

CAD-Based Training of an Expert System and a Hidden Markov Model for Obstacle Detection in an Industrial Robot Environment

Knut B. Kaldestad * Geir Hovland * David A. Anisi **

* *Department of Engineering, University of Agder, Grimstad, Norway*

E-mail: {knut.b.kaldestad, geir.hovland}@uia.no

** *Department of Technology & Innovation, Div. of Process Automation - Oil, Gas & Petrochemicals, ABB, Oslo, Norway*

E-mail: david.anisi@no.abb.com

Abstract: Deploying industrial robots in harsh outdoor environments require additional functionalities not currently provided. For instance, movement of standard industrial robots are pre-programmed to avoid collision. In dynamic and less structured environments, however, the need for online detection and avoidance of unmodelled objects arises. This paper focus on online obstacle detection using a laser sensor by proposing three different approaches, namely a CAD-based Expert System (ES) and two probabilistic methods based on a Hidden Markov Model (HMM) which requires observation based training. In addition, this paper contributes by providing a comparison between the CAD-based ES and the two versions of the HMM, one trained with real sensor data, and one where virtual sensor data has been extracted from the CAD-model and used during the training phase.

Keywords: Obstacle Detection, Industrial Robots, Expert System, Hidden Markov Model.

1. INTRODUCTION

Industrial robots have been used since the 1960s for solving repetitive, routine, heavy and dangerous tasks, such as coating, painting, pick and place, welding, assembly and inspection (Nof, 1999). The traditional industrial robot works in structured, indoor environments and does not stop its process unless its safety switch circuit is broken or it's stopped by the operator. Standard industrial robots are pre-programmed such that the robot path avoids any obstacles in its vicinity, and does therefore not need to know or have any awareness of these objects' locations.

Historically, the main driver for using robots within manufacturing industries has been to achieve better quality and productivity by increased automation. In most industries, this is still true today. Recently, however, the need for deploying industrial robots in rather unstructured outdoor environment has arisen. Within the oil and gas industry, for instance, (Anisi et al., 2010, 2011) the applications generally stand out from other industries as the main driver has been to automate tasks that have been difficult or even impossible for people to undertake based on Health Safety and Environment (HSE) issues. Applying robotics in this way has resulted in an improvement in HSE, but often with an associated dip in production. Although this is contradictory to the general goal of automation, work is now being done towards maintaining focus on HSE and at the same time improving the efficiency and profitability of the facilities.

Today, industrial robots are not able to work independently in harsh and unstructured outdoor environments, which involves detection and avoidance of unmodeled objects. The work presented in this paper takes a step in that direction by focusing on online obstacle detection using two different approaches. The first approach is a CAD-based Expert System (ES). The ES has initial knowledge about the environment, and therefore knows what objects are allowed and at which positions they are allowed. The second approach is using a Hidden Markov Model (HMM). This is a probabilistic method which relies on training of observation of the environment. Alternative methods could, for example, be Bayesian decision networks and support vector machines. The alternative methods are however beyond the scope of this paper, hence, the focus is on performance and limitations of the ES and HMM methods.

Related work describing the combination of online obstacle detection and a laser rangefinder in an industrial robotics environment are hard to find. Most of the related research has focused on mobile robots and different applications such as localisation and mapping. In research described by Wolf et al. (2005), the authors investigate a method for mobile robots to detect the state of the terrain. The robots are equipped with a laser range scanner and able to produce a map in 3D when driving forward. To be able to determine whether parts of the scanned area belong to one of two states – flat terrain or rougher terrain – the authors use an HMM approach. The state estimation is based on comparison of successive scans. The approach in Wolf

et al. (2005) differs from this paper in a few ways. The approach in this paper focuses on obstacle detection and not mapping. In addition, the presented method requires only one scan to determine the state, a larger number of states is used, and the probability of the observations of many models are calculated and evaluated.

The Little Helper described by Hvilshøj et al. (2009) is an industrial robot mounted on a mobile robot platform and provides another point of reference. The project purpose is to devise and develop an industrially usable mobile robot concept. The robot operates in a semi-structured indoor environment, where it picks up objects that are placed at different positions and are located by vision sensors. One of the outlined scenarios: A work station calls on the robot, the robot moves to the workstation and performs a manipulation task with the robot arm. When the task is complete, the robot releases the task and moves away from the work station. This approach is similar to the approach described in this paper in the way of handling environments that are not fully structured, and using a industrial robotic arm for manipulation tasks. The work presented in this paper goes beyond the one conducted for the Little helper by focusing on online obstacle detection and in particular comparing CAD-model-based ES with HMM.

