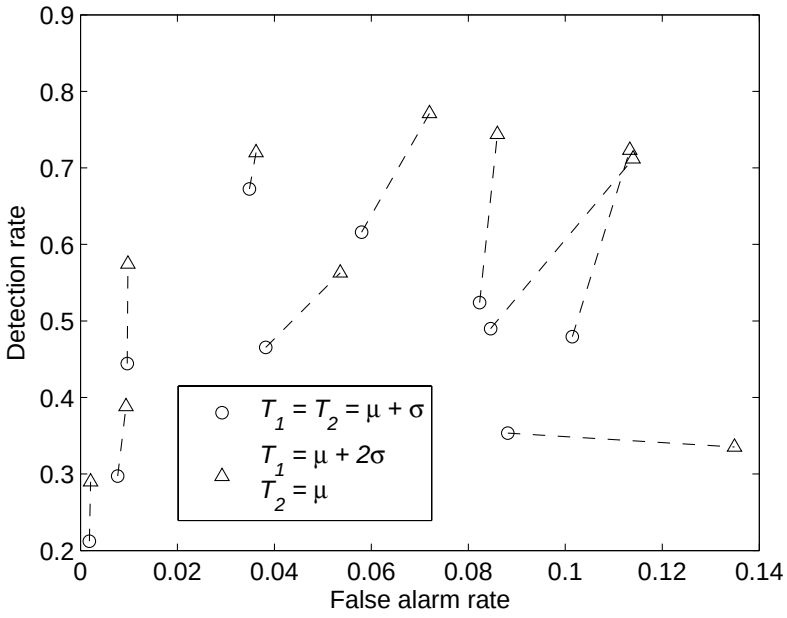


Figure 9. Comparison of detection performance using a two step hysteresis threshold vs. a single threshold., The figure shows how, for the majority of the analyzed data sets, the detection rate can be improved when comparisons are made at a similar false alarm rate.

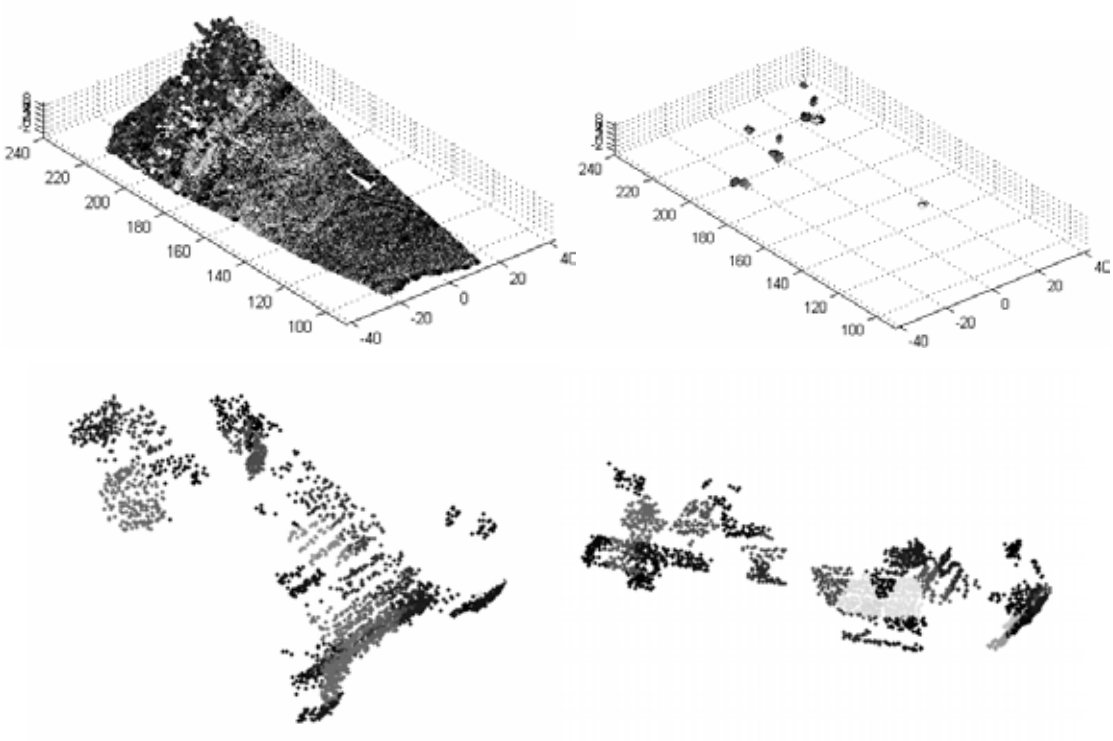


Figure 10. Detection of vehicle targets using local flatness estimation and a two-step hysteresis threshold with $T_1 = \mu_S + 2\sigma$. Top left: The complete data set 2 with vehicles in the tree line. Top right: Detected points after hysteresis threshold. Three military vehicles can be seen in the tree line, which has been removed by the algorithm, as well as points on the Volvo V70 in the field. Bottom left: Enlargement of detected points on the T-72. Bottom right: Enlargement of detected points on one MT-LB/pbv401. Axes in meters.

4.2 Change detection

If we have several data sets from a particular scene but acquired at different times, we can compare the datasets and determine whether any changes have occurred. This provides a tool for pin-pointing suspicious regions that could correspond to moving targets, even though the targets itself may be difficult to detect.

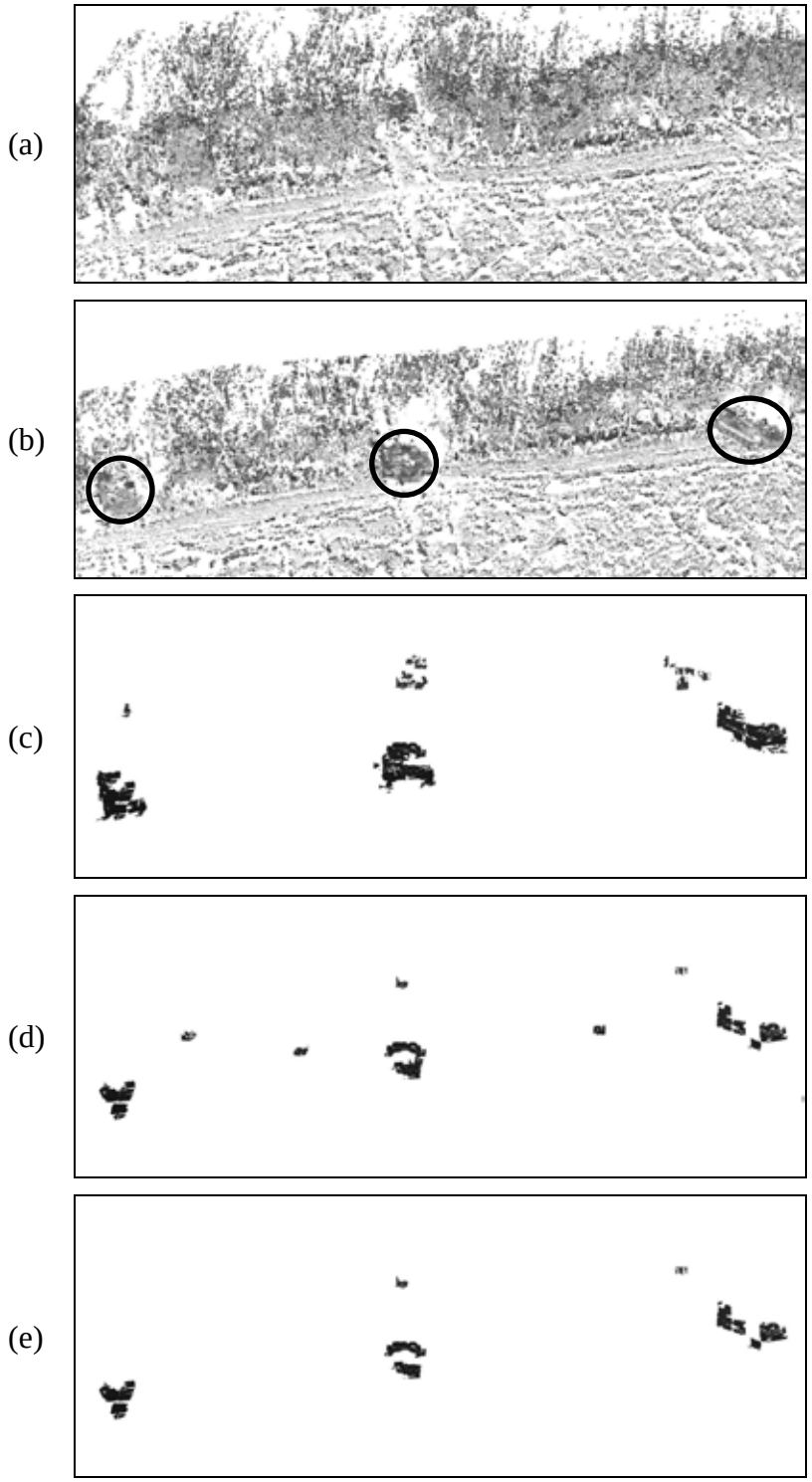


Figure 11. Change detection results. (a),(b). Reference point clouds. (b) New point cloud with changes marked. (c) Density change detection results. (d) Surfaceness change detection results. (e) Intersection of the change detection results in (c) and (d).

