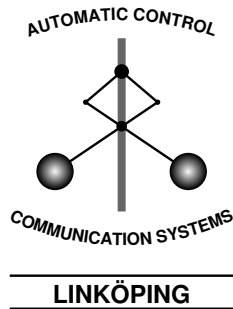


# Ground Target Recognition in a Query-Based Multi-Sensor Information System

Jörgen Ahlberg, Martin Folkesson, Christina Grönwall, Tobias  
Horney, Erland Jungert, Lena Klasén, Morgan Ulvklo

Division of Automatic Control  
Department of Electrical Engineering  
Linköpings universitet, SE-581 83 Linköping, Sweden  
WWW: <http://www.control.isy.liu.se>  
E-mail: [jorahl@foi.se](mailto:jorahl@foi.se), [marfol@foi.se](mailto:marfol@foi.se), [stina@isy.liu.se](mailto:stina@isy.liu.se),  
[tobho@foi.se](mailto:tobho@foi.se), [jungert@foi.se](mailto:jungert@foi.se), [lena@foi.se](mailto:lena@foi.se), [morgan@foi.se](mailto:morgan@foi.se)

16th October 2006



Report no.: LiTH-ISY-R-2748

Technical reports from the Control & Communication group in Linköping are  
available at <http://www.control.isy.liu.se/publications>.



## Abstract

We present a system covering the complete process for automatic ground target recognition, from sensor data to the user interface, i.e., from low level image processing to high level situation analysis. The system is based on a query language and a query processor, and includes target detection, target recognition, data fusion, presentation and situation analysis. This paper focuses on target recognition and its interaction with the query processor. The target recognition is executed in sensor nodes, each containing a sensor and the corresponding signal/image processing algorithms. New sensors and algorithms are easily added to the system. The processing of sensor data is performed in two steps; attribute estimation and matching. First, several attributes, like orientation and dimensions, are estimated from the (unknown but detected) targets. These estimates are used to select the models of interest in a matching step, where the target is matched with a number of target models. Several methods and sensor data types are used in both steps, and data is fused after each step. Experiments have been performed using sensor data from laser radar, thermal and visual cameras. Promising results are reported, demonstrating the capabilities of the target recognition algorithms, the advantages of the two-level data fusion and the query-based system.

**Keywords:** Multi-sensor fusion, query languages, infrared sensors, laser radar, range data, target recognition, target detection.



# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>The Query-Based Information System</b>	<b>4</b>
2.1	The Query Execution Process . . . . .	4
2.2	The Query Processor . . . . .	5
2.3	Data Fusion . . . . .	7
2.4	The Simulation Environment . . . . .	8
<b>3</b>	<b>Sensor Data Analysis</b>	<b>8</b>
3.1	Sensor Data . . . . .	8
3.2	The Target Recognition Process . . . . .	8
3.2.1	Attribute Estimation . . . . .	9
3.2.2	Model Matching . . . . .	9
3.3	Attribute Estimation from 2D Image Data . . . . .	10
3.4	Attribute Estimation from 3D Scatter Point Data . . . . .	11
3.5	Model Matching on 2D Image Data . . . . .	11
3.6	Model Matching on 3D Scatter Point Data . . . . .	12
<b>4</b>	<b>Experiments and Results</b>	<b>13</b>
4.1	Data Acquisition . . . . .	13
4.2	Attribute Estimation on 2D LWIR Data . . . . .	14
4.3	Attribute Estimation on 2D DEM Data . . . . .	17
4.4	Attribute Estimation on 2D NIR Data . . . . .	17
4.5	Attribute Estimation on 3D Data . . . . .	18
4.6	Cross-Validation . . . . .	18
4.7	Model Matching . . . . .	18
<b>5</b>	<b>The Execution of a Query: Recognition and Fusion</b>	<b>22</b>
<b>6</b>	<b>Discussion</b>	<b>24</b>
<b>7</b>	<b>Conclusions</b>	<b>25</b>



# 1 Introduction

We present an information system with capability for detection and recognition of ground targets, mainly military vehicles, using various types of electro-optical sensors. The long term goal is to cover the complete process of target recognition, from the sensors to the decision support in a network centric defence. Thus, we need to develop algorithms for target detection, target recognition and data fusion. We also need a system architecture that controls the choice of sensors and algorithms, as decision support tools [17] in the near future will access data from a large number of sensors located on different platforms. These tools will be integrated in a command and control system, and might be so complex that they require users with special training. To reduce the requirements on the users, efforts must be made to design a *usable system* [29]. The users will have a higher trust in such a system and also be able to focus on their primary tasks.

The system must be able to autonomously perform tasks that would otherwise require the users' full attention and/or specialist knowledge. In the case of target recognition, the system itself must select sensors and algorithms for sensor data analysis, i.e., from a user's perspective, the computational model must be *sensor data independent* [22]. To establish sensor data independence, the system must be capable to 1) select sensors considering availability, coverage, weather, and light conditions; 2) select sensor data analysis algorithms considering functionality, complexity, available sensor data, and requested target types; and 3) control the sensor data fusion and determine the interconnections between the controlling part and the fusion process. A system with these capabilities is able to select the most appropriate sensor(s) and recognition algorithm(s) and to access, analyze, and eventually fuse the information gathered from the sensor data analysis.

It is also of importance that the users are allowed to define relevant and application-oriented goals [27], and that the system can acquire the information needed to reach the given goal, i.e., the system must be goal-driven. To accomplish this we have developed a query language and implemented a query-driven system. To allow the system to select any available sensor, all sensors must have a standard interface to the system, so that new or reoccurring sensors (and sensor types) can be plugged in without any interaction from the user. From the system's point of view, a sensor and a sensor data analysis algorithm (performing target detection or recognition) forms a sensor node that can receive a query and return information on the targets registered by the sensor.

Many attempts have been made to develop visual user interfaces for query languages, e.g. for SQL, but only a few have touched the issues of spatial and/or temporal queries and few cases are concerned with sensor data. An approach fairly related to this work is presented in [26], where a method based on case-based reasoning uses input from a scenario and the user interaction is carried out by a query language.

In this paper, we present the query-driven system with several sensor nodes for recognition of targets using various sensors. Our current system include visual and thermal cameras as well as different types of laser sensors. The outline of this paper is as follows. In Section 2 the query-based information system is described, followed, in Section 3, by the sensor data analysis, i.e., target recognition on signal/image processing level. Section 4 describes the sensors, data acquisition, and experimental results. Section 5 explains in detail

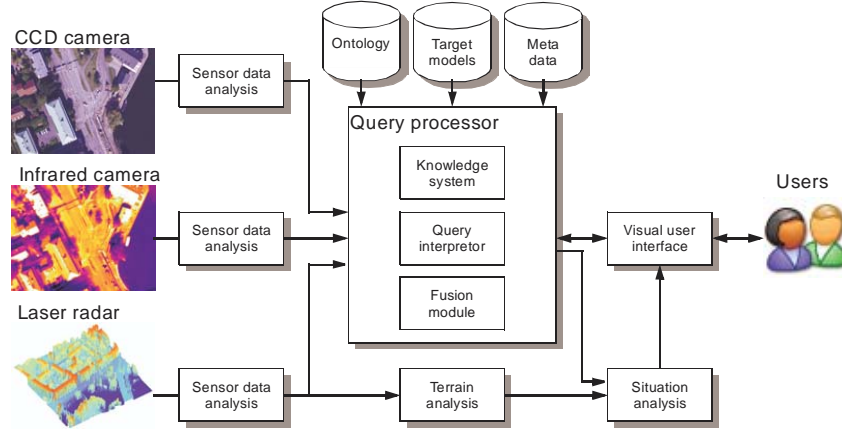


Figure 1: The query-based system's architecture.

the target recognition and data fusion process using numerical values from the experiments. In Section 6, the discussion is found and Section 7 contains our conclusions.

## 2 The Query-Based Information System

A schematic view of the system's architecture is found in Figure 1. The system is, from left to right, divided into the sensor nodes, the query processor and, the visual user interface. The sensor nodes can be distributed across the network. The query processor includes a knowledge system connected to an ontology. The data fusion module is integrated in the query processor. A query interpreter forms the link to the visual user interface. The sensor nodes include sensors and sensor data analysis algorithms for target detection and recognition. This is an open system where it is easy to add/change computational modules. It is a simple, generic computational model, where the data structure can deal with heterogeneous data. Situation analysis, terrain analysis, and user interface fall outside the scope of this paper and are described elsewhere [14, 19, 25, 30].

