Machine Vision and Applications (2003) 14: 15–25
Digital Object Identifier (DOI) 10.1007/s00138-002-0098-6

Machine Vision and
Applications

A system to navigate a robot into a ship structure

Markus Vincze[1], Minu Ayromlou[1], Carlos Beltran[3], Antonios Gasteratos[3], Simon Hoffgaard[2], Ole Madsen[2], Wolfgang Ponweiser[1], Michael Zillich[1]

[1] Automation and Control Institute ACIN, Vienna University of Technology, Gusshausstrasse 27–29/361, 1040 Vienna, Austria; e-mail: vincze@acin.tuwien.ac.at
[2] Aalborg University, Department of Production, 9220 Aalborg, Denmark
[3] Laboratory for Integrated Advanced Robotics, LIRA, University of Genova, 16145 Genova, Italy

Abstract. A prototype system has been built to navigate a walking robot into a ship structure. The 8-legged robot is equipped with an active stereo head. From the CAD-model of the ship good view points are selected, such that the head can look at locations with sufficient edge features, which are extracted automatically for each view. The pose of the robot is estimated from the features detected by two vision approaches. One approach searches in stereo images for junctions and measures the 3-D position. The other method uses monocular image and tracks 2-D edge features. Robust tracking is achieved with a method of edge projected integration of cues (EPIC). Two inclinometres are used to stabilise the head while the robot moves. The results of the final demonstration to navigate the robot within centimetre accuracy are given.

Keywords: Navigation – Ship building application – Computer vision – Model-based tracking

1 Introduction

Robot navigation is a common problem in mobile robotics. In most cases, it is considered a 2-D problem. The sensor data is projected to the ground plane and then used for path planning and robot control. The task of navigating a climbing robot into a ship structure requires 3-D navigation, since the robot will be also able to climb walls.

The main motivation for this project is the business demand of the end user Odense Steel Shipyard (OSS), Denmark, who are looking for a robotic operator that, ultimately, can replace human workers performing welding and inspection tasks for ship construction. Of particular interest is the ability to execute the final welding task at the dock, where conditions for the human workers are exhausting and dangerous. Weather and working conditions are hard along the year and the rate of accidents is high. In [14] a specific robot was constructed to step on the stiffeners. However, it could not navigate; hence, it required manual control.

The objective of the RobVision project was to develop a system for navigating and positioning a robotic vehicle in the body of a large vessel during production. The specific technical goal is to develop a vision system that finds and tracks the robot location relative to the 3-D structure of the ship, with respect to a CAD-model provided by the ship manufacturer. In the next step of development, the robot will in the next step be equipped to deliver work packages for inspection, welding and other tasks.

Correspondence to: M. Vincze

This paper focuses on the RobVision system aspects: the conclusions that led to the design of this system, the key performance issues (reliability of sensing and accuracy of pose), and the lessons learned and how to improve the system towards industrial usage. The paper proceeds by reviewing related work. Section 2 presents the system requirements, and Sect. 3 gives a system overview. Then the main components are outlined: feature extraction (Sect. 4), control of the head and 3-D feature measurement (Sect. 5), and tracking 2-D features and pose estimation (Sect. 6). Section 7 presents the results of the demonstrations, and Sect. 8 presents the lessons learned.

2 Related work

The work of this project is related to navigating mobile robots in indoor environments and having the robots grasp parts Most systems rely on laser range sensors or sonic sensors and navigate in 2-D (e.g. [11]). For 3-D navigation approaches could be used that hold a 3-D CAD map of the building and use landmarks, such as walls or pillars, for navigation, e.g. [10,18,19, 22]. The robot assumes a rough position and matches the landmarks of its map to those detect ed by the vision system. The main problems are changing backgrounds and high computational demands. For example, a space application where the background is dark and the object consists of parts of different surface characteristics requires dedicated hardware to run at a frame rate [29]. Probably the most successful system that uses vision to control a mechanism is the automatic-car and air-vehicle approach using dynamic vision [12]. It integrates the dynamic aspects of a continuously operating system and image data to update the model description of the world.