The work presented in this paper is an extension of the work described in (Kaldestad et al., 2012). The main extension is the inclusion of a CAD-file to define the ES. The HMM is then trained on generated data from the ES, which are indirectly generated from the CAD-file. If accurate maps of the environment are available, the presented method eliminates the need for measurement based training of the HMM.

The remaining of this paper is organized as follows: Section 2 presents the problem formulation, Section 3 explains the ES while Section 4 introduces the two HMM approaches. Finally, Section 5 presents the experimental results and Section 6 concludes the work.

2. PROBLEM FORMULATION

In industrial environments, the robot movement is typically restricted to its pre-programmed trajectories in combination with static or temporary world zones preventing the robot's tool center point (TCP) to either leave or enter the manually defined world zones. However, as the demand for handling more dynamic environments increases, so does the demand for planning trajectories dynamically. To this end, this paper focuses on the problem of on-line detection of unknown and unmodeled objects which constitutes the initial part of dynamical trajectory planning.

The laboratory setup depicted in Fig. 1 gives a schematical overview of the robot cell. The robot (C) is manipulating objects on the work bench (I) with a tool located in the tool holder (J). Objects (A) and (H) are static objects in the robot cell and together with the wall (D) these are a part of the CAD-file map. Furthermore, objects (F) and (G) are objects that are allowed in the robot cell at these positions and are part of the CAD-file map that includes objects. The laser (B) continuously scans 0° – 180° in the x-y plane at a height z. While manipulating, robot movement

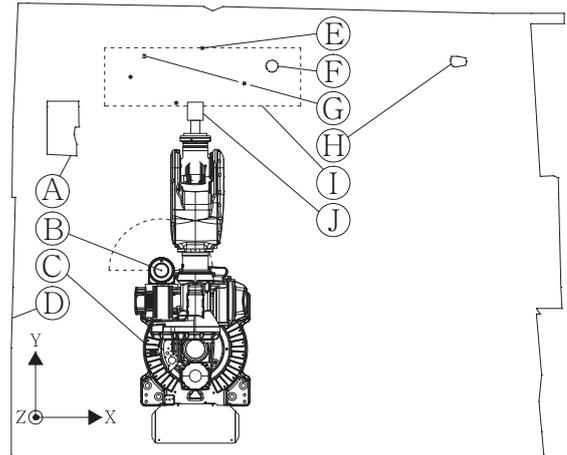


Fig. 1. A schematic overview of the robot cell

will rotate and translate the laser which is mounted on the robot's first axis. The four black dots on the workbench such as (E), are the positions where laser data have been collected for the work described in this paper.

In the work described in this paper, two different approaches for on-line obstacle detection will be investigated, where the Expert System will be described in the following section.

3. EXPERT SYSTEM

The Expert System (ES) described in this paper relies upon a description of the environment (map and objects) in terms of a CAD-model. When a CAD-model of the environment exists, the distances from the centre of the laser sensor to any shape described by the model can be calculated using geometrical relations as will be detailed below. Then by comparing this to the distances measured on-line by the laser, it is possible to detect unmodeled obstacles entering the robot workspace. However, a straightforward comparison between the measured and the modeled distance would lead to an unacceptably high rate of false object detection alarms. To remedy this behaviour, systematic sensor measurement errors (± 30 mm), as well as inaccuracies in the provided CAD-model will be taken into account before triggering alarms.

More precisely, with an estimated CAD-model accuracy of ± 10 mm, the laser measurements are allowed to deviate within the threshold of $\varepsilon = \pm 10$ mm) before triggering the alarm indicating obstacle detection. Despite this threshold, the cases when the laser measurements deviate more than ε is observed typically when the laser beam hits near the edge of an object. In this paper, such measurements are recognized by being considerably greater than that of the modeled distance to the object. This non-predictable behaviour of the laser measurements, was handled by extending the functionality of the ES in accordance with the bottom two boxes in Fig. 2, which shows the overall program design.