### 2.1 The Query Execution Process

The core of the computational model is the *query*. A user query can be "Report all terrain vehicles in the specified area for the last two hours". The query is entered using the visual user interface (the area of interest is typically specified by marking an area on a map with a pointing device). The execution of a query, that is, everything performed from the reception of a query (entered by the user) to the presentation of the query result, is performed in a process controlled by the ontology. The query execution process is illustrated in Figure 2. In practice, the query is a data structure with fields for different target attributes, and values are entered in these fields during the execution. The two basic levels in the process are described below, details are in [20].



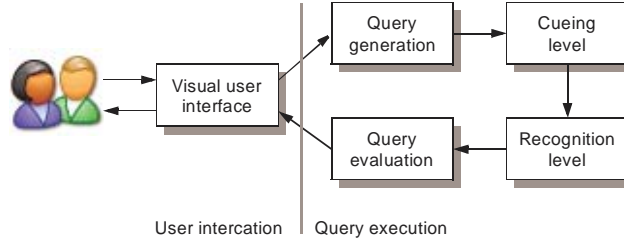


Figure 2: Overview of the query execution process.

The first level is the *cueing level* where the area of interest (AOI) can be large and the time interval of interest (IOI) can be long. Cueing in this sense means finding potential targets and indicating their positions. It is performed by a sensor node including a) a target detection algorithm and b) a sensor with wide-area coverage like an synthetic aperture radar [32]. The output from the cueing level is one query structure for each detected target. Each query contains the position of the detection with an uncertainty interval, thus specifying a new AOI (see Figure 3) combined with the restrictions from the user (such as target type and IOI).

The second level is the *recognition level* where the target recognition takes place. The process includes two major steps: (a) estimation of the attributes of potential targets, and (b), matching of the potential targets to models selected from a target model library. A query received from the cueing level is refined by the attribute estimation to include precise information on the target’s position, dimensions, and orientation. The refined query is then handed to the model matching step, where it is further refined to include target type. Multiple algorithms can perform attribute estimation and model matching, respectively, and data fusion takes place after both steps. The computational model is outlined in Figure 4.

An algorithm for selecting sensors and sensor data analysis algorithms is developed [20]. The algorithm uses the ontological knowledge-base in conjunction with knowledge-base rules to determine which sensors and sensor data analysis algorithms are the most appropriate under the current circumstances. The circumstances are given by the query, the meta data conditions, the external conditions (weather, light), and the terrain background.

## 2.2 The Query Processor

We introduce a query language for sensor data, that can handle various, heterogeneous sensor data types. The query language is called  $\Sigma$ QL [5, 6]. The query processor includes an ontological knowledge system that supports automatic selection of sensors and sensor data algorithms for the data analysis. A database containing a library of target models is attached to the  $\Sigma$ QL-processor, that is used in the target recognition. A meta-database containing the available information that has been registered by the sensors is also attached to the query processor. The query system includes, contrary to conventional query

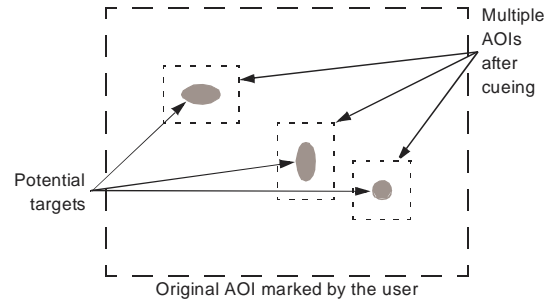


Figure 3: The cueing level, identifying potential targets.

User input	
Query generation	
Query execution	Cueing level
	for all selected sensor/algorithm pairs
	Attribute estimation
Query execution	Recognition level
	for all selected sensor/algorithm pairs
	Attribute estimation
	Attribute fusion
	for all selected sensor/algorithm pairs
	Model matching
	Model fusion
Query evaluation	
Output to user	

Figure 4: The computational model for processing of a user query.

languages, a sensor data fusion module to fuse information extracted from the sensor data. These data emanate generally from multiple sensors whose sensor data altogether are of heterogeneous type. Recent developments of  $\Sigma$ QL allowing more qualified queries are described in [7].

The visual user interface is designed to allow both simple and complex queries of spatial/temporal type. For the time being, complex queries concern vehicles and different types of spatial and temporal relationships that may occur between the vehicles and between vehicles and background information [31]. Background information typically consists of geoinformation. The most important spatial relationships are topological relations, directions and, distances. Queries that allow the combination of spatial and temporal conditions are possible as well. Due to the uncertainties in the sensor data, these relations include uncertainties that must be considered when the system responds to the queries. Conditions in the queries are here expressed visually, contrary to traditional text-oriented query languages where the conditions are included in where-clauses in (often long) text strings.

## 2.3 Data Fusion

There are two fusion processes in the query system. The first is the attribute fusion that considers the attribute set estimates (ASEs), and the second is the model match fusion that considers the results from the model matching algorithms.

The first step of the ASE fusion is to identify clusters of similar ASEs. This is done by identifying sets of ASEs that have common elements in the attribute space. Each cluster is then represented by a single ASE, which is called the prototype of the cluster. The clusters are assumed to correspond to qualitatively different interpretations of the data. For example, some ASEs might claim that the target's orientation is north, while others claim that it is south.

The model match fusion is performed by sorting by the confidence values provided by the model matching algorithms followed by a pruning, keeping only the best match results for each target model. The structure of matching results given by the ASE fusion allows the sorting/pruning to take into consideration different initial interpretations of the image data.

After the attribute estimation, the queries include information on target dimensions, orientations, etc. The next step is to select target models compatible with the attribute estimates of each query. Each query is thus again refined and possibly degenerated into several subqueries, one for each compatible target model. Each of the resulting subqueries then continues to the matching step, to be evaluated against the data. An example is shown in Section 5.

The intermediate fusion step allows information to be shared between the different processes. This potentially increases the overall performance of the system. The intermediate fusion unfortunately also introduces a risk of data incest. Data incest is present when dependent data sources are assumed to be independent. In this application, the risk of data incest is avoided by the quantitative matching [13].

## 2.4 The Simulation Environment

To demonstrate and test the query language and to make it possible to easily integrate new sensor types, it has been integrated with a simulation framework [32]. An important motivation for this is to make it possible to apply dynamic queries over long time periods. The simulation framework includes a scenario generator which makes it possible to create and run complicated scenarios including a large number of events. Input data to the query language comes from sensor models corresponding to instances of sensor types placed on both stationary and dynamic platforms. In this way, queries can be concerned with estimation of both spatial and temporal attributes of objects. For example, vehicles are simulated within the framework and the movements are described in terms of events in the scenario generator. Besides the query language, other types of services can also be attached to the framework and used for various purposes in the same way as the query system.

## 3 Sensor Data Analysis

In this section we describe the sensor data, the target recognition process and the used algorithms. Two algorithms were developed within this project (Sections 3.4 and 3.6), two are modified standard algorithms (Sections 3.3 and 3.5).

### 3.1 Sensor Data

We consider three fundamentally different types of sensor data:

1. 2D images. The samples contain intensity values and are ordered in a rectangular mesh.
2. 3D point scatters. The samples are unordered and each sample contains  $(x, y, z)$  coordinates of the recorded object's surface.
3. Gated viewing (GV) data. A sequence of 2D images where each image contains the reflected laser pulse response at a certain time interval.

In practice, the data sources are cameras, laser scanners, and gated viewing lasers, respectively. Moreover, the 2D images are assumed to be acquired from an airborne down-looking camera that operates in the visual, (reflective) near infrared, or (thermal) long wave infrared band. The GV sensor is assumed to be oriented horizontally. The GV sensor is omitted in this presentation. However, the algorithm for matching on GV data is based on the range template matching algorithm described in [40]. Our application of the algorithm is described in [1].

### 3.2 The Target Recognition Process

The target recognition process (i.e., the recognition level in the query execution process) is performed in four steps; attribute estimation, attribute fusion, model matching and model match fusion, see Figure 4.

Several constraints are applied to the target recognition process in order to keep complexity down. First, it is assumed that detection and coarse localization is already performed at the cueing level. Second, we constrain the dimensions

and orientations of the target. Third, in the model matching step, the number of target models is quite small. These constraints are communicated via the query received from the cueing level.

### 3.2.1 Attribute Estimation

As mentioned, the first step is to estimate the target attributes. The attributes are position, orientation, length, width, height, speed, and temperature. Note that none of the currently available sensor nodes can estimate all these attributes. The attributes should be used as input to the model matching by (a) pointing out which sensor data is to be matched, and (b) reduce the number of possible models.