Assume that we have two registered data sets from a particular scene, $\mathbf{P}_1$ and $\mathbf{P}_2$ ($\mathbf{P}_2$ acquired last), and want to detect targets that have *moved within*, *entered* or *left* the scene. First, the data sets are partitioned into a number of cells (Section 4.1). Let $f_{i,k}$ denote the value of a certain feature associated with cell k in data set P_i, i=1,2. For each cell, the difference between the features in $\mathbf{P}_2$ and $\mathbf{P}_1$ are computed.

Due to non-ideal conditions, e.g. trees moving in the wind, registration errors and occlusion effects, we expect the differences to be non-zero, albeit small, for most cells. What we are looking for is *changes that cannot be ascribed to natural variations*. Hence, we expect cells that have undergone significant change to manifest themselves as *outliers* in the distribution of feature differences. Let $\overline{\Delta}_f$ and σ_f denote the mean and standard deviation of the feature differences, respectively. A change is detected if

$$f_{2,k} - f_{1,k} > \overline{\Delta}_f + n_\sigma \cdot \sigma_f \,, \tag{4}$$

where the choice of n_σ corresponds to the significance of the changes. So far, we have chosen $f_k \in \{d_k, s_k\}$, where d_k and s_k are defined in Section 4.1. Some change detection results are shown in Figure 11.

4.3 Spin-images

A spin-image, introduced by Johnson [11], is a local surface descriptor that forms a rotationally invariant two-dimensional projection of data points in a local surface environment. The basis for spin-image generation is the *oriented point* – a data point on the surface of an object and its surface normal vector.

The spin-image generation scheme used within this project largely follows that of [11]. While [11] and [12] use a mesh surface representation, we construct spin-images from the 3-D laser radar point cloud directly. This requires a pre-processing stage in which the surface normal vector is estimated, and PCA is used for this purpose, as suggested by [13]. This is a critical step for correct spin-image generation, and improper estimation parameter values or excessive noise in the data can be expected to degrade the performance of spin-image based methods, as indicated by [14]. Promisingly robust target recognition using spin-images and data from an airborne laser radar system has been reported by [15], indicating that sufficient data quality for spin-image generation can be assumed in an operational system.

Once generated, spin-images are used in model-based recognition [11, 12, 15]. The local and rotationally invariant basis of spin-images makes them robust to variations in the target aspect angle and location. By contrast, other model-based template matching techniques can require initial alignment and pose estimation, as well as the generation and use of many model templates [16] to handle these variations that will always be an intrinsic part of the detection and recognition problem in a natural scene.

A set of model spin-images is constructed from data points of known objects, and compared with spin-images generated from points in an unfamiliar scene. In order to reduce the required amount of image comparisons, the model spin-images are clustered into forming a much smaller number of spin-image prototypes that are representative of the different surfaces that make up the object. We accomplish this through *k*-means clustering of spin-images generated from all data points on the model object, as suggested in [17].

As an example, a T-72 tank positioned in an open field is scanned by laser from a range of 185 m, and spin-images are computed for all 9745 data points on the object. In Figure 12, the ten resulting spin-image prototypes and their corresponding surface point clusters are shown. The spin-image generation parameters are summarized in Table 4.

Table 4. Spin-image generation parameters.

Parameter	Value
Surface normal estimation neighborhood size	0.25 m
Spin-image bin size	0.1 m
Spin-image support distance	1.0 m

The model spin-images are matched against two different scenes, each containing a T-72 tank in a similar aspect angle. In the first scene the target is placed in a line of trees at a range of 195 m and in the second scene it was backed up further into the forest to a range of 210 m. These data sets are presented as number 1 and 2 in Table 2 in Section 4.1. The numbers of data points on the T-72 target are 7685 and 4041, respectively. All available data points are included in the generation of spin-images, which are calculated for all points pertaining to vehicles in the scene, plus an equal number of randomly sampled environment points for comparison. The total number of spin-images then becomes 56238 and 24190, respectively.

Each scene point is classified as belonging to the cluster whose prototype spin-image yields the smallest distance measure to the spin-image of that scene point. The distance metric used is simply the Euclidian distance (sum of squared differences), although it is suggested e.g. by [11] to use a more elaborate method. Results can be studied in Figure 13, where the identified affiliation of target points is shown. In both examples, it is clearly seen that several target parts are correctly classified. This result is a promising step supporting the continued analysis e. g. using geometric consistency grouping [11] for verification of recognition.

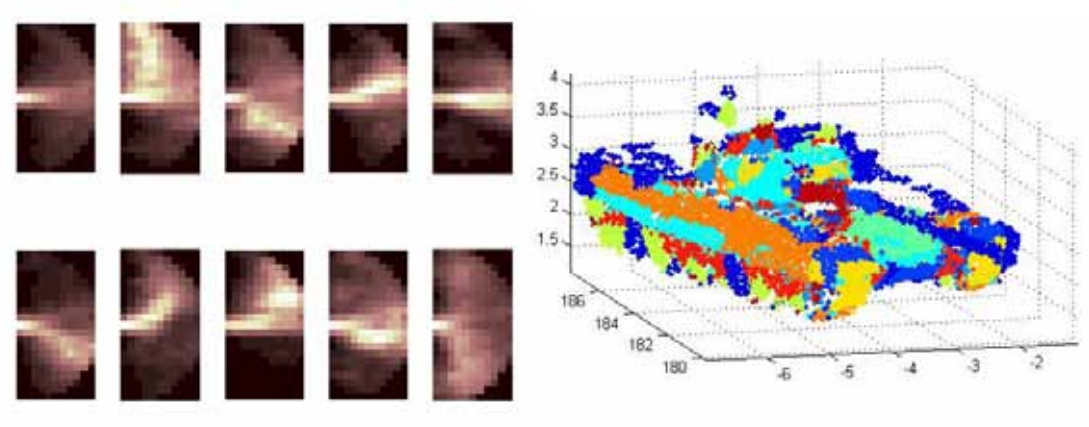


Figure 12. Ten prototype spin-images (left) created by clustering all spin-images from the T-72 tank (right). The different colors represent the associated prototype spin-image of every point on the vehicle.

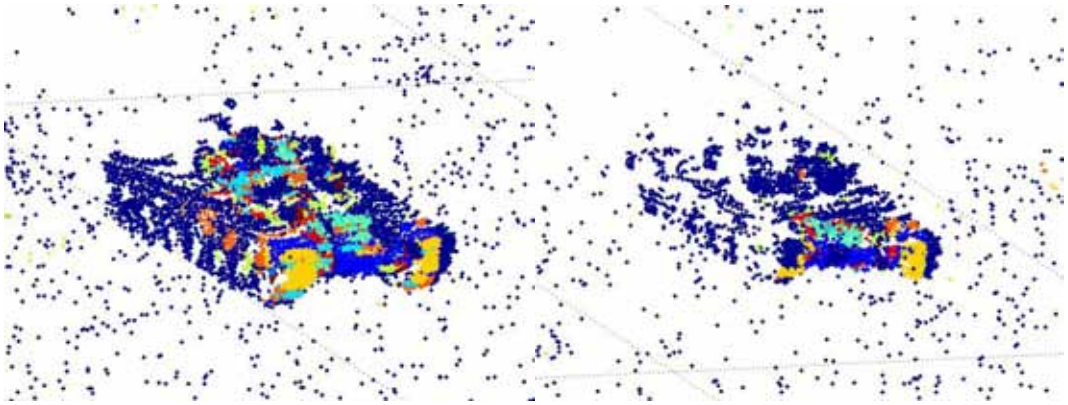


Figure 13. Spin-image matching results of a T-72 tank in a line of trees (left) and in the forest (right). Compare the cluster classification with that of the model object in Figure 12. Correspondences are easily seen e.g. on the front tracks (yellow), fender and under the front (blue), side (orange) and on the front armor (light green) as well as the area where the front armor connects to the fender (red).