As previously mentioned, the ES needs to calculate the modeled distances from the laser centre to the environment described by the CAD-model. For this purpose, the orientation of the robot's first axis (upon which the laser sensor is mounted) and the position and orientation of the

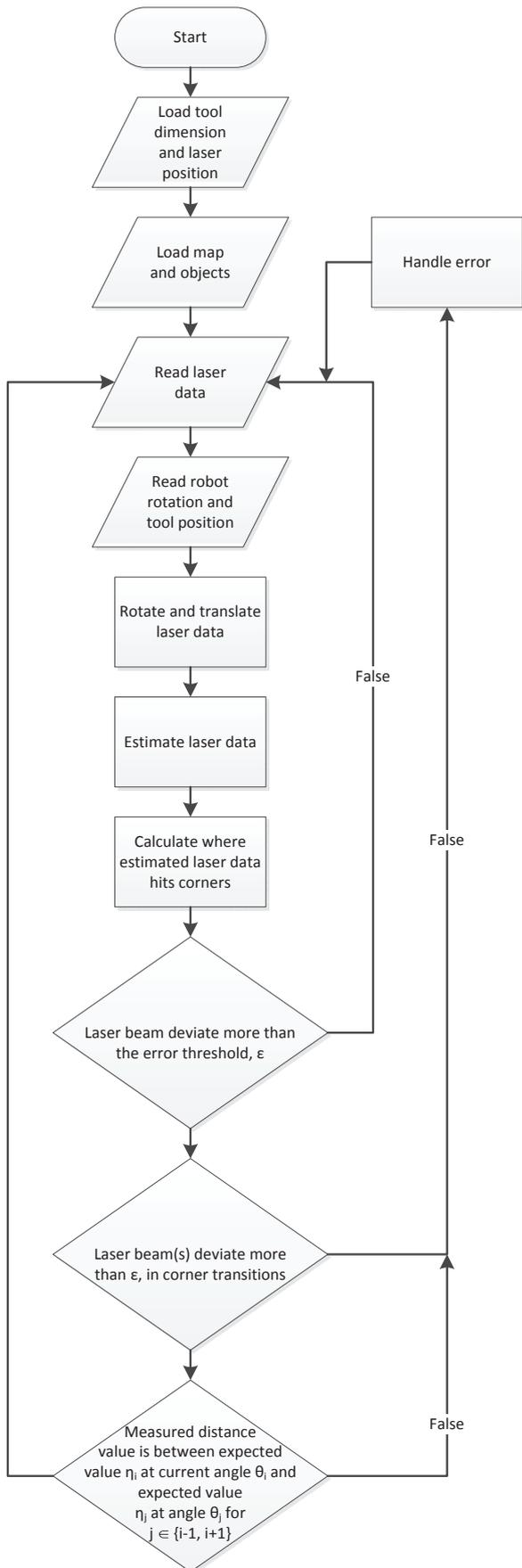


Fig. 2. The ES program design.



Fig. 3. FARO laser tracker used to calibrate the position of the laser scanner relative to the base of the robot.

robot tool need to be known. This data is readily available on most industrial robot controllers of today. However, the ES also requires knowledge about the position of the laser sensor relative to the robot base coordinate system. In practice, this quantity is most often not known or measurable directly with sufficient accuracy. Therefore, finding the position of the laser relative to the robot base is performed as follows.

To allow accurate calculation of the relative position between the laser scanner and the robot base, a FARO laser tracker was utilized. It measures points in 3D space with a worst case accuracy of $18 \mu\text{m} + 3 \mu\text{m}/\text{m}$. As an example, the accuracy 5 m out from the device will be $18 \mu\text{m} + 3 \mu\text{m}/\text{m} \times 5 \text{m} = 33 \mu\text{m}$. The laser tracker was positioned in the front-right of the robot (see Fig. 3), and a local coordinate system was created, with x-axis and y-axis direction in accordance with Fig. 1, and z-axis perpendicular to the x-y-plane to define a right-handed coordinate system. Next, a laser tracker target which reflects the laser beam was centrally aligned on top of the laser sensor. Then, the first axis of the robot was rotated at angles θ_1 and θ_2 degrees. Measurements were conducted by the FARO laser tracker at each of the points, enabling us to extract the radius defining a circle centered at the origin of the robot base coordinate system, and passing through the measurement points. As shown in Fig. 1, this radius equals the distance between the laser to the centre of the robot. The angle offset from the centre of the robot to the laser was found by maximizing the y-distance; this is when the laser position will be 90° relative to the robot centre. At the point where y was maximized, the angle rotation of the robot's first axis was read from the robot controller. This equals the negative angle offset of the laser.