The input to the attribute estimation is a query specifying the approximate position and size of the target. The position originates from the target detection at the cueing level, and is assumed to be specified with a precision in the order of tens of meters. Thus, an AOI of approximately  $50 \times 50$  meters is considered. It is also assumed that the query specifies an approximate size of the target, originating from the user's request and/or the knowledge base. The default intervals 3–11 and 1.5–5.5 meters are used for length and width respectively, covering all target types we are currently interested in. If the user has specified a certain class of targets, the intervals are narrowed.

The output from the attribute estimation is a refined query, including the estimated attributes and the refined position. Each estimated attribute is given with an uncertainty interval. If multiple sensors nodes operated on the same target, the results are fused before sent to the model matching.

### 3.2.2 Model Matching

In the model matching step the common target model library is used, where each model is described by its 3D structure (facet/wireframe models), appearance (visual or thermal textures), and, in some cases, algorithm specific attributes (e.g., pre-processed imagery, see below). 3D (facet) models of reasonable resolution are commercially available<sup>1</sup>. Detailed models, including thermal signatures, can also be created using 3D lasers scanners, thermal cameras and/or simulations<sup>2</sup>.

Based on the user's query and the estimated attributes, a set of target models are selected for the matching process. The input to each model matching algorithm is thus a query and a model. The output is a match between the model and the sensor data pointed out by the query. Additionally, the query might be further refined by the matching algorithm.

To be able to perform comparison and fusion, the results are transformed into a normalized framework with a confidence value between 0 and 1, where 1 means a perfect match. The normalization is based on matching results of important target data sets. The confidence values from the model matching constitute the output from the sensor data analysis module, and are handed back to the sensor fusion module for a final decision.

This division in attribute estimation and model matching might be suboptimal from a computer vision point of view (compared to combining the two

---

<sup>1</sup><http://www.facet3dmodels.com>

<sup>2</sup>ThermoAnalytics Inc., <http://www.thermoanalytics.com>.

steps). However, it has shown to be an advantage when integrating the full system as it enables intermediate fusion and pruning of the number of hypotheses. Additionally, if the model matching fails (e.g., because the correct model was not present in the model library), the results from the attribute estimation are still available and returned to the user.

### 3.3 Attribute Estimation from 2D Image Data

In 2D images, active shape models (ASMs) [9, 10, 35] provide a technique for estimating contours of objects belonging to a specific class. Here, we use an ASM to find rectangular-shaped objects – the assumption of rectangularity holds well when searching for vehicles, provided that the images are in top view. Since the ASM is a greedy algorithm and needs an initial position, the ASM fitting is preceded by detecting corners and lines using standard detectors from the literature and combining them into rectangles according to the following procedure. The numerical constants were determined by training on synthetic data.

1. Corner detection: On the image data within the ROI, a standard Harris corner detector is applied [18].
2. Rectangle forming: All combinations of three corners forming three corners of a rectangle within the format given by the query are found and stored in a rectangle candidate list.
3. Edge detection: On the image data with the ROI, a standard Canny edge detector is applied [3].
4. Rectangle matching: Each rectangle candidate is matched to the image by integrating, over its four sides, the projection of the gradient onto the side normal, i.e., the match value for each side is

$$m_i = \oint_{\mathbf{x}} |\mathbf{n} \cdot \mathbf{g}(\mathbf{x})| d\mathbf{x},$$

where  $\mathbf{x}$  is the 2D image coordinates varying from one rectangle corner to another,  $n$  is the normal to the side, and  $\mathbf{g}(\mathbf{x})$  is the image gradient at  $\mathbf{x}$ . Each rectangle gets the score  $m = m_1 + m_2 + m_3 + m_4$ , and the 10 best rectangles are returned and used to initialize the active shape models.

5. ASM fitting: An ASM, described below, is fitted to each of the (up to) 10 rectangles. The ASMs also return a quality-of-fit measure, and the best one is selected.

Since we already after step 4 have extracted a few rectangular objects and also ranked them, the job could be assumed done. However, in practice neither the ranking nor the estimate of position and size was reliable enough to use as the refined attribute estimate, and thus the ASM is used. The ASM follows the standard recipe from the literature [8, 35] but with some important restrictions and simplifications due to our specific circumstances. First, since we assume rectangular targets, the single deformation mode is the change of length/width ratio (the other free parameters are rotation, scaling, and translation). Second,

we do not train the landmark profiles, instead we assume that we have an edge at the target border, thus basically following [9]. The mean square error between the rectangle and the ASM control points is calculated. The root mean square is considered to be an estimate of the standard deviation of the estimation performance and used in uncertainty interval calculations for the estimated attributes (length, width, orientation).

### 3.4 Attribute Estimation from 3D Scatter Point Data

When the input data consists of 3D point scatters, we can use geometric feature extraction [15]. In this approach we assume that man-made objects, especially vehicles, can be approximated by one or several rectangles. The 3D point scatter describing the target is detected using differentiation of height data [4]. We first study the object in top view and then rotate to side and front/back views (3D data can be rotated to any projection). The 3D size and orientation estimation consists of five steps:

1. Transform data to top view perspective,
2. Estimate a rectangle based on top view data  $(x, y)$ . The main directions of the target are given by the orientation of the rectangle. The yaw angle is given by the orientation of the rectangle's main axis,
3. Project the data set into the direction  $(x', y')$ , where  $x'$  is parallel to the main and  $y'$  is parallel to the secondary axis,
4. Estimate a rectangle based on side view data  $(x', z)$ . The pitch angle is given by the orientation of this rectangle,
5. Estimate a rectangle based on back/front view data  $(y', z)$ . The roll angle is given by this rectangle's orientation.

This rectangle fitting algorithm calculates the rectangle that with minimal area contains the data samples [15]. The rectangle is computed in linear  $O(N)$  time if input data is sorted and in  $O(N \log N)$  time for unsorted data, where  $N$  is the number of samples. Its performance is analyzed in [16]. The uncertainty intervals of the size and orientation estimates are functions of  $N$  and signal-to-noise ratio in data, given by the algorithm's performance.

### 3.5 Model Matching on 2D Image Data

A library of target models with thermal infrared textures is available. To match one of these models to image data, the model is adapted to the image in terms of 3D translation and rotation. For this purpose, generative models and methods are popular in the computer vision community. A generative model can, given a parameter set, generate an image to be compared to the input image (the image that should be analyzed). Since such algorithms are typically greedy, they need a good initial suggestion of the model parameters.

Our approach is based on multiscale Gabor filters in a sparse grid, in some aspects similar to [39]. All models in the target model library have been analyzed by the filter probes, and the outputs are stored in the target model library.

Traditional alignment methods, based on image signal energy, might have problems with thermal fluctuations due to variations in weather conditions and the internal thermal states of the target. Local phase is therefore more suitable for the estimation of a local displacement between a thermal infrared image and a target model. Spatial linear local phase models, often called instantaneous frequency, are estimated from the set of Gabor filters. The target is represented using a sparse grid of such models.

The steps in the alignment method are explained in the following:

1. The ontology indicates what model in the target model library to use for matching, and also the sensor position relative to the target given the platform navigation data. A set of preprocessed local frequency models are chosen, representing the current target hypothesis. Such a model is only capable for target recognition in a limited span of model variation parameters (translation, scaling, target model rotations, and in plane rotation).
2. Assuming approximately level ground and known target size (from the target model), we can initialize the parameters fairly well and perform an initial warping of the AOI in the image toward the target model.
3. The target model grid is placed over the image according to the hypothesis, and the neighborhood of each node in the grid is examined using the Gabor filters. The local motion estimation is performed using regression of the normal flow, exploiting the fact that the difference in spatial position between the image and the target model equals the local phase difference divided by the local frequency.
4. A global 8-parametric displacement model is estimated using an iterative reweighted least mean squares method [12, 38]. The iteration is stopped after a given number of iteration or when the mean displacement update is under a given threshold. The output is a warped image aligned to the target model.
5. Performance evaluation is done by combining four different scalar features extracted from the final iteration. These features are (a) mean normal flow after final alignment, (b) scalar product between normalized filter responses, (c) deviation from equal scale ratio (in width and height) in the target model image, and (d) deviation from right angles in the final warping. Each feature is mapped to the interval [0,1] and a combined confidence value is computed as the geometric mean.