4.4 Local image variance

As mentioned above, the 3-D data can also be treated as one intensity and one range image, as can be seen in Figure 14. We use conventional statistical image analysis methods like the 2-D variance to segment data. An initial test to use the local variance directly on the range image to segment the image is shown in the lowest part of Figure 3 [1]. The local variance in vertical regions (68x1 pixels) are thresholded, and the objects standing up from ground appears clearly. This method works well if the scene is not too cluttered and the aspect angle is rather flat. Segmentation was in [1] also tested on combinations of variance of available data, i.e. intensity and range. In [7], a method to align 2-D images from other sensors to 3-D data is studied. With this method we can align for instance RGB images as the one presented in Figure 14.

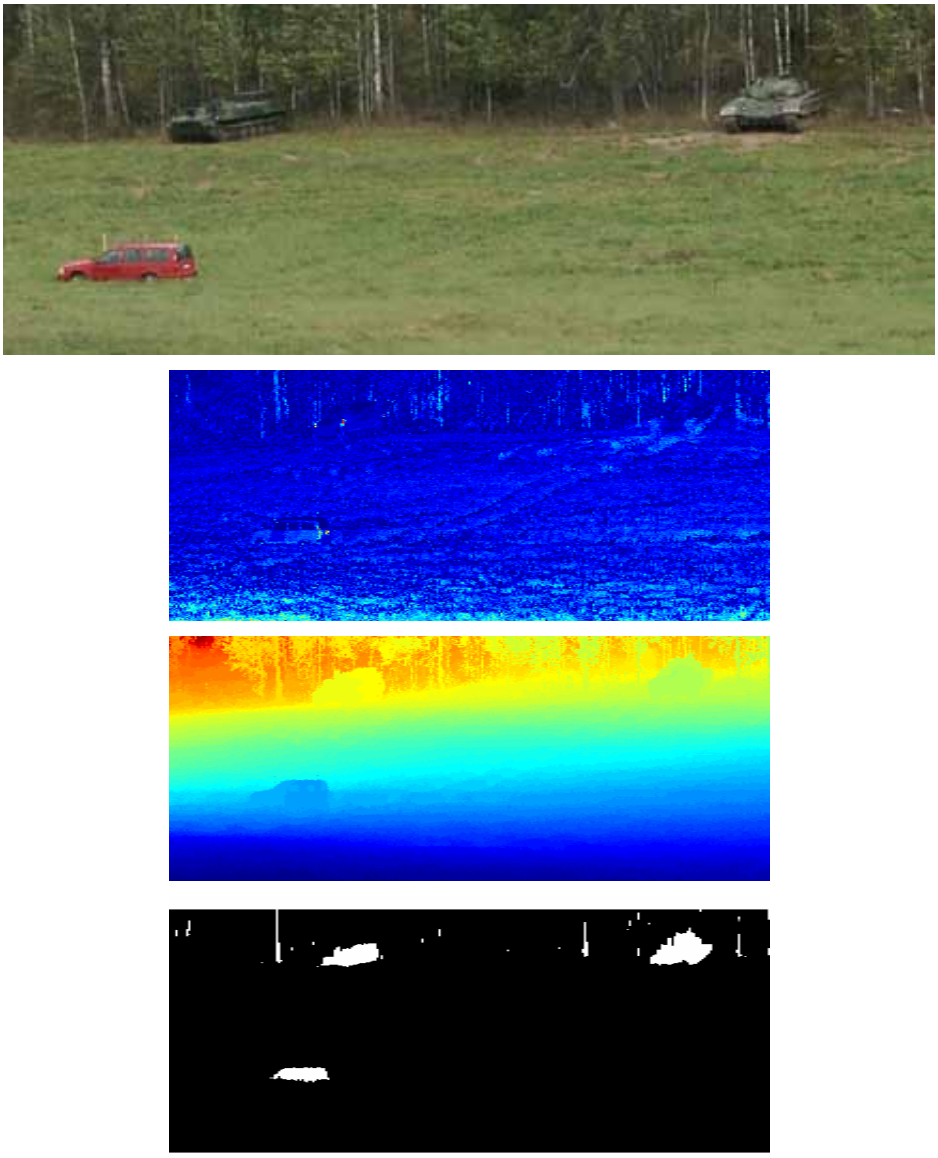


Figure 14. The information from a 3-D sensor. RGB photo of the scene (upper), intensity in laser wavelength (upper middle) and range (lower middle). The lowest image shows the thresholded local variance in vertical regions (68x1 pixels). The vehicles and some tree stems are deteced.

4.5 Fusion of height and reflectance

Early work on vehicle detection, based on 3-D data, is presented in [18, 19]. The approach can only handle vehicles that are placed on a relatively flat surface in open terrain with clear separation from the background. The rectangle fitting method can be used for object/background segmentation, which is applied as preprocessing in Papers A-B in [20].

As Figure 15 shows, the first data set used to evaluate the method were data from mine measurements. The slope of the ground surrounding the object is estimated by projections of 3-D data. The 3-D data are represented by *(x,y,z)*, where *(x,y)* is position and *z* is range. First the slope is estimated in *(x,z)* projection and the data set is rotated so that the background is flat in that projection, we now have the coordinates *(x,z')*,. The slope estimation and rotation is then repeated for the *(y,z')* projection. The result is a rotated coordinate system *(x',y',z'')*, where *(x',y')* is position on a flat surface and *z''* are height values. When the ground is flat, the object and ground can be separated by height. An example is shown in Figure 15.

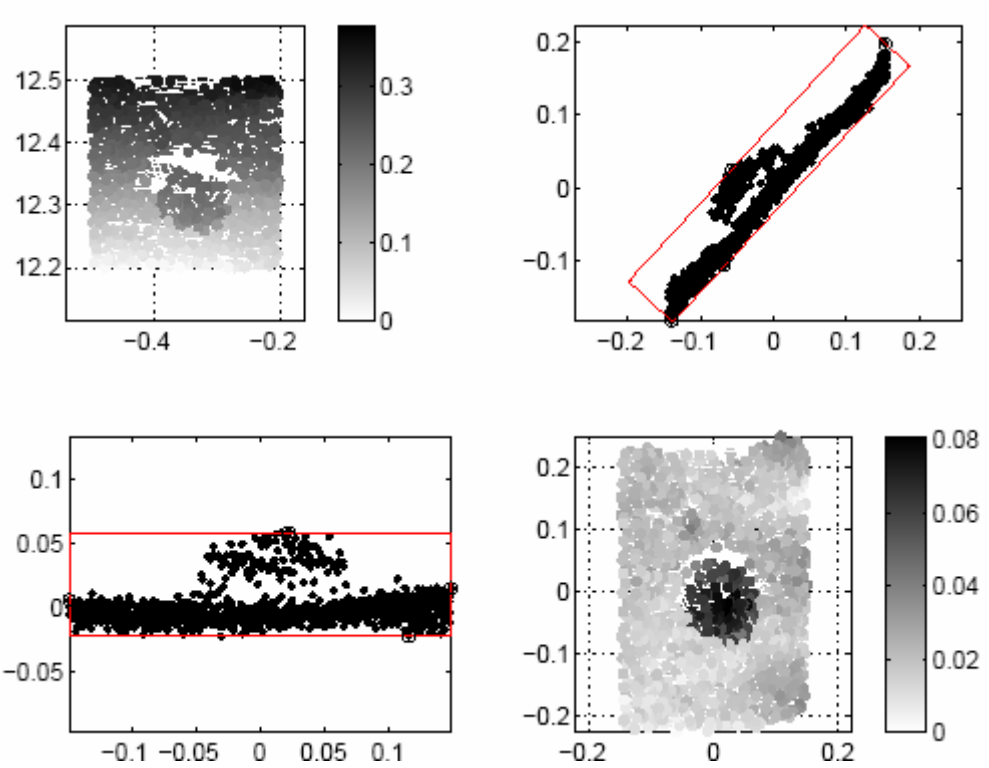


Figure 15. Example of rotation of a scene (land mine on gravel road). Top, left: original range data in (*x*,*y*,*z*) coordinates, top, right: estimated rotation in (*x*,*z*) projection, bottom, right: estimated rotation in (*y*,*z*') projection, bottom, right: final data set in (*x*',*y*',*z*'') coordinates. Axes in meters.

Figure 16. **Photograph of two mines on a gravel road.**

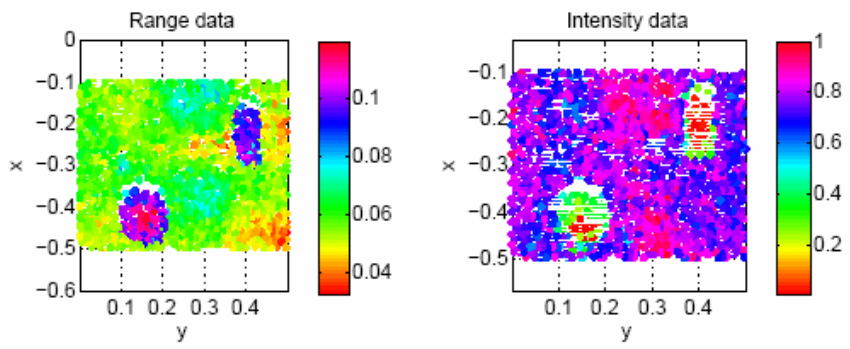


Figure 17. Range data (left) and normalized intensity data (right) of the mine scene, axes in meters.