Having calculated the exact position of the laser sensor in the environment, the distance from the sensor to the closest point in the environment is calculated as follows. The CAD-model describing the environment (map and

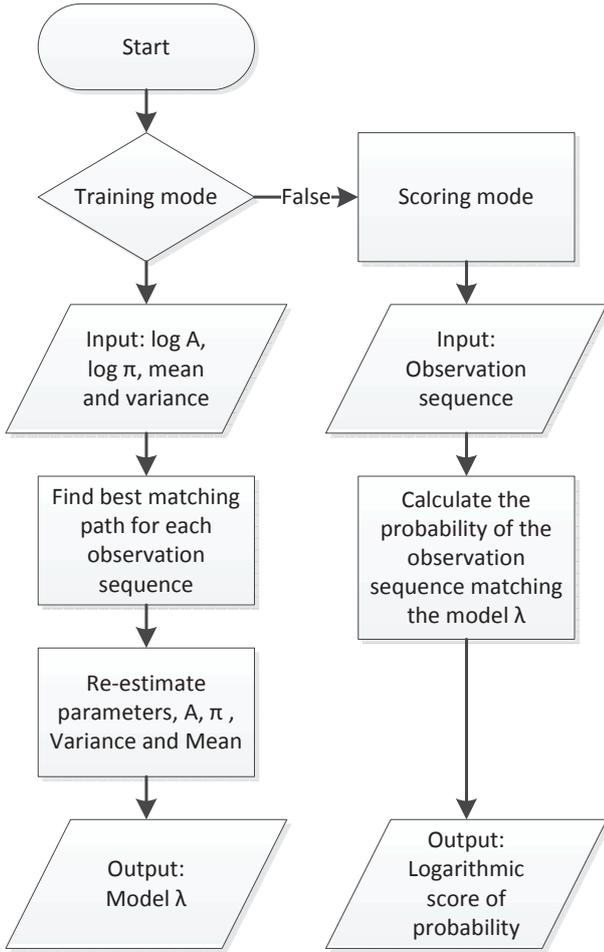


Fig. 4. Flowchart of the HMM training and scoring algorithm.

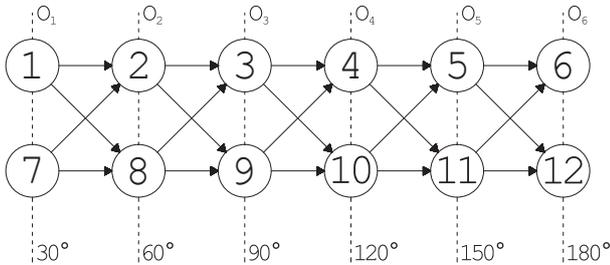


Fig. 5. Illustration of the HMM transition matrix, \mathbf{A}

obstacles) consists of a number of corner points associated with each object. Each two consecutive points belonging to the same object, are then used to define a line. Then, for each ray centred at the laser position and defined by the angle $\theta \in [0, 180]$ degrees from the x-axis, the intersection point between the ray and all these line segments are calculated. To finally arrive at the distance to the closest point in the environment, the intersection with the lowest length value is recognized. The method is repeated until a closest intersection for all angles in $\{0, 0.5, \dots, 180\}$ are extracted.

4. HIDDEN MARKOV MODEL

The following notation will be in accordance with Rabiner (1989). The HMM approach provides a confidence value

for laser length data matching the model $\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi})$. Here, \mathbf{A} is a 12×12 transition matrix (see Fig. 5), \mathbf{B} is the observation matrix consisting of the mean and variance – both of size 12×61 . Finally $\boldsymbol{\pi}$ is the 12×1 initial state distribution vector. The observation matrix \mathbf{B} contains six sets of measured data, one for each 30° segment. The arrows in Fig. 5 show the allowed state transitions.

Two different sensor training data sets are used to estimate the HMM, λ ; one uses measured laser data, the other uses estimated (virtual) data extracted from the CAD-model (cf. Fig. 2). The second approach will be an advantage if real sensor data is not available before the system is implemented. It would not only allow for smart scheduling of the computation time in advance of system deployment, but it will also allow the system to instantly begin operating without making numerous measurement of the normal situation where the robot is located. To be able to represent laser measurements, noise that is similar to the standard deviation of the laser sensor is applied to the virtual data. Further, the signal mean and variance are calculated for use in the \mathbf{B} matrix.