### 3.6 Model Matching on 3D Scatter Point Data

The 3D scatter matching is used to match the sensor data with a 3D model of similar resolution. We assume that a ground target viewed in different projections can be approximated by a set of rectangles and that in some views the rectangles will describe the functional parts of the target. On the other hand, a laser beam does not penetrate dense materials like metal surfaces. Thus, we only collect data from the parts of the object that are visible from the laser radar’s perspective (so-called self-occlusion). Further, in this application we



can neither assume that the target is placed in a certain position relative to the sensor, nor can we assume a certain orientation (or articulation) of the target. The algorithm is described in detail in [15], the main steps are:

1. Segment the target into parts of approximately rectangular shape, the segments are stored in a binary tree. The main parts of the object are stored in (some of) the terminating leaves.
2. Traverse the terminating leaves and search for possible target parts by geometric comparisons. Dimensions of a data subset (stored in a leaf) are compared with the models's dimensions.
3. Match the entire object with a wire-frame model. The model's functional parts are rotated to the estimated orientations. The distance between the target points and the model facets is calculated and the relative mean square error is used to determine the matching score. The relative mean square error is defined as the mean square error normalized with the data variance [2].

## 4 Experiments and Results

A prototype of the information system, its query language, and sensor nodes, has been implemented. The prototype includes:

- A  $\Sigma$ QL query processor. The query processor includes an ontology and its knowledge-base designed to allow sensor data independence from an end-user perspective;
- A data fusion module that fuses sensor data in two steps, i.e., to support 1) attribute estimation and 2) target model matching;
- A visual user interface that allows the application of queries in a sensor data independent way;
- Means for visual presentation of query results;
- A number of algorithms for analysis of sensor data.

### 4.1 Data Acquisition

In order to collect experimental data, seven vehicles were placed in open terrain. Three sensors, one infrared sensor, one scanning laser radar, and one visual CCD-camera, were mounted on an airborne platform. The visual sensor is not described further as visual data is not used in this paper. The platform was flying at 25 meters per second at an altitude of 130 meters above ground. GPS and inertial navigation systems were used for position determination. The seven vehicles, including main battle tanks (MBT), an anti-tank gun (ATG), a truck and a car, were placed in open terrain. Their dimensions and how they were sampled by the sensors are listed in Table 1. Figure 5 illustrate an example of collected data.

Table 1: The targets and acquired data.

Target	Target class	Target type	Dimensions Length $\times$ Width [m]	No. of samples (3D)	Pixel size (LWIR) [m]
A	MBT	T72	$7.13 \times 3.52$	110	0.19
B	ATG	BTR70	$7.42 \times 2.78$	109	0.20
C	Truck	Scania P113	$8.40 \times 2.50$	89	0.20
D	MBT	T72	$7.13 \times 3.52$	184	0.16
E	Car	Volvo 440	$4.45 \times 1.69$	22	n/a
F	MBT	T72	$7.13 \times 3.52$	124	0.21
G	MTB	T72	$7.13 \times 3.52$	144	0.21

The infrared sensor is a long wave IR (LWIR) sensor operating in the band 8.0–9.2 micrometers (thermal infrared). The sensor has a field of view of 15 degrees (240 pixels) along the flight direction and 20 degrees (320 pixels) perpendicular to the flight direction. The image rate is 50 Hz and the precision 14 bits per pixel.

The scanning laser radar operates in the near infrared (NIR) at 1.06 micrometers with 0.1 mJ/pulse and a sampling rate of 7 kHz. The footprint on ground is approximately 0.14 meter and the distance between the samples approximately 0.3 meter along the scanning lines and 0.5 meter between the scanning lines. The field of view is 20 degrees perpendicular to the flight direction. The scanning constitutes a zigzag pattern on the ground and the resulting data is in point scatter format containing 3D position and reflected intensity in each sample, i.e., the data is an unordered set of samples  $(x, y, z, r)$ .

## 4.2 Attribute Estimation on 2D LWIR Data

The attribute estimation using rectangle forming and ASM was evaluated on LWIR images of six of the seven vehicles (unfortunately, the seventh vehicle is not present in the LWIR data set). The pixel size varied between 16 and 21 centimeters due to variation in flight altitude. The attribute estimation was considered a success in five cases and a failure in one (target D). Figure 6 illustrates one of the successful cases. For target D, where the estimation failed, the reason was that the vehicle had been stationary for some time with the engine running. During this time the exhaust plume from the vehicle heated an almost rectangular area on the ground beside the vehicle, making that area clearly visible in the LWIR. The ASM was fitted to the area being the union of the vehicle and heated ground. Note that the vehicle was a main battle tank with a strong and quite characteristic exhaust plume, not comparable to, for example, a car that hardly could heat the ground in that manner. The estimated dimensions, orientations and the estimated errors are given in Table 2. In all five successful cases the true dimensions are inside the estimated interval.

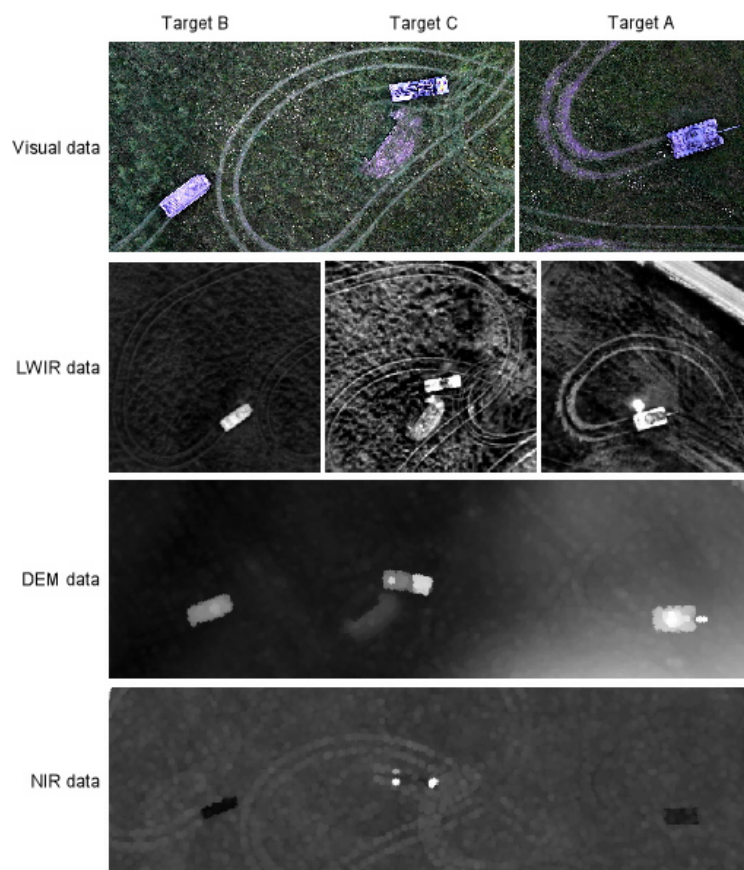


Figure 5: Example of collected vehicle data. From left to right: target B,C, and A. From top to bottom: Visual data, LWIR data, DEM data, and NIR data.

Table 2: Estimates of length, width, and orientation using different sensor data.

Target	Sensor	Estimated	Estimated	dim. [m]	Estimation	error [m]
Class	data	orientation				
Dimensions		[deg.]	Length	Width	Length	Width
A	3D	$316.1 \pm 2.3$	$6.34 \pm 0.82$	$3.49 \pm 1.02$	0.79	0.03
MTB	DEM	$317.1 \pm 4.1$	$6.74 \pm 0.58$	$3.70 \pm 0.58$	0.39	0.18
$7.13 \times 3.52$ m	NIR	$317.2 \pm 4.9$	$7.00 \pm 0.70$	$4.00 \pm 0.70$	0.13	0.48
	LWIR	$314.9 \pm 4.1$	$6.99 \pm 0.56$	$3.92 \pm 0.56$	0.14	0.40
B	3D	$335.2 \pm 2.0$	$7.30 \pm 0.69$	$2.64 \pm 0.83$	0.12	0.14
ATG	DEM	$335.4 \pm 3.1$	$7.45 \pm 0.44$	$2.72 \pm 0.44$	0.03	0.06
$7.42 \times 2.78$ m	NIR	$336.0 \pm 4.3$	$7.65 \pm 0.61$	$2.88 \pm 0.61$	0.23	0.10
	LWIR	$333.4 \pm 3.5$	$7.60 \pm 0.48$	$2.92 \pm 0.48$	0.18	0.14
C	3D	$306.3 \pm 2.0$	$7.64 \pm 0.80$	$2.38 \pm 0.96$	0.76	0.12
Truck	DEM	$305.6 \pm 6.5$	$7.97 \pm 0.92$	$2.58 \pm 0.92$	0.43	0.08
$8.40 \times 2.50$ m	NIR	Fail	Fail	Fail	-	-
	LWIR	$307.0 \pm 3.9$	$7.95 \pm 0.54$	$2.57 \pm 0.54$	0.45	0.07
D	3D	$324.9 \pm 1.9$	$7.30 \pm 0.86$	$4.09 \pm 0.98$	0.17	0.57
MBT	DEM	$316.4 \pm 3.6$	$6.87 \pm 0.51$	$3.77 \pm 0.51$	0.26	0.25
$7.13 \times 3.52$ m	NIR	$316.4 \pm 4.3$	$7.09 \pm 0.61$	$4.09 \pm 0.61$	0.04	0.57
	LWIR	Fail	Fail	Fail	-	-
E	3D	$230.9 \pm 4.6$	$3.52 \pm 0.89$	$1.42 \pm 0.97$	0.93	0.27
Car	DEM	$230.6 \pm 1.5$	$4.46 \pm 0.21$	$1.80 \pm 0.21$	0.01	0.11
$4.45 \times 1.69$ m	NIR	Fail	Fail	Fail	-	-
	LWIR	No data	No data	No data	-	-
F	3D	$317.8 \pm 2.0$	$6.56 \pm 0.69$	$3.51 \pm 0.83$	0.57	0.01
MBT	DEM	$316.5 \pm 4.1$	$6.72 \pm 0.58$	$3.74 \pm 0.58$	0.41	0.22
$7.13 \times 3.52$ m	NIR	$318.6 \pm 5.3$	$7.12 \pm 0.76$	$3.97 \pm 0.76$	0.01	0.45
	LWIR	$316.1 \pm 6.9$	$6.50 \pm 1.02$	$2.96 \pm 1.02$	0.63	0.56
G	3D	$274.6 \pm 2.1$	$7.11 \pm 0.95$	$4.08 \pm 1.10$	0.02	0.56
MBT	DEM	$278.0 \pm 3.6$	$6.73 \pm 0.51$	$3.10 \pm 0.51$	0.40	0.42
$7.13 \times 3.52$ m	NIR	Fail	Fail	Fail	-	-
	LWIR	$271.1 \pm 3.5$	$6.90 \pm 0.53$	$3.89 \pm 0.53$	0.23	0.37