These types of detection and object/background segmentation methods apply for the simple case with relative flat ground surface and no occluding clutter or background. Furthermore, these methods do not take advantage of the intensity information in the data set.

In Paper F in [20], a Bayesian approach for object/background segmentation is proposed. For separation of data into object and background samples, and estimation of the variances of the classes, Gaussian mixtures based on Expectation Maximization (EM) are used. A mixture of two Gaussian functions is fitted to data. These estimates are used as a priori information in a Bayesian classifier. Bayesian hypothesis testing for two classes is applied for classification of data into object and background data and clustering of object data.

This approach was tested on a scene with two mines on a gravel road, see Figure 16 for a photograph of the scene and Figure 17 for range and intensity data. The mixture of two Gaussian functions fitted to the combined range and intensity data is shown in the left part of Figure 18. In the right part of Figure 18, the segmentation and clustering of data are shown. Both objects are detected and clustered with few miss-classifications. This is the first result and further studies are needed. For example, higher order Gaussian mixtures that include position, and more complicated scenes must be investigated. A detailed description is found in [20], Paper F. The dimensions of a clustered mine can then be determined by the rectangle estimation method further explained in Section 6.1, see Figure 19.

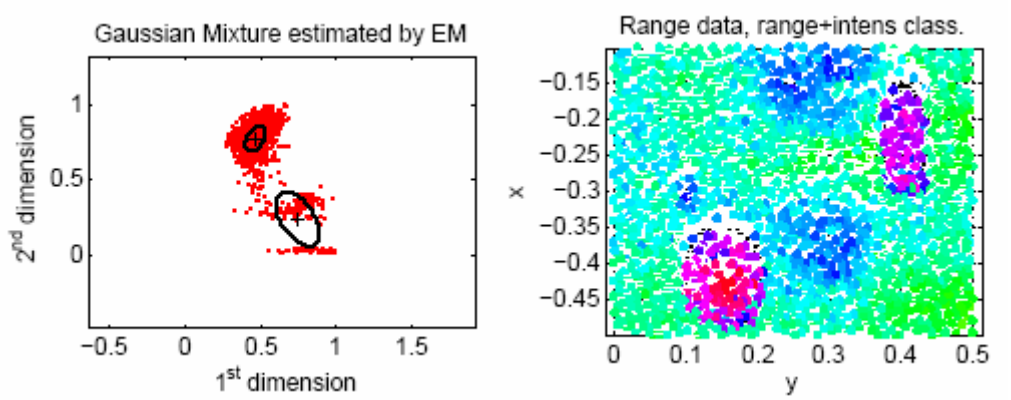


Figure 18. **Two-dimensional Gaussian mixture estimation (left) and the resulting classification and clustering (right). Axes in meters in the right part.**

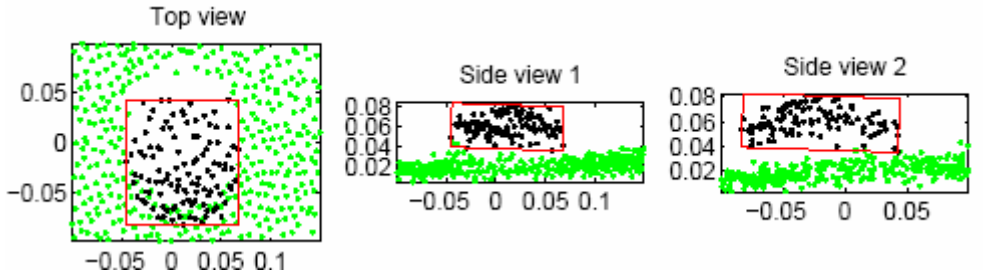


Figure 19. An example of dimension and orientation estimation of the mine in Figure 15. Object data (black), background data (gray), and the estimated rectangles are shown. Axes in meters.

5 Segmentation and classification

This section treats some methods that support the other methods with for instance ground estimation and tree removal. One classification method is also reported.

5.1 Ground estimation by watershed

The task in ground estimation is to identify data points that correspond to bare earth and then, based on this information, to intelligently "guess", i.e. interpolate, the ground level at all other positions. The input to the ground estimation technique presented in this section is a point cloud, rotated so that the *z*-axis point upwards (Figure 20a). Then, a z_{min}-image is created by rasterizing the point cloud into a grid of bins and keeping only the lowest point in each bin (Figure 20b).

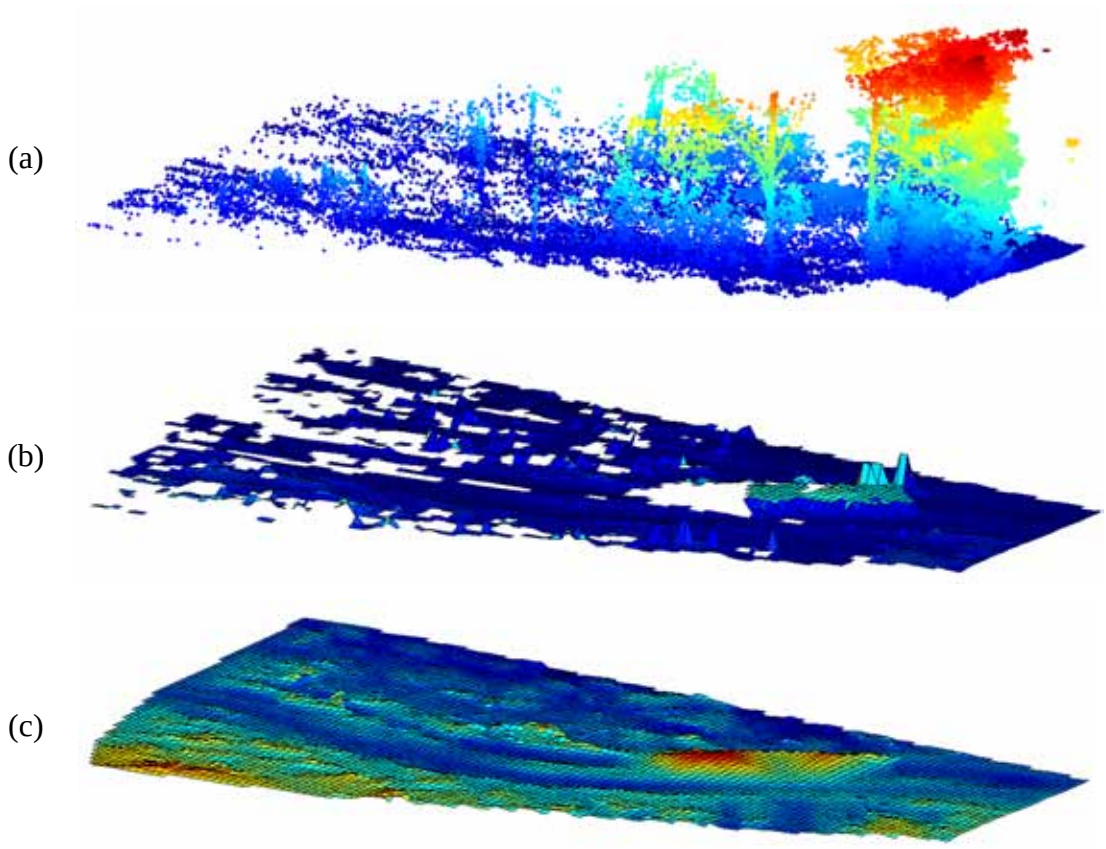


Figure 20. **Ground estimation illustrations. The color corresponds to height above the ground (blue=low, red=high). (a) Original point cloud. (b) z_{min} image (see text for details). (c) Ground model.**

Since it is highly unlikely that in every cell there are points corresponding to ground hits, we cannot let the z_{min} image itself represent the ground. Thus, a set of possible ground points is extracted by performing a watershed-like segmentation of the z_{min} image similar to the one proposed in [21]. In this report the method is extended to cope with uninterpolated images, i.e. images that contain empty pixels (to which no elevation value has been assigned). The result from the segmentation is regions where the data "point downwards", so called *basins*. The lowest point in each sufficiently large basin together form an initial set of possible ground pixels, *seeds*. Once a set of

seeds has been established, *region growing* starts from the bottom-most seed in the image, with a threshold governing to how steep elevation changes are accepted. When the region cannot grow anymore without violating the threshold constraint, the next seed is chosen from which the growing starts again.