Another series of techniques that can be used for 3-D navigation relate to object recognition. Object recognition matches image features to features in a database of multiple objects [15,30]. The match reports object hypotheses, which are subsequently verified to report the most likely object. As a by-product of this process, most approaches report an estimate of the object pose. Impressive results have been shown using edge features, e.g. [5,9,15,?]. However, object recognition suffers from two common problems. (1) Matching requires extensive search and does not scale to operate in real time for 3-D objects of reasonable complexity [6]. Newest results on using indexing [4,5] still require several seconds in simple cases and minutes in more complex images. Therefore, most approaches are not used for navigation. An exception is a notable work that realises fast indexing by exploiting image and stereo lines, thou gh the authors concede the "reliability bottleneck" introduced by using one type of feature [10]. (2) The recognition rates are high, under the assumption of good feature extraction. Invariant (to perspective distortion [30] or to illumination [1]) features enable robust recognition; however, this requires a solution to the equally difficult problem of robust feature segmentation.

While, in the above works, geometric features, such as lines, are extracted from the image for pose estimation, the approach in [17] uses individual control points along object edges. The control points are searched normal to the edge direction and then used for pose estimation. This work is improved [25] by adding a median filter to detect outliers for fitting the line and the pose. They report the improvements using black-and-white objects with little clutter in the background. In [28], this approach is extended with an image-processing method that enables the tracking of realistic objects in front of a cluttered background, using cue integration.

Regarding reliable feature extraction, cue integration has been found to be a feasible technique [7]. An approach studied most closely is voting in cue integration. Voting is a model-free approach and requires a common classification space. Plurality voting gives the best results when using four simple blob trackers [23]. In [20], the authors show that weighted consensus voting of five cues for view-based tracking performs better than a fuzzy-fusion method and the single cues. The approach in [3] uses voting to integrate four cues to find planar surfaces but requires a good initial start segmentation to give good results.

3 Technical requirements

The ship-building process requires reducing the time in the dock, which is the production bottleneck. In particular, welding and inspection are very time consuming. To automate the welding and inspection task, the following scenario has been proposed: A walking robot is autonomously navigated through the ship structure. Using sensors, it can estimate its 3-D pose while moving. 3-D information is needed, because the robot has to step over T-trusses, which produce a rigid structure, and because it has to climb walls to reach welds at the ceiling. For the same reasons, a walking and climbing robot is needed. To automate the task, the idea is to utilise the existing CAD model of the ship for task specification.

The RobVision project presented a demonstrator to enable autonomous robots in ship building. The approach is flexible, because it can operate in any environment that has distinct features and that is modelled or that can be modelled. In more detail, the requirements for the system are the following:

- Manual specification: The task must be specified manually, if possible, off-line and using the CAD model.
- Reliable navigation: The navigation must ensure that the robot is not lost in the structure. The initialisation should be possible from a coarsely known starting location and should take seconds. The robot can be fixed during this time.
- Accuracy: The robot must be placed anywhere in the entire structure within ± 10 cm. The seam-following sensor for welding will perform the final adjustment to reach the welding accuracy required. However, a target goal of ± 1 cm has been set, so that the accuracy of the structure's production can be evaluated and reported.
- Operation at robot-walking speed: A velocity of 3 cm/s requires an update rate of 0.1 s to give sufficient feedback for controlling the robot motion. Hence, tracking should operate at three frame cycles ($= 120$ ms).
- Autonomous behaviour: The selection of specific behaviours (initialisation, tracking, welding) should be done according to the situation and without user interference.
- Automated operation: The navigation should be done automatically, using the manual task specification of the path, including intermediate target poses. This requires automation of the process of extracting an adequate portion of the CAD model as a reference for the sensing system.

The main goal was to achieve the 3-D navigation task and to obtain robust visual tracking. Robustness was tackled by developing a method of robust visual finding and tracking by integrating redundant low-level image cues and high-level object knowledge. For the extraction of basic visual cues, independent and complementary modules were designed (see Sects. 5 and 6).

4 System overview

4.1 Specifying the task

OSS has developed an off-line path-planning program, called PathPlanner, to plan the motion of the walking robot through a ship's structure. Figure 1 (and the top left corner of Fig. 2) show an example of a typical mock-up section and the path specification. The operator interacts with PathPlanner to specify intermediary points (IPs), which describe the desired robot path. Each robot position is defined by a pose, i.e. position (X, Y, Z) and orientation (roll, pitch, yaw), a tolerance, and a gait. The tolerance specifies the required accuracy that the robot controller needs to achieve before changing focus to the next position in the path. The gait is the required walking mode for the robot to pass the local part of the path (for example, a high gait to step over a truss, or a narrow gait to step out of a hole in the mock-up). PathPlanner is now operated by TriVision (www.TriVision.dk).