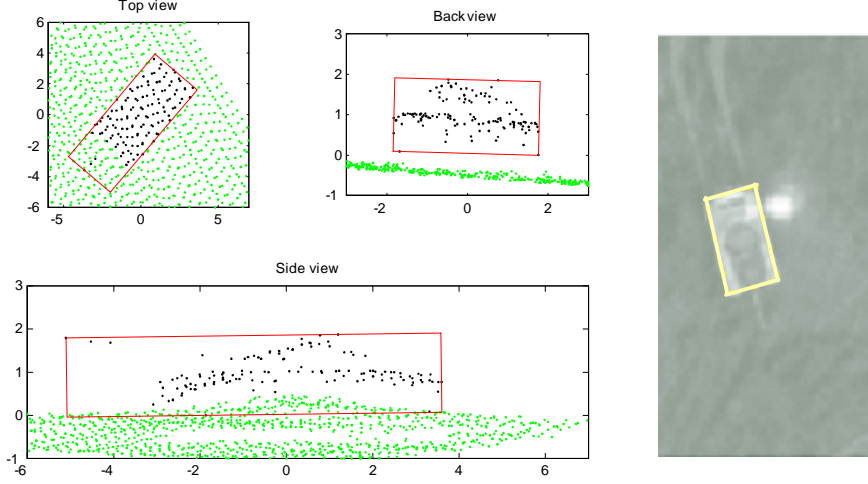


Figure 6: The attribute estimation on target A. Left: Laser radar data, background sample are grey and target samples black. The rectangle shows the estimated 3D size and orientation. Axes in meters. Right: LWIR data, the rectangle shows the estimated length, width and 2D orientation.

### 4.3 Attribute Estimation on 2D DEM Data

The 3D laser scanner data was resampled using normalized convolution [24] to two different 2D image data sets. The first set consists of elevation images, where the pixel at  $(x, y)$  has the intensity value  $I(x, y) = z$ . i.e., a digital elevation map (DEM). The attribute estimation using rectangle forming and ASM was evaluated on the DEMs of all seven vehicles and considered a success in all seven cases. The estimated dimensions and orientations and the estimated errors are given in Table 2. The true errors as well as the estimated errors are comparable to the LWIR case.

### 4.4 Attribute Estimation on 2D NIR Data

The second set of resampled laser scanner data consists of reflective images, where the pixel at  $(x, y)$  has the intensity value  $I(x, y) = r$ , i.e., basically the same images as would have been acquired by a CCD-camera operating at that wavelength (NIR at  $1.06 \mu\text{m}$ ). Note that the properties of LWIR and NIR are very different, since NIR is in the reflective domain (like visual light) while LWIR is in the thermal. By visually inspecting the data (see Figure 5) it is clear that the NIR data is less suited for the used technique, since the object's border can easily be obscured by the object's and the background's internal structures.

The attribute estimation using rectangle forming and ASM was evaluated on the NIR images of all seven vehicles, and considered a success in only four cases. One of the failures (target E) is due to the low very number of on-target samples. When examining target C, the algorithm is confused by the strong reflective markers on the target (the bright spots in Figure 5). The reason for the remaining failure (target F) is yet to be understood. The estimated

dimensions and orientations and the estimated errors are given in Table 2.

#### 4.5 Attribute Estimation on 3D Data

The attribute estimation was performed on the 3D scatter laser radar data using the geometric feature extraction. For all cases the estimation is considered successful, as the true values were within the uncertainty interval. As expected, the best performance was reached for the targets that were placed in the middle of the scanning pattern (aspect angle approximately 0 degrees). Targets E and F were registered in the very borders of the scan where the sampling is even more irregular due to the zigzag pattern. This affected the length estimates. On targets A, D, and G samples on the target’s barrel were detected. For target A the barrel was pointing forward and for targets D and G the barrel was rotated 20 and 40 degrees respectively. The segmentation was rather rough resulting in that barrel samples close to the body were assigned to be part of the target’s body. This affected, although not vastly, the width and orientation estimates for targets D and G.

#### 4.6 Cross-Validation

The attribute estimation methods returned estimates of the length and width that were close to the true values, and, in all cases except one, the errors were within the estimated uncertainty interval. For the orientation, the true values are not known, instead we evaluate the algorithms using cross-validation. For targets A, B, C, E, and F, the different orientation estimates are consistent, see Table 2. For target D, however, the orientation estimate from 3D data differs a few degrees, probably due to the influence of on-barrel samples. For target G, the LWIR estimate differ approximately 5 degrees. The reason is that the LWIR sensor was not firmly fixed to the airborne platform, and its orientation could deviate up to 10 degrees from the laser scanner’s orientation when the airborne platform was changing course. This was confirmed by visually inspecting the data and noticing that a building close to target G have a slightly different orientation in the LWIR data compared to the laser radar data. The cross-validation for targets A, B, C, and F are illustrated in Figure 7.

#### 4.7 Model Matching

The model matching was performed both on 2D LWIR using generative models and 3D laser radar using point scatter matching. In Figures 8-9 the matching of target A (a T72 main battle tank) is shown, and the confidence values are shown in Figure 10. We observe that for both sensor types the correct pair of target-model had the highest confidence, and that the models are ranked in the same order for both sensor nodes. The second most confident model, the T80, is very similar to a T72 in terms of 3D shape, but only has a somewhat similar thermal signature. This exemplifies how some features are easy to discriminate using one sensor, but can hardly be noticed with another, and thus a multi-sensor system gives an advantage.

In Table 3 the confidence values of 3D scatter matching is shown. If the relative mean square error is larger than 0.07 the misfit of target-model is too large and the confidence value is not calculated, this is marked with ‘-’ in the

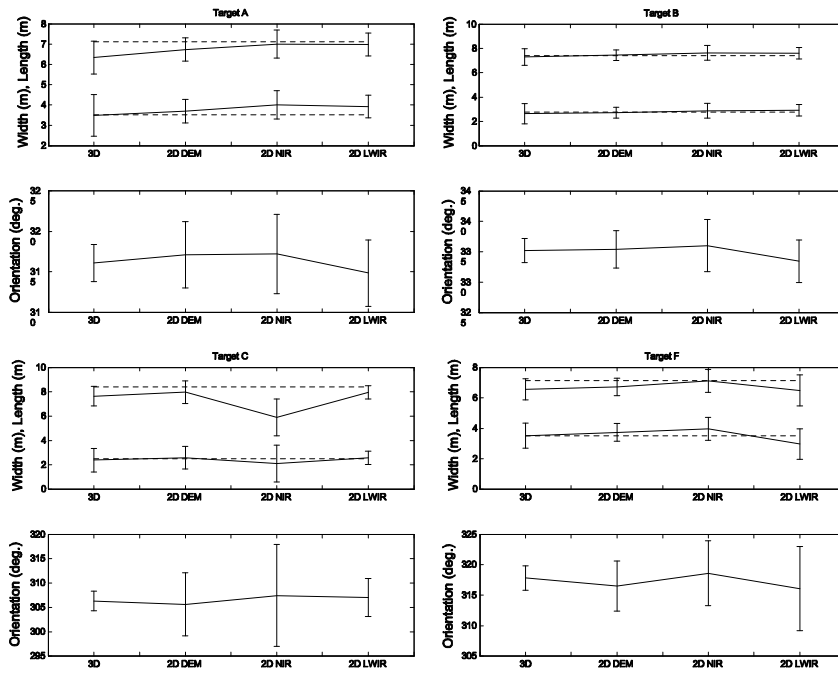


Figure 7: Cross-validation of estimated attributes on targets A, B, C, and F. For length and width estimates, the dashed lines represent the true values.