After the region growing process there are typically regions of unvisited pixels for which we wish to estimate the ground level. By applying bilinear interpolation, the final ground model is obtained (Figure 20c).

5.2 Tree removal by local Principal Component Analysis

When handing over information from a 3-D sensing system to an operator it is preferrable to clean up the data (remove the clutter) to simplify the operator task.

After aligning the *z*-axis to vertical direction, the ground level is estimated with the technique presented in Section 5.1. Once the ground model has been created, all points that lie within a height interval (a few meters) from the ground are identified. From these points, a 2-D top-view histogram is created, in which each value represents the number of laser hits that have occurred in the corresponding area in the scene. We expect vertical structures as trees to appear as peaks in this histogram. The points corresponding to each local maximum are then selected to form a point subset, hereafter referred to as an object. At this point, however, all objects do not necessarily correspond to relevant structures in the scene, but may have arisen from laser hits in undergrowth, bushes and foliage. Therefore, we compute a number of features through PCA and exclude those who are not deemed relevant, i.e. have no evident principal direction, whose principal direction does not point approximately upwards, etc.

If we for instance extract a rough model of the ground [21] we will be able to decide the height above the ground for each point. This makes it possible to, like a cheese slicer, disregard points lower than the ankles and higher than 3 m. One example is shown in Figure 21 and Figure 22, where Figure 22 is a projection from above where the vehicles appear even more clearly.

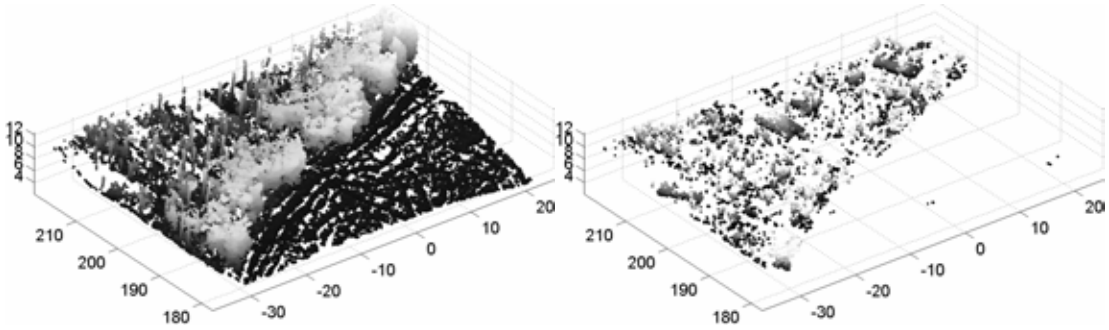


Figure 21. A 3-D data set of a treeline with 3 vehicles; unprocessed (left) and the ”slice” between 0.2 and 3 m above ground (right).

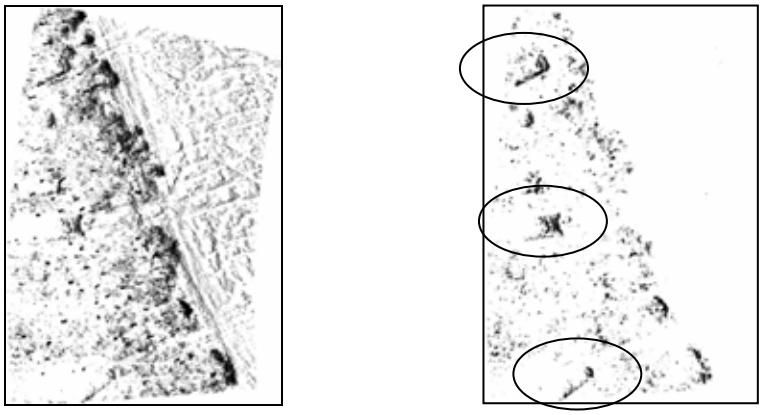


Figure 22. The same data sets as in Figure 21, but seen from above. The three military vehicles are marked with ellipses.

5.3 *Supervised classification by HNet*

In Section 4.3, it was shown how spin-images provided a sufficiently accurate surface description to classify various parts of a vehicle. In the above section, spin-images are compared using the Euclidian distance, although more complex methods are preferred. Initial tests using the HNet software by AND Corporation as a classifier were performed as an alternative. In Figure 23, results from initial testing are shown. The T-72 tank from data set 2 of Table 2 is analyzed, and the color-coded points indicate the target part class; compare this with Figure 13. These results show the potential of the HNet software to provide an aid in object classification based on spin-images. The method has not yet been compared to other classification methods.

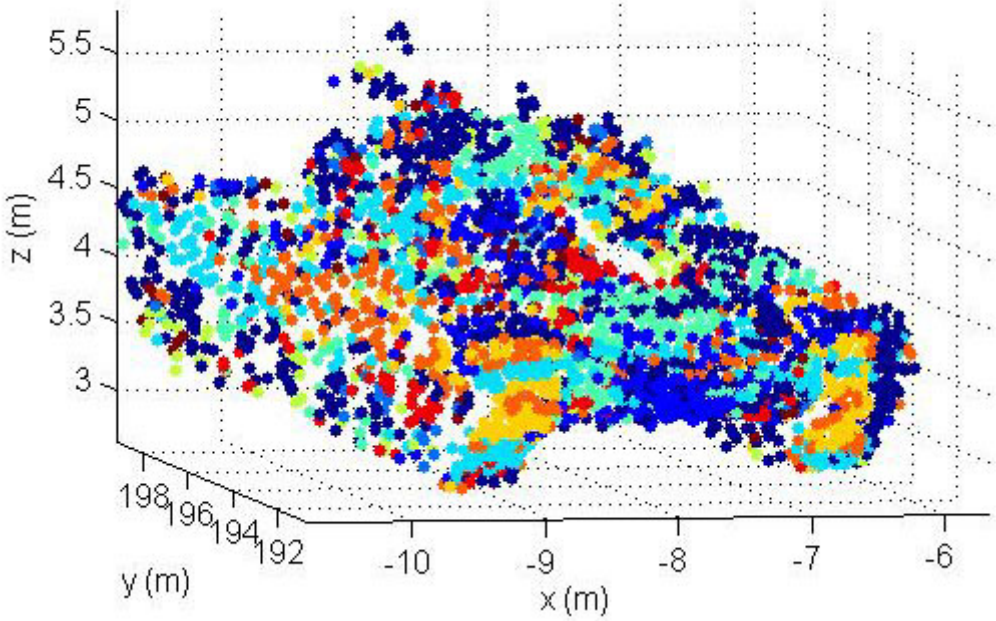


Figure 23: Identified spin-image cluster regions output by the HNet classifier during initial testing.

6 Recognition

This section describes some work concerning recognition of an object when the extents of the target is known and the data points corresponding to an objects is separated from the background, i.e. the segmentation is done. The methods cover automatic object measurement by rectangle fitting, together with treatment of targets divided into e.g. body, tower and barrel. A method to compare an object with a CAD-model library is also presented, together with some mathematical extensions to enhance the performance.

6.1 Rectangle fitting

The basis for the object recognition approach presented in the thesis [20] is rectangle fitting. The method has been described separately as *Rotating Calipers* [22-24]. A short description of the method is presented in this section, evaluation of its performance is found in [20] (Paper A, Paper F), and [4].

A straight line in 2-D can be described as $n_1x+n_2y-c=0$, where the normal vector $\mathbf{n}=(n_1,n_2)^t$ defines the slope of the line, c is the distance to origin, and $(\cdot)^t$ is matrix transpose. The object points $\varphi_i=(x_i,y_i),\ i=1,...,N$ are inside or on the side of the rectangle if

$$\begin{aligned}
&\text{Side 1}: n_1x_i + n_2y_i - c_1 \geq 0,\ i = 1,...,N\\
&\text{Side 2}: -n_2x_i + n_1y_i - c_2 \geq 0,\ i = 1,...,N\\
&\text{Side 3}: n_1x_i + n_2y_i - c_1 \leq 0,\ i = 1,...,N\\
&\text{Side 4}: -n_2x_i + n_1y_i - c_2 \leq 0,\ i = 1,...,N
\end{aligned} \qquad \textbf{(1)}$$

where $\mathbf{nn^t}=1$. The normal vector $(n_1,n_2)^t$ is orthogonal to side 1 and side 3 of the rectangle, the normal vector $(-n_2,n_1)^t$ is orthogonal to side 2 and side 4 of the rectangle and c_i is the Euclidean distance between side i and the inertia point of the rectangle, $i=1,2,3,4$. By introduction of the rotation matrix

$$\mathbf{R}^+=\begin{pmatrix}0 & -1\\ 1 & 0\end{pmatrix},$$

the parameter vector $\theta=(n_1,n_2,c_1,c_2,c_3,c_4)^t$, the regression vector $\varphi=(\varphi_1,...,\varphi_N)^t$, $\mathbf{1}=(1,1,...,1)^t$ (column with N ones), and $\mathbf{0}=(0,0,...,0)^t$ (column with N zeros), equation (1) can be written as

$$\begin{pmatrix}\varphi & -\mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \varphi\mathbf{R}^+ & \mathbf{0} & -\mathbf{1} & \mathbf{0} & \mathbf{0}\\ -\varphi & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0}\\ -\varphi\mathbf{R}^+ & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1}\end{pmatrix}\theta\geq 0$$