4.2 System architecture

Figure 2 shows the arrangement of the system components for robot navigation. The task of CAD to Vision (C2V), developed by AAU, is to take the user-defined path, to extract

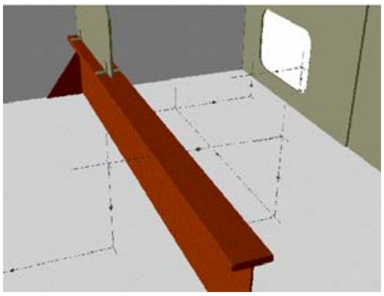


Fig. 1. View of PathPlanner for off-line specification of the robot welding and inspection task

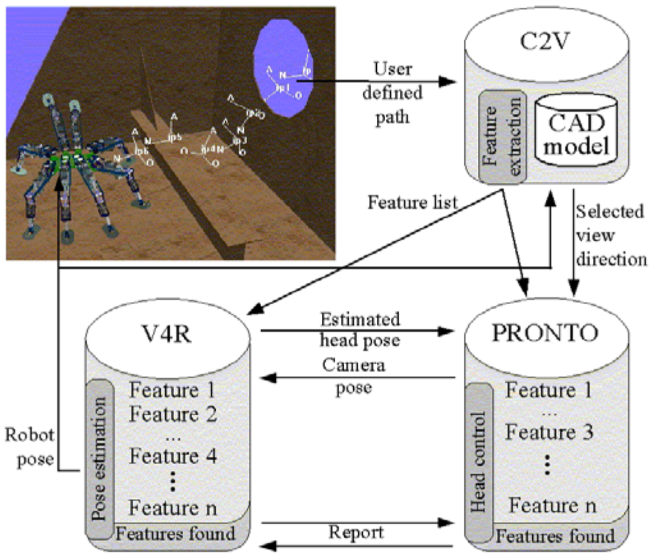


Fig. 2. Principal approach of the RobVision project indicating the main functions of the system components C2V (view generation and feature extraction), PRONTO (head control and 3-D feature finding), and V4R (2-D feature tracking, pose estimation). In the *top left picture*, the white coordinate systems are the intermediate target poses defined by the user. The trajectory between these intermediate target poses is calculated automatically by V4R. The robot uses this information and the robot-pose message of the V4R system to stay on the trajectory

features that are visible, and to send these features to the two vision systems PRONTO (developed by LIRA) and Vision for Robotics (V4R), developed by the Institute of Flexible Automation (ACIN). Hence, C2V needs to select good views that contain many features. The goal is to automatically select features that are expected to be robust and can be used by the vision systems to reliably calculate pose.

Using the view direction, PRONTO controls the head to look at the specified directions. Figure 3 shows the head mounted on the robot. PRONTO also searches for junction features and measures the 3-D position.

The features list is also used by V4R to find 2-D line, junction, and ellipse-arc features. The two vision systems mutually report about features found to increase the reliability of finding and tracking features. Finally, at each tracking cycle, V4R estimates the pose of the head and the pose of the robot, with respect to the coordinate system of the ship. Knowing its present pose from the message of V4R and the path described

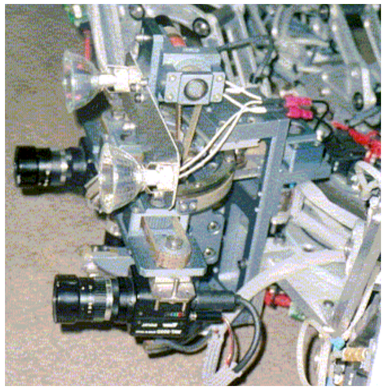


Fig. 3. A close-up of the EuroHead mounted on the robot front

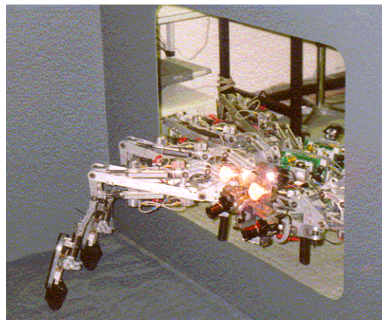


Fig. 4. The Robug IV robot entering the mock-up

by the IP, the robot can calculate its trajectory and can traverse to the next IP and, eventually, to the final target pose.

The robot Robug IV has been developed by Portech, Portsmouth, UK. The robot can carry a weight of 50 kg and is actuated pneumatically. The body of the robot is designed to carry a robot arm for the welding or inspection task. Figure 4 shows the robot at the entrance hole of the mock-up.