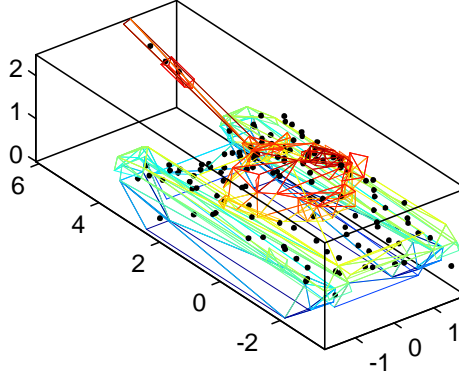


Figure 8: Matching target A with the T72 model, using the 3D scatter matching method. The target samples and the model facets are shown. Axes in meters.

Table 3: Results from 3D scatter matching.

	MBT	MBT	MBT	MBT	MBT	APC	ATG
Target, type	T72	T80	Leclerc	Leopard 2A6	M1A1	BMP1	BTR80
A, T72	<b>0.92</b>	0.91	0.59	0.83	0.87	0.59	0.74
B, BTR70	0.86	0.89	0.89	0.92	0.92	0.80	<b>0.93</b>
C, Truck	0.50	0.16	0.46	<b>0.60</b>	0.32	-	0.32
D, T72	<b>0.93</b>	0.93	0.91	0.81	0.86	-	0.18
E, Volvo 440	0.52	<b>0.85</b>	0.78	0.77	0.74	0.40	0.83
F, T72	0.89	<b>0.92</b>	0.89	0.83	0.89	0.83	0.78
G, T72	0.91	<b>0.93</b>	0.87	0.84	0.90	0.12	0.22
No. of faces	240	1526	462	1359	380	384	683

table. We observe that matching scores for a T72 target with T72 and T80 models are similar, due to similarities in the chassis. The matching is made with a rather small amount of target samples and in most cases few facets on the model, which makes confusion more likely. For targets A, D and G the model's turret and barrel are rotated according to results of the attribute estimation. For target F there are no samples detected on the barrel. This results in similar confidence values in four of five matches with tanks. For two targets, C and E (truck and car, respectively), correct models are not present in the library. For the truck, the confidence values are low for all models, indicating that the correct model is not present. For the car, good matching scores are received for both tanks and anti-tank guns vehicles! This is because there are few samples on that target and these samples are not well distributed. In the future, the data quality should be taken into consideration when computing the confidence value.

Overall the results are acceptable. The T72 tanks are recognized as tanks, most likely a T72 or a T80, and the anti-tank gun vehicle is recognized as well. To be able to more accurately determine the target type (i.e., the type of tank), we need higher resolution on both targets and models.



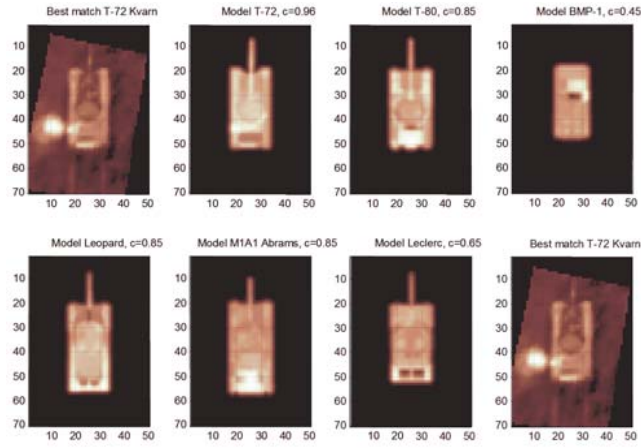


Figure 9: Matching with Gabor probe modelling for LWIR data on target A, confidence values are given.

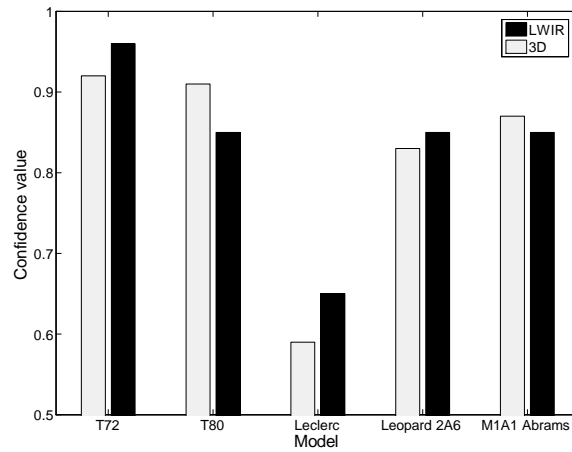


Figure 10: Resulting confidence values when matching target A to five different tank models.

Table 4: The queries entering input to and output from the cueing level. Empty fields omitted (sensor, type, confidence value, height, speed, and temperature).

Field	Input	Target A	Target B	Target C
Class	MBT	MBT	MBT	MBT
Position	(500,500) $\pm$ 500	(560,400) $\pm$ 20	(620,460) $\pm$ 20	(590,440) $\pm$ 20
Orientation	0–360	0–360	0–360	0–360
Length	6.88–7.92	6.88–7.92	6.88–7.92	6.88–7.92
Width	3.40–3.80	3.40–3.80	3.40–3.80	3.40–3.80

## 5 The Execution of a Query: Recognition and Fusion

In this section, we will describe in detail how a query is processed until a final answer is given to the user. Assume that the user enters the following query “Report all main battle tanks (MBTs) present in the one kilometer wide area around the position (500,500)”, where the specified area covers targets A, B, and C. Since all tanks in the target model library are within the dimensions  $7.4 \pm 0.52 \times 3.6 \pm 0.2$  meters, these values are inserted into the query, see Table 4. The query is then sent to the cueing level, which returns three queries, one for each detected target in the area. The three queries are each sent to the four available sensor nodes (3D, DEM, NIR, and LWIR) for attribute estimation. The 11 queries (from the first 12 rows of Table 2) that are returned are then fused into four hypotheses (fusion of two queries are done by computing the intersection), see Table 5.

1. All queries regarding target A are consistent and fused to one hypothesis of a target being approximately  $6.78 \times 3.78$  meters in size, which is consistent with the MBT class. The corresponding attribute estimations on 3D and LWIR data are shown in Figure 6 .
2. All queries regarding target B are consistent and fused to one hypothesis of a target being approximately  $7.50 \times 2.79$  meters in size, which is not consistent with an MBT. The hypothesis is thus discarded.
3. Two queries regarding target C (LWIR and DEM) are consistent and fused to one hypothesis not being consistent with an MBT. The hypothesis is thus discarded.
4. The 3D query regarding target C is not being consistent with an MBT, and is thus discarded.

Thus, only one hypothesis remains, and is entered in the query. All target models consistent with this query are extracted from the target model library, and for each extracted model, a query is sent to the model matching step. In this particular case, two models (representing two different MBTs) are found. Since both LWIR data and 3D data are available, two model matching algorithms are invoked. Additionally, each algorithm is invoked twice, because the orientation is known only up to a 180 degree turn, resulting in a total of 8 queries, see Table 6. Note the reduction in complexity due to the attribute estimate fusion.

Table 5: The queries representing the four hypotheses after attribute fusion.

Field	Target A	Target B	Target C	Target C
Sensor	3D/DEM/ NIR/LWIR	3D/DEM/ NIR/LWIR	3D	DEM/LWIR
Class	MBT	MBT	MBT	MBT
Position	(562.6,399.3) $\pm 0.3$	(619.8,458.8) $\pm 0.3$	(591.1,437.1) $\pm 0.3$	(591.4,437.3) $\pm 0.5$
Orientation	314.8–318.4	333.2–336.9	304.3–308.3	301.1–310.9
Length	6.43–7.16	7.12–7.89	6.84–8.44	7.41–8.49
Width	3.36–4.28	2.44–3.16	1.42–3.34	2.03–3.11

Table 6: The eight queries entering the model matching step.