A rectangle that contains all samples φ inside or on the rectangle's edge is found by

$$\min \ (c_3 - c_1)(c_4 - c_2)$$

subject to

$$\begin{pmatrix} \varphi & -1 & 0 & 0 & 0 \\ \varphi \mathbf{R}^+ & 0 & -1 & 0 & 0 \\ -\varphi & 0 & 0 & 1 & 0 \\ -\varphi \mathbf{R}^+ & 0 & 0 & 0 & 1 \end{pmatrix} \theta \geq 0$$

$$\mathbf{n}^t \mathbf{n} = 1.$$

This rectangle will also contain the convex hull of the data set. An example is shown in Figure 24. In both [23] and [22] (similar) algorithms are given for calculation of the minimal area in linear time, i.e., $O(N_v)$ where N_v is the number of vertices in the convex polygon. Further, the convex hull can be calculated in $O(N \log_2 N)$ time, where N is the number of samples, if data are unsorted and in $O(N)$ time if data are sorted.

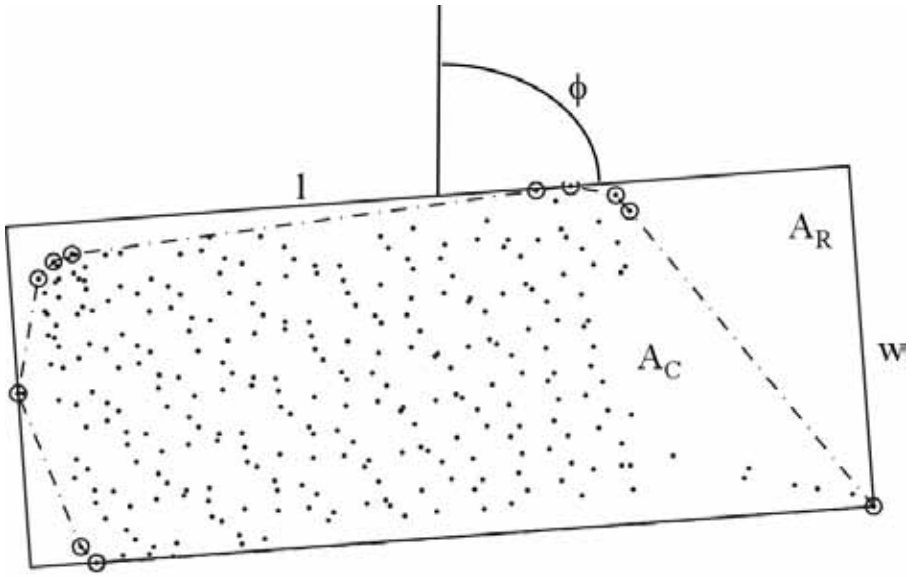


Figure 24. Illustration of the rectangle estimation. A set of samples (dots), the convex hull (dashed line), and the estimated rectangle (solid line) are shown. The samples belonging to the convex hull are encircled. The length (*l*), width (*w*), orientation (ϕ), convex hull area (A_C), and rectangle area (A_R) are indicated.

6.2 Recognition of articulated objects

In [23, 24] it was shown that the rectangle fitting method could be used for dimension and orientation estimation of man-made objects. An example of rectangle fitting of a land mine in Figure 16 is shown in Figure 19.

The rectangle fitting method has also been used in an approach for recognition of articulated objects, see [20], Paper A. In Paper B of [20], a penalty function for the number of functional parts, and an iterative least squares fitting method with outlier rejection are proposed.

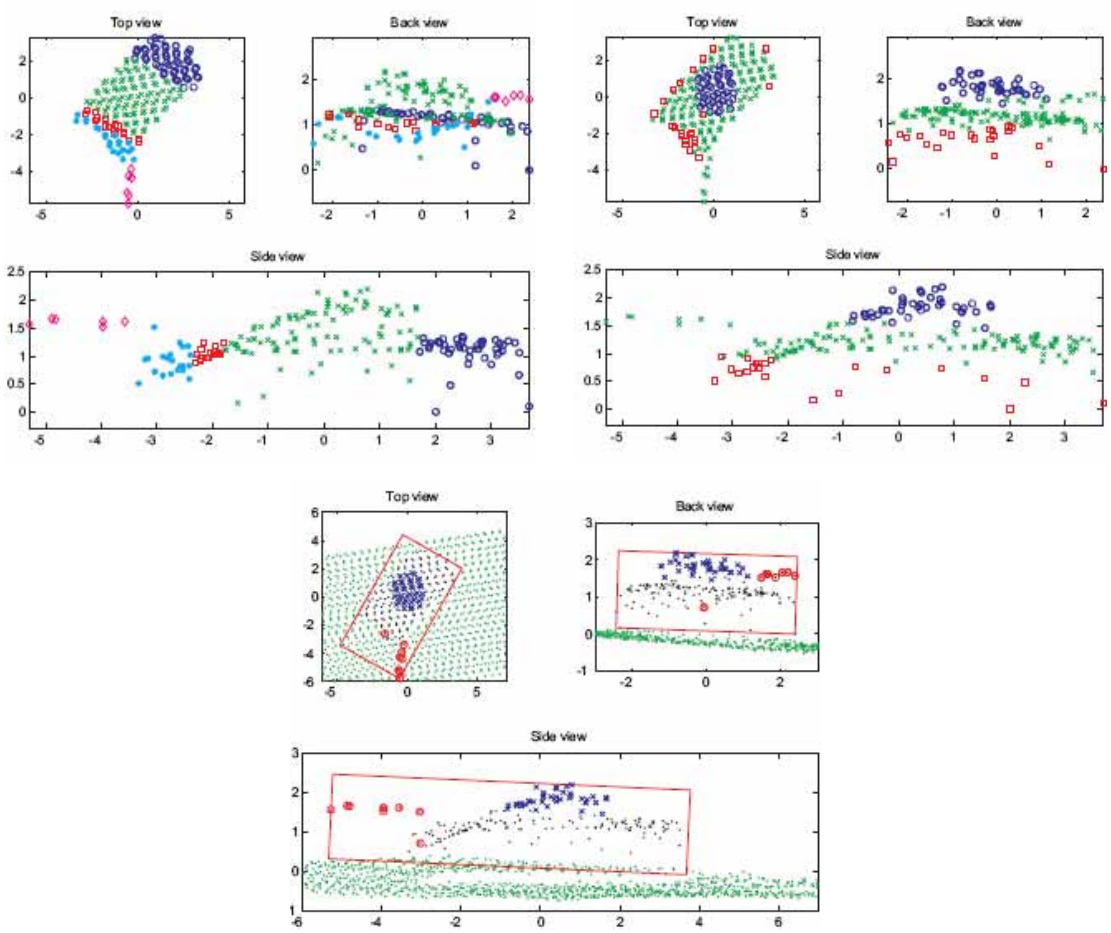


Figure 25. Result of size and orientation estimation, segmentation, and node classification. Top left: Side view, short side segmentation. Data are divided into five segments, where one is identified as a barrel (marked with rhombus). Top right: Side view, long side segmentation. Data are divided into three segments, where one is identified as a turret (marked with circles). Bottom: The rectangles show the estimated size and orientation. Identified barrel samples are marked with 'o' and turret samples with 'x'. Axes in meters.

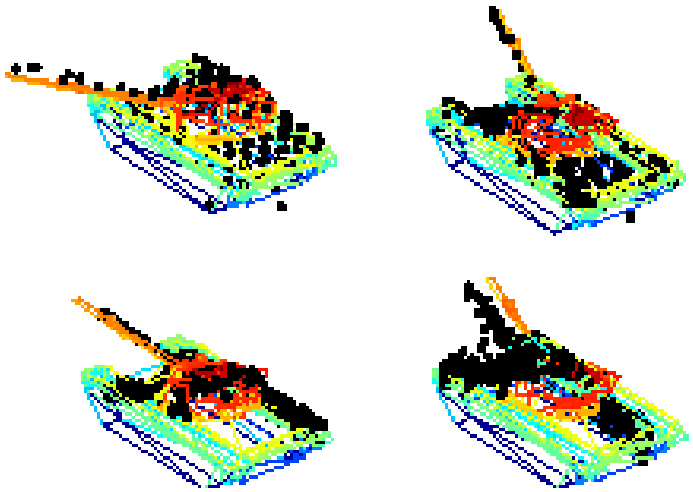


Figure 26. Matching results, tank (T-72) data collected with three different laser radar systems are matched with the T-72 model.