5. EXPERIMENTAL RESULTS

5.1 Expert system

The ES, being an extension of the one presented in Kaldestad et al. (2012), has increased complexity by adding a map of the environment where previously a filter was used to filter out anything but the tool. If any unfiltered object entered the area, an error would be thrown. As a result of adding a map, the system is more error sensitive. This sensitivity is especially evident when the laser beam hits in a corner location of an object. This situation often leads to a wrong distance measurement by the laser. There is currently no known method of accurately correct for these situations and as described earlier, beams in corner regions are filtered out. In the experiments the filtering angle θ is 1° , and the systematic error of the system, η , is set to 40 mm. Fig. 6 shows the ES detecting an object that is not a part of the map, and the system responds by throwing an error.

Table 1 shows the results of the 12 test cases in the experimental studies with the ES. It is notable that the ES has classified all scenarios correctly in all four positions. The “no object” and “objects” columns constitute the normal scenarios, i.e., cases when no unexpected obstacle is present. The objects in question are (G) and (F) from Fig. 1. The “objects + obstacle” column represents the abnormal situation when an arbitrary obstacle is placed in an arbitrary location in the robot cell.

Table 1. Expert system performed successfully in all four test positions, in total 12 test cases.

Test set	No object	Objects	Objects + obstacle
Pos 1, λ_1	✓	✓	✓
Pos 2, λ_2	✓	✓	✓
Pos 3, λ_3	✓	✓	✓
Pos 4, λ_4	✓	✓	✓

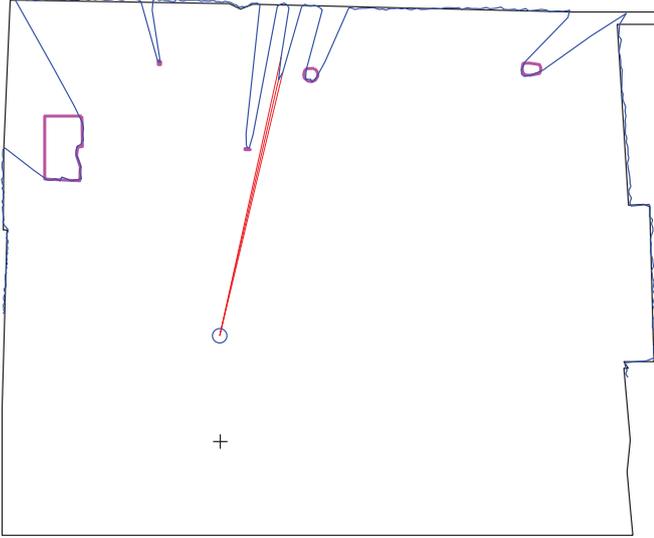


Fig. 6. The blue line is reflections from a laser measurement, an object which is not a part of the CAD-model is detected and is represented by red beams.

5.2 Hidden Markov Models

The HMM results can be found in Table 2–5 where each table presents the scores based on twelve different observations at one of the four positions marked (E) in Fig. 1. From the tables, the log score for the “no object” and “objects” scenarios, which constitute the normal training cases, are quite similar in all the four tables. As for the abnormal “objects + obstacle” case goes, since the obstacle is not a part of the model λ , it is expected that a model score with lower probability is expected. The results show that the score for “objects + obstacle” is significantly higher than the trained models (“no object” and “objects”).

Table 2. Scores $-\log[P(\mathbf{O}_{i,j}|\lambda_1)]$ with 12 test sets $i \in \{1, \dots, 4\}$ and $j \in \{1, \dots, 3\}$.

Test set	No object	Objects	Objects + obstacle
1	1868	1862	2005
2	1867	1862	2005
3	1868	1863	2005
4	1868	1862	2006

Table 3. Scores $-\log[P(\mathbf{O}_{i,j}|\lambda_2)]$ with 12 test sets $i \in \{1, \dots, 4\}$ and $j \in \{1, \dots, 3\}$.