Field	Target A,	unrotated		
Sensor	3D	LWIR	3D	LWIR
Type	T72	T72	Leclerc	Leclerc
Orientation	314.8–318.4	314.8–318.4	314.8–318.4	314.8–318.4
Field	Target A,	rotated		
Sensor	3D	LWIR	3D	LWIR
Type	T72	T72	Leclerc	Leclerc
Orientation	133.8–138.4	133.8–138.4	133.8–138.4	133.8–138.4

Table 7: The queries representing the two hypotheses after model match fusion.

Field	Hypothesis 1	Hypothesis 2
Sensor	LWIR	LWIR
Class	MBT	MBT
Type	T72	Leclerc
CV	0.96	0.65
Position	(562.6,399.3) $\pm 0.3$	(562.6,399.3) $\pm 0.3$
Orientation	314.8–318.4	314.8–318.4
Length	6.43–7.16	6.43–7.16
Width	3.36–4.28	3.36–4.28

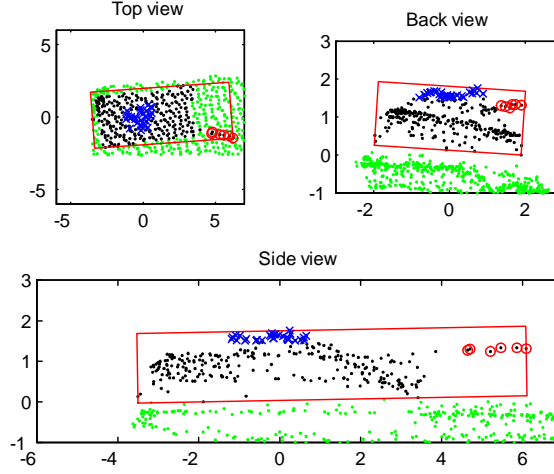


Figure 11: Barrel (o) and turret (x) identification for target F. Note the different barrel positions. Axes in meters.

From the model matching eight queries are returned with confidence values from Figure 10 (the LWIR and 3D model matches are illustrated in Figure 9). The queries are grouped per target type and sorted, and the query with the highest confidence value in each group is kept. In this example, one query for the T72 model and one for the Leclerc model is kept, see Table 7. The one with the highest confidence value is returned to the user (alternatively, a few of the best queries are presented depending on the user's settings). The final answer is that there is one tank present in the area; it is of the type T72 with the confidence value 0.96.

## 6 Discussion

This paper has been concentrated on target recognition of stationary targets in open terrain. There have also been work on target detection based on thermal data [33] and laser radar imagery in more difficult scenes [36].

The sensor data used in this application have high resolution and details on the targets can be seen. This means that articulated parts and deformations of the target can be identified. Approaches for identification of articulated parts, like barrel, hatches, doors, and appearance of fuel-tanks or back-packs have been reported in [15, 28, 34, 37]. An example using [15] is shown in Figure 11. In LWIR data it is possible to detect the engine exhaust plume and trails of a ground vehicle. These features are important indicators of target activity. Furthermore, the direction of the plume differs with the target type and velocity. In Figure 12, the exhaust plume and vehicle trails from a real LWIR image of a T72 is compared to the simulated version in the object model library.

The identification of articulations and deformations can be extended to temporal sequences using the framework presented in [23]. In [23], model-based recognition is used for simultaneous tracking and recognition of targets. By

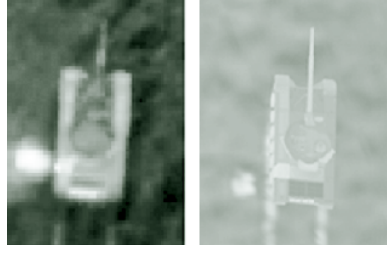


Figure 12: Models for exhaust plume and vehicle tracks are included in the target model library. Left: LWIR image of a target A. Right: Synthesized model with activated exhaust plume and vehicle trails.

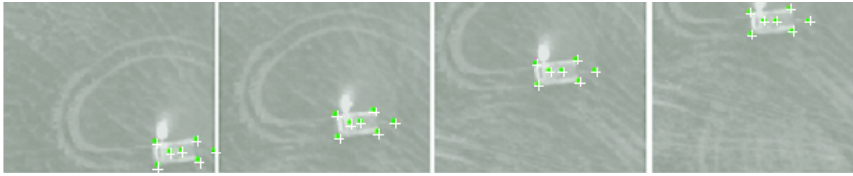


Figure 13: Tracking using model-based reconstruction applied on LWIR data on target A. The marked points on the object are tracked. In this example the sensor platform is moving and the target is still.

tracking a few critical points of the target, e.g., corners and turret, see Figure 13, in a series of consecutive images, its movements and shape variations can be followed through the image sequence. Theoretically, this method should be able to handle both moving targets and moving sensor platforms.

A ground target recognition system must be able to handle (partly) occluded and/or camouflaged targets. The analysis methods must be capable to perform their task even if parts of the target are not registered. Here, the ability to penetrate sparse objects, such as some categories of vegetation and camouflage nets, is a key issue to overcome the problem. Detection of partly hidden objects using laser radar is reported in [11, 21, 36, 37].

## 7 Conclusions

We have demonstrated a query-based system for ground target recognition based on multi-sensor data. The computational model and the sensor data algorithms have been described. By selecting a suite of algorithms for estimation of attributes and model matching in different kind of data, and fusing them on two levels, we have created a system where a user query is refined until a final answer is delivered. We have also demonstrated how the two-level fusion and the division of the target recognition in two steps improve performance and decrease computational complexity.

A field trial has been performed, with an airborne sensor platform flying over a set of vehicles. The sensor platform produced 3D and 2D data in different

wavelengths (visual, near infrared, and long wave infrared). Data sets in form of 3D point scatters, digital elevation maps (DEM), 2D near infrared (NIR) and long wave infrared (LWIR) images were extracted, and all algorithms were applied to all vehicles, with a few exceptions due to lack of data.

Specifically, we propose two attribute estimation algorithms and two model matching algorithms. The attribute estimation algorithm based on 3D data was able to deliver satisfactory results for all targets except for one case where very few and irregular samples were available. The attribute estimation on DEM data was successful for all targets, while the attribute estimation on NIR data failed, as expected, in several cases. The attribute estimation based on LWIR was successful except where heated ground disturbed the thermal signature of the target. Even when the algorithms were not able to deliver satisfactory results, the results were fused into hypotheses consistent with reality in all cases. Model matching algorithms were applied to 3D and LWIR data, giving satisfactory results except for degenerate cases with few and irregular on-target samples.

## References

- [1] P. Andersson, L. Klasén, M. Elmqvist, M. Henriksson, T. Carlsson, and O. Steinvall. Long range gated viewing and applications to automatic target recognition. In *Swedish Symposium on Image Analysis*, pages 89–92, Stockholm, March 2003.
- [2] L. Brieman, J.H. Friedman, R.A. Ohlsen, and C.J. Stone. *Classification and Regression Trees*, chapter 8.3. Monterey Wadsworth and Brooks, 1984.
- [3] J.F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
- [4] C. Carlsson, E. Jungert, C. Leuhusen, D. Letalick, and O. Steinvall. Target detection using data from a terrain profiling laser radar. In *Proceedings of the 3rd International Airborne Remote Sensing Conference and Exhibition*, pages I-431 – I-438, Copenhagen, Denmark, July 1997.
- [5] S.-K. Chang, G. Costagliola, E. Jungert, and F. Orciuoli. Querying distributed multimedia databases data sources in information fusion applications. *IEEE Transactions on Multimedia*, 6(5):687–702, 2004.
- [6] S.-K. Chang and E. Jungert. Principles of visual information retrieval. chapter Query Languages for Multimedia Search, pages 199–217. Springer Verlag, 2001.
- [7] S.K. Chang, E. Jungert, and X. Li. A progressive query language and interactive reasoner for information fusion support. *Journal of Information Fusion*, *accepted for publication*, 2005.
- [8] T.F. Cootes. Homepage, <http://www.isbe.man.ac.uk/~bim/>.
- [9] T.F. Cootes, G.J. Edwards, D.H. Cooper, and J. Graham. Active shape models - ‘smart snakes’. In *Proceedings of the British Machine Vision Conference*, pages 266–275, Leeds, UK, 1992.