The method handles general, irregularly sampled, scattered, 3-D data. It takes advantage of the 3-D structure and that the dimensions are known in laser radar data. The estimation of initial position and segmentation into functional parts is based on the assumption that man-made objects, like vehicles and buildings, in certain projections are of rectangular shape. A man-made object of complex shape can be decomposed into a set of rectangles and in some views the rectangles will describe the functional parts of the object. In a general application we cannot assume that the object is placed in a certain orientation relative to the sensor or that the object is articulated in a specific way.

The object recognition method consists of four steps:

1. Estimate the object's 3-D size and orientation using the rectangle estimation method described in Section 6.1.
2. Segment the object into parts of approximately rectangular shape. The functional parts can be found in some of the rectangles.
3. Identify the functional parts by simple geometric comparisons and estimate their dimensions and orientations.
4. Match the entire object with a wire-frame model. The model's functional parts are rotated to the estimated orientations.

The goal with identification and fitting of functional parts for vehicles is to simplify the model matching. If the object's parts are identified, matching with model can be performed regardless of the relative position of the functional parts. Different configurations of a vehicle can be handled in a structural way. If the functional parts of a tank (the barrel and turret) can be extracted, the hypothesis that the object is a tank is strengthened. When the object's functional parts are identified, the recognition can be simplified as the number of degrees of freedom reduce. Further, for a tank the orientation of the barrel indicates the tank's intention, which can be useful in security or military applications.

An example of identification of functional parts for a tank is shown in Figure 25. The segmentation into rectangular parts is performed in top, side, and front/back view projections. For every projection the segmentation is performed along both the main and the secondary axis, where the axes are estimated with rectangle fitting. In total, the object is segmented in six different ways and all rectangles are compared with the library model's main parts using geometric rules for dimensions and orientations. The matching with a facet model (CAD model) is shown in Figure 26.

6.3 Matching of articulated objects with facet models

The model matching in Section 6.2 is based on global matching of data and model. This approach can be developed to modular matching, where the articulated parts are matched in controlled way. Further, to control the number of articulated parts that are valid for the particular data set, a penalty function is proposed. This work is presented in [20], Paper B and [25]. In this section, Least Squares fitting with point correspondence between the data set and the model is described. After that, the case where point correspondence is not present is described. In the latter case it is common to use the ICP-algorithm described in Section 3.2. Finally, an extension of ICP that includes outlier rejection is proposed.

6.3.1 Least Squares fitting with point correspondence

First, the global Least Squares fitting problem of two 3-D point scatters with point correspondences is presented [26]. The problem is then extended to modular Least Squares fitting where the object's articulation is treated, as proposed in [25] and [20], Paper B. An illustration is shown in Figure 27. The vertices of a facet model are used as the point set representing the model and a rotated and translated copy of the model samples represents the object. The object samples are contaminated with Gaussian noise with zero mean and standard deviation of 0.1 m (on an object of size 9.65×3.52×2.49 meter). In Figure 27, the results of global Least Squares fitting and the result of modular Least Squares fitting are shown. The Mean Square Error (MSE) is reduced approximately 500 times in this case. The model samples are represented by the facet model in the figure.

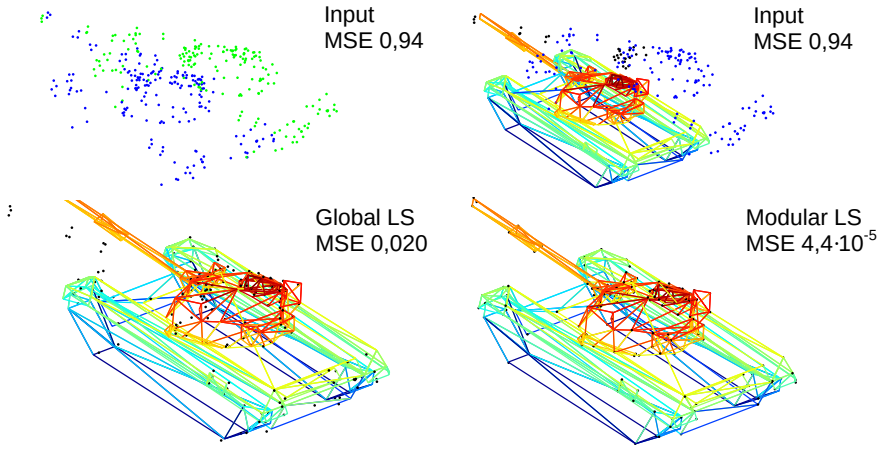


Figure 27. Geometric fitting of two point scatters with point correspondence. The two point scatters (top, left), the model point scatter represented by the facet model (top, right). Fitting using global Least Squares (bottom, left) and modular Least Squares (bottom, right). The MSE of the fit is given.

6.3.2 Least Squares fitting of 3-D points and facet model

In most cases, two point sets with point correspondence are not available. Instead there is a point scatter describing the object and the model is a facet model, denoted $\mathcal{M}$. It is then possible to fit the object samples with their projections on the closest facets. Due to the projections, the fitting problem is a nonlinear problem which can be solved within the ICP framework. Define $\boldsymbol{P}$ as the point set describing the object and $\boldsymbol{Q}$ as the point set describing the model, where $\boldsymbol{Q}$ is the projection of the elements in $\boldsymbol{P}$ to the closest model facet, i.e.,

$$Q = \text{Proj}(P|\mathcal{M})$$

If the orthogonal projection of an element in $\boldsymbol{P}$ is not on a facet, the projected sample is set to the closest facet edge. Again, we use ICP to iteratively enhance the registration of two data sets, as for instance $\boldsymbol{Q}$ and $\boldsymbol{P}$. In applications with noisy data an outlier rejection is needed; elements in $\boldsymbol{Q}$ that have too long distances to the corresponding samples in $\boldsymbol{P}$ will be rejected. The outlier distance depends on the uncertainty in data and the resolution in the facet model. An iterative algorithm for fitting a 3-D point set with a facet model, when the number of functional parts is fixed to J, is proposed in Algorithm 1.

Algorithm 1. Modular ICP with Outlier Rejection

- Estimate the object's orientation, including orientation of functional parts, and place the model in similar position. This gives the initial rotations $\mathbf{R}_1^0,...,\mathbf{R}_J^0$ and translations $\mathbf{T}_1^0,...,\mathbf{T}_J^0$.
- For iteration *k*, calculate the closest points of $P_j^k, j=1,...,J$ on the model $\mathcal{M}$, $Q_j^k = \text{Proj}(P_j^k|\mathcal{M})$, to get point correspondences.
- Reject outlier elements in Q_j^k and their corresponding elements in $P_j^k, j=1,...,J$.
- Estimate rotations $\mathbf{R}_1^k,...,\mathbf{R}_J^k$ and translations $\mathbf{T}_1^k,...,\mathbf{T}_J^k$.
- Calculate the MSE of the estimation error, $V^k(\mathcal{M})$.
- If $\tau > V^k(\mathcal{M})/V^{k-1}(\mathcal{M})$, terminate. Otherwise, continue to iteration *k+1*. The threshold τ is user-defined.

If Algorithm 1 is compared with the original ICP-algorithm, the outlier rejection in step 3 is added and the termination criterion is relative instead of absolute. The impact of the outlier rejection is illustrated in Figure 28. The data set is simulated using the vertex points from a model (a T-72 chassis), the samples are rotated 10 degrees and translated 0.5 meters in 3-D. Gaussian noise with zero mean and standard deviation 0.05 meters is added. To simulate outliers, Gaussian noise with zero mean and standard deviation 3 meters is added to seven samples. Algorithm 1 is applied to 100 data sets of this type, both with an outlier rejection distance of 1 meter and without outlier rejection. Tests have shown that an outlier distance of 5σ or larger is sufficient, where σ is the standard deviation of the noise in input data. Statistics of the root mean square error for the last iteration, $\sqrt{V^k(\mathcal{M})}$, in each example are shown in Figure 28. The root mean square errors are more than 5 times higher when the outlier rejection is not applied. The final fit for the data set in the top of Figure 28 is shown in the bottom image of Figure 28, outlier rejection was applied.