Test set	No object	Objects	Objects + obstacle
1	1909	1912	2046
2	1909	1912	2048
3	1909	1912	2046
4	1909	1912	2047

Table 4. Scores $-\log[P(\mathbf{O}_{i,j}|\lambda_3)]$ with 12 test sets $i \in \{1, \dots, 4\}$ and $j \in \{1, \dots, 3\}$.

Test set	No object	Objects	Objects + obstacle
1	1870	1889	2066
2	1870	1890	2066
3	1870	1890	2069
4	1870	1889	2067

Table 5. Scores $-\log[P(\mathbf{O}_{i,j}|\lambda_4)]$ with 12 test sets $i \in \{1, \dots, 4\}$ and $j \in \{1, \dots, 3\}$.

Test set	No object	Objects	Objects + obstacle
1	1865	1868	2672
2	1864	1869	2664
3	1865	1869	2678
4	1865	1868	2665

For the virtual data to be able to function as a means for training the HMM, the parameter systematic error must be tuned. Fig. 7 shows that the laser measurements are not spread uniformly across the systematic error of ± 30 mm. The measurements are rather taking a normal distributive shape. Table 6 shows the standard deviation of the four different measurements presented in Fig. 7.

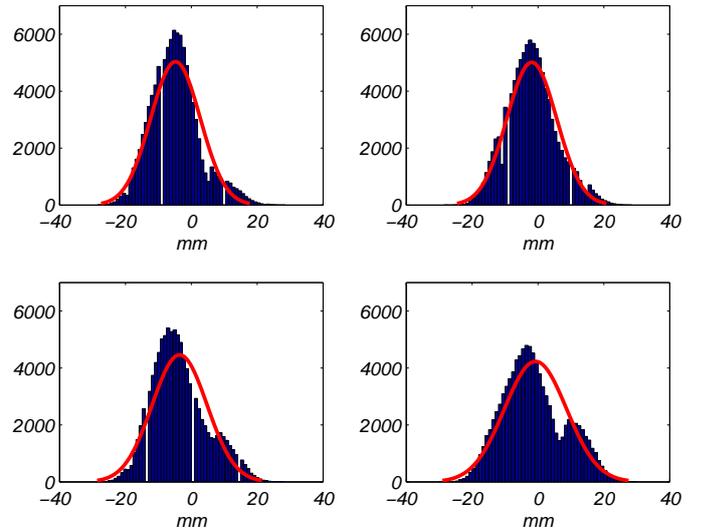


Fig. 7. Systematic error of the laser, for four arbitrary angles. Each histogram is created from 100 000 laser measurements.

Table 6. Standard deviations from Fig. 7.

Upper left	Upper right	Lower left	Lower right
7.527	7.562	8.346	9.4244

The algorithm for training the HMM creates random normal distributed values for the estimated measurement. The reason for the random variables is because the HMM training must have a variety of observations to perform the training on. The effect of changing the systematic error is shown in Fig. 8. The top part of the figure shows that by increasing the systematic error the probability of a HMM trained on virtual data increases for a laser observation with object in the robot cell, while it is more or less steady for a virtual data observation. The top part of the figure also shows that the difference between a laser observation with object and a laser observation with object and obstacle gets smaller as the value of the systematic error increases. This difference is clearer in the bottom part of the figure, and is shown by the red line. It is important to have this difference sufficiently high, as this difference tells how well an accepted HMM score (no obstacle in the robot cell) can be differentiated from an HMM score with obstacle in the robot cell. The black vertical line shows the chosen systematic error at 8.55 mm, which corresponds to the mean of the standard deviation from Table 6.

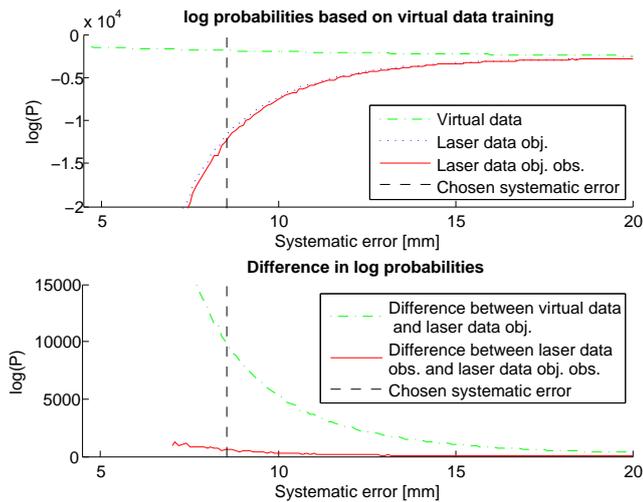


Fig. 8. Model score versus systematic error.