4.3 Simulating and integrating communication

The stable communication between several components is of utmost importance for systems integration. A communication tool has been developed that provides the capability to simulate the components, to test each component individually, and to conduct hardware tests in the loop.

The core of the communication tool is a supervisor process running on a computer with a Windows operating system. The components need to be specified only once. For each component, it is then possible to run a simulation on the actual system, which can both run on the same or another computer. The communication protocol utilises the TCP/IP standard. Therefore, the components can run on any operating system. The system components are detected automatically; therefore, no reconfiguration is needed when switching from the simulated component to the actual component. This communication tool has been successfully applied within RobVision to test the communication of individual components and to enable rapid integration of the components. It has been used with Ethernet and Fibre Channel.

Fibre channel has been used to synchronise the images for the two vision systems. Using fibre channel, one image is transferred in about 4 ms. Hence, it can be assured that both vision systems operate on the same image.

4.4 Discussion of system approach

Before describing the system components in more detail, we give the reasons for selecting this setup. After a study of walking robots (CLAWAR thematic network www.uwe.ac.uk/clawar/), the Portech robot was found to be the most advanced system in terms of walking abilities and payload capacity in relation to its own weight. However, it turned out that the pneumatic actuation is a disadvantage, since it introduces jerks in the body motion.

While mounting many cameras on the body was an option, it was decided to use an active-head approach. The main advantage of this approach is that the head can compensate online for the angular part of the irregular robot body motion.

The CAD model is a huge source of information. With the active head, it is possible to optimise viewpoint selection. This ability adds to reliability and makes C2V adaptive and flexible in other environments and situations. The principles for choosing good view points are given in Sect. 4.

There were two reasons for using the two-vision system. The practical reason was that the partners brought considerable experience in different techniques, and the integration onto one platform (either Windows or Linux) was going be a large burden. The technical reason was that these systems presented complementary functionalities. First, Pronto is primarily used for initialisation, while V4R is used for tracking. Pronto needs several seconds, and V4R operates in 120-ms tracking cycles. For a reliable pose estimation, the integration of many features gives better accuracy, as will be shown in Sect. 8. Pronto measures junctions in 3-D, using the stereo head, while V4R measures junction, line, and ellipse features in one 2-D image. Pronto is too slow for tracking. Fast model-based tracking in V4R works only in one image, which uses less computing power while still enabling full 6-D pose estimation.

5 Feature extraction: CAD to vision

The basic idea of C2V is to select good view points, to send new view points along the path, and to provide model and feature information to the vision systems, which then find features in the images and determine pose from the features found. Sending and supervising the present viewpoint operates at a rate of 1 update per second.

The C2V component automatically extracts distinct features from the object model to enable visual navigation. It also automatically evaluates the quality of view points to look at for the vision system. The CAD subsystem concentrates on determining the reference features, which are defined as a coherent set of an intermediate point, the corresponding robust features that should be seen in a view, and a robot gait. The features are geometric references determined from the CAD model of the workpiece, i.e. surface boundaries and intersections represented as lines, circles, half circles, junctions (intersections of edges), regions, etc. By robust, we mean features that will not be confused by the vision subsystem when viewed from the specific intermediate point or that are too small to be significant. C2V consists of three major systems:

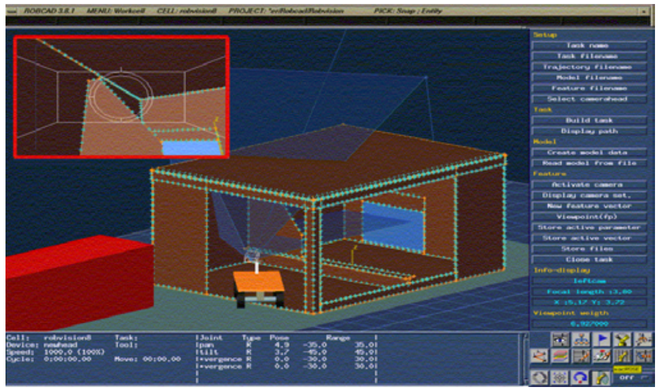


Fig. 5. Example of mock-up in the C2VoffLine system. For part of the demonstration, a cart was used because the robot was not always available. The *top left* is the view as seen from the left camera of the stereo head. From this view, features are extracted automatically