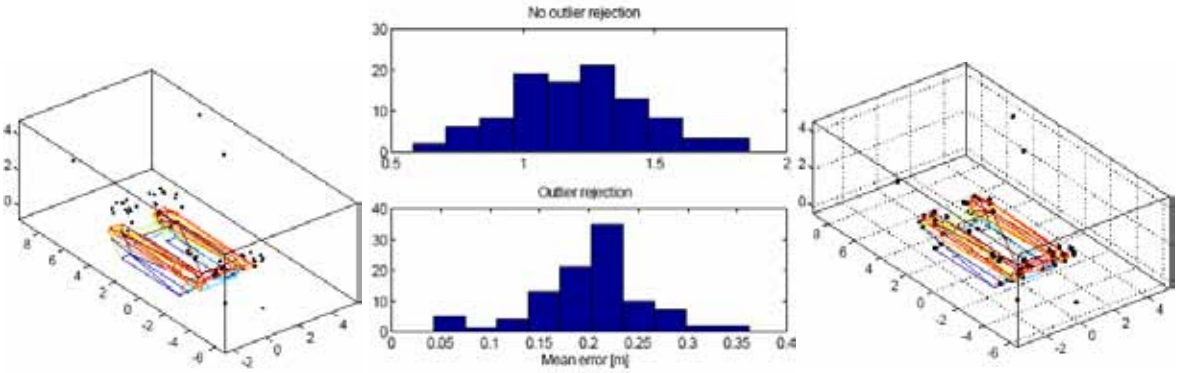


Figure 28. Example of ICP with outlier rejection. Left: Initial fit. Middle: Statistics of final root mean square error for 100 trials. Right: Final fit, outlier rejection of 1 meter was applied. Axes in meters.

7 Discussion

The well spread effort reported here are distributed over registration of 3-D laser data, detection of abnormal objects or structures in terrain, and detection of known geometric signatures in natural environment. Also covered are supportive methods for segmentation as adaptive ground level estimation and automated tree removal, and recognition methods to use when data is segmented and a target has to be verified.

The methods show potential to fill the chain of needed signal processing operation from raw data to recognition, but since the goal has not been to develop a real-time system, effort has not been put on those aspects, but rather on performance issues. There are many parts in this work that need to be more thoroughly examined, e.g. more objective performance estimations of some methods.

An experience from this work is the problems with simulated data. While methods for recognition from 2-D data in many cases can rely on simulated training data to a large extent, 3-D data is so complex to reconstruct, that it is very difficult to get realistic data. One reason can be that 2-D simulations often are based on textures, while 3-D simulations need to be based on geometric shapes. Visually they can be similar, but when it comes to real calculations the differences are crucial.

This work has also been useful to get a feeling for the performance of the laser scanner system described in Section 2.1. The range limits of this system have been tested and the uncertainties of laser systems have been estimated. These limits will of course also set the limits to the performance estimations of the recognition methods.

For future military applications, focal plane arrays are expected to be more frequently used. This will affect the signal processing mainly by the amount of data delivered to the processor, since the mechanics often are the bottle-neck of the current systems. Another important future issue can be the range accuracy. Generally, focal plane arrays has worse range accuracy, due to priority issues in the electronics. This would indeed affect the signal processing possibilities negatively.

7.1 Future work

This report shows several separate methods to process 3-D data. A future task will be to connect these methods to complete a chain of operations from raw 3-D laser data until performed recognition. To do this we have to specify the application, since it will not be realistic to make a general program for all systems and applications.

In most work with the methods described in this report we have had no information about the surroundings, the global position, and the zenith direction. In the future we expect to have systems delivering some of this information to support the processing. When the zenith direction is available a lot of assumptions can be made, and the methods can be more specified and optimized. If also the GPS position and the system orientation is known an initial guess for the registration would be just straight-forward geometry.

Another issue is the system usage. In this report we have only considered general 3-D laser data, without a system desire. A proper use would either be an unmanned surveillance system placed to watch for intrusions, or a multi-sensor platform, where another large-field-of-view sensor gives a position for the laser system to examine further. If this was the case, a more specific and better-performing signal processing chain could be developed.

8 References

[1] T. Chevalier, P. Andersson, C. Grönwall, F. Gustafsson, J. Landgård, H. Larsson, D. Letalick, A. Linderhed, O. Steinvall, and G. Tolt, "Årsrapport 3D-laser 2005," FOI, Linköping, User Report, FOI-R--1807--SE, Dec 2005.

[2] Website: *TopEye AB*, "http://www.topeye.com ", 2005

[3] O. Steinvall, L. Klasén, T. Chevalier, P. Andersson, H. Larsson, M. Elmqvist, and M. Henriksson, "Grindad Avbildning - fördjupad studie," FOI, Sensor Technology, Linköping, Scientific Report, FOI-R--0991--SE, Nov 2003.

[4] C. Grönwall, F. Gustafsson, and M. Millnert, "Ground target recognition using rectangle estimation," *IEEE Transactions on Image Processing*, 15(11):3400-3408, 2006.

[5] C. English, S. Ruel, L. Melo, P. Church, and J. Maheux, "Development of a practical 3D automatic target recognition and pose estimation algorithm," in *SPIE D&S*, vol. 5426. Orlando, FL, USA: SPIE, 2004, pp. 112-123.

[6] P. B. van Wamelen, Z. Li, and S. S. Iyengar, "A fast expected time algorithm for the 2-D point pattern matching problem," presented at Pattern Rcognition, 2004.

[7] K. Chan, "Registration of 3-D laser radar data with hyperspectral imagery for target detection," FOI, Linköping, Technical Report, FOI-R--2101--SE, Nov 2006.

[8] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes. ," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239-256, 1992.

[9] Y. Liu, "Improving ICP with easy implementation for free-form surface matching," *Pattern Recognition*, 37(2):211-226, 2004.

[10] A. Wiklund, "Registration approaches for noisy 3D data representing natural scenes," FOI, Linköping, FOI-R--1994--SE, 2006.

[11] A. E. Johnson, "Spin-Images: A Representation for 3-D Surface Matching," PhD Thesis, Robotics Institute. Pittsburgh, Pennsylvania: Carnegie Mellon University, 1998. (robotics.jpl.nasa.gov/people/johnson/thesis/thesis.html)

[12] S. Ruiz-Correa, L. G. Shapiro, and M. Meila, "A New Paradigm for Recognizing 3-D Object Shapes from Range Data," presented at Ninth IEEE International Conference on Computer Vision (ICCV 2003), Nice, France, 2003.

[13] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface Reconstruction from Unorganized Points," *Computer Graphics*, 26(2):71-78, 1992.

[14] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik, "Recognizing objects in range data using regional point descriptors," presented at European Conference on Computer Vision (ECCV), 2004.

[15] A. N. Vasile and R. M. Marino, "Pose-Independent Automatic Target Detection and Recognition using 3-D Ladar Data," in *SPIE D&S*, vol. 5426. Orlando, FL, USA, 2004, pp. 67-83.

[16] Q. Zheng, S. Z. Der, and H. I. Mahmoud, "Model-based Target Recognition in Pulsed Ladar Imagery," *IEEE Trans. Image Processing*, 10(4):565-572, 2001.

[17] Y. Shan, H. S. Sawhney, B. Matei, and R. Kumar, "Shapeme Histogram Projection and Matching for Partial Object Recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(4):568-577, 2006.

[18] C. Carlsson (later Grönwall), E. Jungert, C. Leuhusen, D. Letalick, and O. Steinvall, "A comparative study between two target detection methods applied to ladar data," presented at the 9th Conference on Coherent Laser Radar, Linköping, 1997.

[19] C. Carlsson (later Grönwall), E. Jungert, C. Leuhusen, D. Letalick, and O. Steinvall, "Target detection using data from a terrain profiling laser radar," in *Third international airborne remote sensing conference and exhibition*, vol. I. Copenhagen, Denmark, 1997, pp. I-431 - I-438.

[20] C. Grönwall, "Ground Object Recognition using Laser Radar Data - Geometric Fitting, Performance Analysis, and Applications," PhD thesis, Department of Electrical Engineering. Linköping: Linköping University, 2006.

[21] J. Landgård, "Segmentering och klassificering av LIDAR-data," Dept. of Electrical Engineering, Linköping University, Linköping, LITH-ISY-EX-05/3729-SE 2005.

[22] G. Toussaint, "Solving geometric problems with the rotating calipers," presented at IEEE MELECON, Athens, 1983.

[23] C. Carlsson (later Grönwall), "Vehicle size and orientation estimation using geometric fitting," Technical Report, Licentiate thesis, Department of Electrical Engineering. Linköping: Linköping University, 2000.

[24] C. Carlsson (later Grönwall) and M. Millnert, "Vehicle size and orientation estimation using geometric fitting," presented at SPIE, Orlando, FL, USA, 2001.

[25] C. Grönwall, P. Andersson, and F. Gustafsson, "Least squares fitting of articulated objects," in *CVPR 2005, Workshop on advance 3D image analysis for safety and security*. San Diego, CA, USA, 2005, pp. 116-121.

[26] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two point sets.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698-700, 1987.