Table 7. Scores based on virtual data $-\log[P(\mathbf{O}_{1,j}|\lambda_k)]$ with 20 test sets $j \in \{1, \dots, 3\}$ and $k \in \{1, \dots, 4\}$.

Test set	No object	Objects	Objects + obstacle
λ_1 (VD)	1881	1880	
λ_1 (LD)	11646	11930	12451
λ_2 (VD)	1883	1882	
λ_2 (LD)	11472	12110	12530
λ_3 (VD)	1882	1883	
λ_3 (LD)	11121	11689	12573
λ_4 (VD)	1885	1885	
λ_4 (LD)	11751	12239	12427

Table 7 shows the probability scores for virtual data and laser data for the models λ_k that are trained from virtual data. The result shows that it is possible to distinguish the two allowed situations (“no object” and “objects”) from a situation where an obstacle is present.

6. CONCLUSIONS AND FUTURE WORK

The ES algorithm is computationally efficient, and is suitable in real time applications. Furthermore, it could provide the system with coordinate location of the obstacle.

The ES is highly dependent on accurate input values (such as a map and the position of the robot), and there are many of these values that can be improved. First, the map of the environment could be measured more accurately by use of, e.g., the FARO laser tracker seen in Fig. 3. Second, a small misalignment of the sensor could have large impact on the measurements, in particular at long distances. This might cause false classification, especially in the CAD-based ES.

An improvement of the problem experienced when measuring corners could be to incorporate a second laser, positioned at another location on the robot.

The possibility of training the HMM directly from the virtual data is an advantage; time can be saved and

there is no need for acquiring large amounts of data after deployment of the robot. It is possible to calculate the HMMs in advance, enabling the system to be up and running at the time of deployment. One limitation of the HMM is the computational requirements for the training and in the current version of the algorithm, it could not provide information of the obstacle location.

There are apparently some challenges with applying a CAD-based map of an unstructured environment, one is how to update a map of an environment that continuously changes. The idea behind the approach presented in this paper, is that large parts of the map will be static. Some parts, however, will be dynamic, but the update frequency of the environment will be so low that on-line updates of the map would be feasible. Dynamic mapping is, however, beyond the scope of this paper and would be a direction of future research.

7. ACKNOWLEDGEMENTS

Knut Berg Kaldestad acknowledges funding from ABB and the Norwegian Research Council through project number 193411/S60.

REFERENCES

- Anisi, D., Gunnar, J., Lillehagen, T., and Skourup, C. (2010). Robot Automation in Oil and Gas Facilities: Indoor and Onsite Demonstrations. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 4729–4734. Taipei, Taiwan. doi:10.1109/IROS.2010.5649281.
- Anisi, D., Persson, E., Heyer, C., and Skourup, C. (2011). Real-World Demonstration of Sensor-Based Robotic Automation in Oil & Gas Facilities. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*. San Francisco, California. doi:10.1109/IROS.2011.6094440.
- Hvilshøj, M., Bøgh, S., Madsen, O., and Kristiansen, M. (2009). The mobile robot little helper: Concepts, ideas and working principles. In *Emerging Technologies Factory Automation, 2009. ETFA 2009. IEEE Conference on*, 1–4. doi:10.1109/ETFA.2009.5347251.
- Kaldestad, K., Hovland, G., and Anisi, D. (2012). Obstacle Detection in an Unstructured Industrial Robotic System: Comparison of Hidden Markov Model and Expert System. In *10th IFAC Intl. Symposiums on Robot Control, Submitted*. Dubrovnik, Croatia.
- Nof, S. (1999). *Handbook of industrial robotics*. Number v. 1 in Electrical and electronic engineering. John Wiley.
- Rabiner, L. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2), 257–286. doi:10.1109/5.18626.
- Wolf, D., Sukhatme, G., Fox, D., and Burgard, W. (2005). Autonomous Terrain Mapping and Classification Using Hidden Markov Models. In *Proc. 2005 IEEE Intl. Conf. Robotics and Automation*, 2026–2031. doi:10.1109/ROBOT.2005.1570411.