- [10] T.F. Cootes, C.J. Taylor, A. Lanitis, D.H. Cooper, and J. Graham. Building and using flexible models incorporating grey-level information. In *Proceedings of the International Conference on Computer Vision*, pages 355–365, Berlin, Germany, 1993.
- [11] C. English, S. Ruel, L. Melo, P. Church, and J. Maheux. Development of a practical 3D automatic target recognition and pose estimation algorithm. In *Proceedings SPIE*, volume 5426, pages 112–123, September 2004.
- [12] G. Farnebäck. *Polynomial Expansion for Orientation and Motion Estimation*. PhD thesis, Linköping Studies in Science and Technology, Dept. of Electrical Engineering, Linköping University, Linköping, Sweden., 2002.
- [13] M. Folkesson, C. Grönwall, and E. Jungert. A fusion approach for coarse-to-fine target recognition. In *Proceedings SPIE*, volume 6242, page 62420H, April 2006.
- [14] J. Fransson and E. Jungert. Towards a query assisted tool for situation assessment. In *Proceedings of the 5th International Conference on Information Fusion*, Annapolis, ML, July 2002.
- [15] C. Grönwall, F. Gustafsson, and M. Millnert. Ground target recognition using rectangle estimation. *Accepted for publication in IEEE Transactions on Image Processing*, 2006.
- [16] C. Grönwall, F. Gustafsson, and M. Millnert. Ground target recognition using rectangle estimation. Technical Report LiTH-ISY-R-2684, Dept. of Electrical Engineering, Linköpings Universitet, Linköping, Sweden, 2006.
- [17] D.L. Hall and J. Llinas (Eds.). *Handbook of Multisensor Data Fusion*. CRC Press, New York, 2001.
- [18] C.G. Harris and M.J. Stephens. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147– 151, Manchester, UK, 1988.
- [19] T. Horney, J. Ahlberg, C. Grönwall, M. Folkesson, K. Silfvervarg, J. Fransson, L. Klasén, E. Jungert, F. Lantz, and M. Ulvklo. An information system for target recognition. In *Proceedings SPIE*, volume 5434, pages 163–175, April 2004.
- [20] T. Horney, E. Jungert, and M. Folkesson. An ontology controlled data fusion process for query language. In *Proceedings of the 6th International Conference on Information Fusion*, Cairns, Australia, July 2003.
- [21] D. Huber, A. Kapuria, R. Donamukkala, and M. Hebert. Parts-based 3D object classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–82 – II–89, June-July 2004.
- [22] E. Jungert, K. Silfvervarg, and T. Horney. Ontology driven sensor independence in a query supported C2-system. In *Proceedings NATO Workshop on Massive Military Data Fusion and Visualization: Users Talk with Developers*, Halden, Norway, sep 2002.

- [23] L. Klasén. *Image Sequence Analysis of Complex Objects – Law Enforcement and Defence Applications*. PhD thesis, Linköping Studies in Science and Technology, Dept. of Electrical Engineering, Linköping University, Linköping, Sweden, 2002.
- [24] H. Knutsson, , and C.-F. Westin. Normalized and differential convolution: Methods for interpolation and filtering of incomplete and uncertain data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 515–523, New York, NY, USA, June 1993.
- [25] F. Lantz and E. Jungert. Determination of terrain features in a terrain model from laser radar data. In *Proceedings of the ISPRS Working Group III/3 Workshop on 3D Reconstruction from Airborne Laserscanner and In-SAR Data*, pages 193–198, Dresden, Germany, 2003.
- [26] P. Louvieris, N. Mashanovich, S. Henderson, G. White, M. Petrou, and R. O’Keefe. Smart decision support system using parsimonious information fusion. In *Proceedings of the International Conference on Information Fusion*, Philadelphia, PA, 2005.
- [27] C. J. Matheus, M. Kokar, and K. Baclawski. A core ontology for situation awareness. In *Proceedings of the 6th International Conference on Information Fusion*, Cairns, Australia, July 2003.
- [28] J. Neulist and W. Armbruster. Segmentation, classification and pose estimation of military vehicles in low resolution laser radar images. In *Proceedings SPIE*, volume 5791, pages 218–225, May 2005.
- [29] J. Nielsen. *Usability Engineering*. Morgan Kaufman, New York, 2001.
- [30] K. Silvervarg and E. Jungert. Aspects of a visual user interface for spatial/temporal queries. In *Proceedings of the Workshop on Visual Language and Computing*, pages 287–293, Miami, FL, USA, 2003.
- [31] K. Silvervarg and E. Jungert. Visual specification of spatial/temporal queries in a sensor data independent information system. In *Proceedings of the 10th International Conference on Distributed Multimedia Systems*, pages 263–268, San Francisco, CA, USA, September 2004.
- [32] K. Silvervarg and E. Jungert. A scenario driven decision support system. In *Proceedings of the 12th International Conference on Distributed Multimedia Systems*, Grand Canyon, AZ, USA, September 2006.
- [33] A. Singh, R.L. Pettersson, J.M. Karlholm, A. Berhdt, and K.E. Brunnstrom. A two-stage approach for target identification and pose estimation in infrared images. In *Proceedings SPIE*, volume 5613, pages 177–188, December 2006.
- [34] E. Sobel, J. Douglas, and G. Ettinger. 3D LADAR ATR based on recognition by parts. In *Proceedings SPIE*, volume 5094, pages 29–40, September 2003.
- [35] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. PWS Publishing, 1998.



- [36] G. Tolt, P. Andersson, T.R. Chevalier, C.A. Grönwall, H. Larsson, and A. Wiklund. Registration and change detection techniques using 3D laser scanner data from natural environments. In *Proceedings SPIE*, volume 6396, page 63960A, October 2006.
- [37] A. Vasile and R.M. Marino. Pose-independent automatic target detection and recognition using 3D LADAR data. In *Proceedings SPIE*, volume 5426, pages 67–83, September 2004.
- [38] A.M. Waxman, B. Kamgar-Parsi, and M. Subbarao. Closed-form solutions to image flow equations for 3D structure and motion. *International Journal of Computer Vision*, 1(3):239–258, 1987.
- [39] X. Wu and B. Bhanu. Gabor wavelet representation for 3-d object recognition. *IEEE Transactions of Image Processing*, 6(1):47–64, 1997.
- [40] Q. Zheng, S.Z. Der, and H.I. Mahmoud. Model-based target recognition in pulsed ladar imagery. *IEEE Transactions on Image Processing*, 10(4):565–572, 2001.



		<b>Avdelning, Institution</b> Division, Department  Division of Automatic Control Department of Electrical Engineering		<b>Datum</b> Date  2006-10-16	
<b>Språk</b> Language  <input type="checkbox"/> Svenska/Swedish <input checked="" type="checkbox"/> Engelska/English  <input type="checkbox"/> _____		<b>Rapporttyp</b> Report category  <input type="checkbox"/> Licentiatavhandling <input type="checkbox"/> Examensarbete <input type="checkbox"/> C-uppsats <input type="checkbox"/> D-uppsats <input checked="" type="checkbox"/> Övrig rapport <input type="checkbox"/> _____		<b>ISBN</b> _____ <b>ISRN</b> _____ <b>Serietitel och serienummer</b> <b>ISSN</b> Title of series, numbering      1400-3902	
<b>URL för elektronisk version</b>  <a href="http://www.control.isy.liu.se">http://www.control.isy.liu.se</a>		LiTH-ISY-R-2748			
<b>Titel</b> Title  Ground Target Recognition in a Query-Based Multi-Sensor Information System					
<b>Författare</b> Author  Jörgen Ahlberg, Martin Folkesson, Christina Grönwall, Tobias Horney, Erland Jungert, Lena Klasén, Morgan Ulvklo					
<b>Sammanfattning</b> Abstract  <p>We present a system covering the complete process for automatic ground target recognition, from sensor data to the user interface, i.e., from low level image processing to high level situation analysis. The system is based on a query language and a query processor, and includes target detection, target recognition, data fusion, presentation and situation analysis. This paper focuses on target recognition and its interaction with the query processor. The target recognition is executed in sensor nodes, each containing a sensor and the corresponding signal/image processing algorithms. New sensors and algorithms are easily added to the system. The processing of sensor data is performed in two steps; attribute estimation and matching. First, several attributes, like orientation and dimensions, are estimated from the (unknown but detected) targets. These estimates are used to select the models of interest in a matching step, where the target is matched with a number of target models. Several methods and sensor data types are used in both steps, and data is fused after each step. Experiments have been performed using sensor data from laser radar, thermal and visual cameras. Promising results are reported, demonstrating the capabilities of the target recognition algorithms, the advantages of the two-level data fusion and the query-based system.</p>					
<b>Nyckelord</b> Keywords Multi-sensor fusion, query languages, infrared sensors, laser radar, range data, target recognition, target detection.					