- C2VoffLine (RobCad): C2VoffLine is the part of the AAU system that simulates the motion of the robot and cameras from the path created by the Odense shipyard. During the simulation of the robot movements, the features that the camera will see during the execution of the task are collected (Fig. 5 gives an example desktop view). To make the system available to a large number of customers, a simplified version, C2VoffLine (Windows NT), has been developed.
- C2VoffLine (Windows NT): The kernel (C2Vkernel) used in the RobCad version is implemented in an NT-based version of C2VoffLine. The NT version does not have all the functionality or the same degree of automation found in the RobCad solution, but this version is fully capable of creating features for demonstrations and can be operated by any user.
- C2VonLine: C2VonLine is the specially designed communication software of C2V that communicates with the entire RobVision network at the demanded rate. The software is used to send models and features generated by C2VoffLine, depending on the present pose of the robot.

The loop between the vision subsystem and the CAD subsystem will then be as follows:

1. After a small robot motion, from the knowledge of the former pose and the direction of the movement, the vision subsystem predicts the robot pose, which is also passed on to the CAD subsystem.
2. If the view has changed considerably due to the robot movement, the CAD subsystem provides new 3-D geometric features detected in the CAD model for these poses, using the present view of the camera(s).
3. If the features or images are recognised by the vision subsystem, the view direction is kept. The robot moves towards its target. Go to 2 and continue the loop from there.
4. If the features or images are not recognised by the vision subsystem, the cameras move to another viewpoint. This viewpoint is suggested by the CAD subsystem evaluating areas that are rich with features to render the task of the vision system easier.

5. Go to 3 and continue the loop from there.

The procedure described above and the use of the model is generic, which enables feature extraction and tracking for any object that is represented by a CAD model. Details on the method used to extract features can be found in [8].

6 Head control and 3-D feature measurement: PRONTO

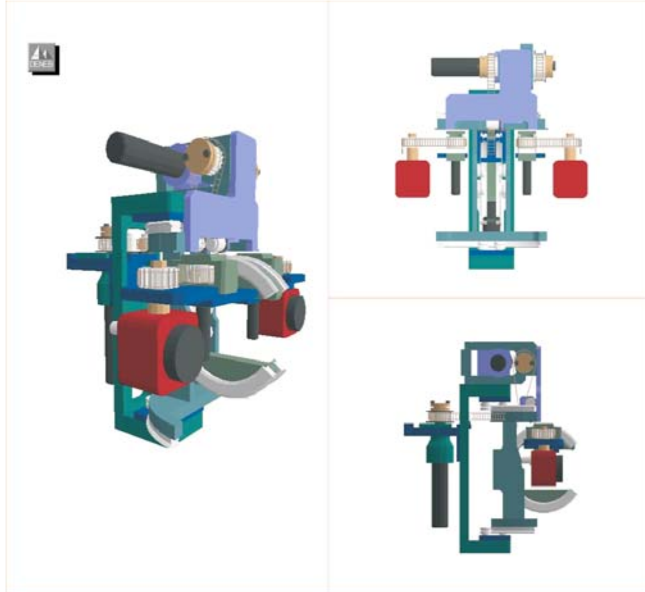
The PRONTO system was developed by partner DIST and fulfils several tasks, which are, in brief:

- head control and calibration
- inclinometre sensor data acquisition, distribution to other systems, and use for head stabilisation
- image acquisition and delivery to other systems and image processing
- communications and synchronization with V4R vision system

PRONTO, therefore, performs a large number of complex and interrelated tasks that need to be synchronized. For these reasons, the software is implemented in C++ as a multi-thread object-oriented distributed application, with the aid of the distributed-programming technology, distributed component object model (DCOM), to create a software architecture that tries to simplify the complexity of this subsystem. From the hardware point of view, the DIST system consists of a computer with two processors running Windows NT, PRONTO and controlling the EuroHead (the head is shown in Fig. 2 above).

Figure 6 shows the CAD model of the head and gives its main specifications. The accuracy of the EuroHead has been evaluated in detail in [13]. A maximum error of 1 cm for a point measurement at 1.1 m distance has been found.

The point-measurement task is solved by finding and measuring the 3-D junctions with a Hough technique for extracting the lines on the image planes of a stereo pair, using the features from C2V. The extracted lines and the junctions are related to the CAD model, using a weighted least-mean-squares method. Then a closed-loop method follows, such that by simultaneously moving the 3 degrees of freedom of the head, the junction is fixed at the principal point of the image in both images. When this is the case, the two cameras are verging on the certain junction, and the direct kinematics of the head are applied in order to determine the 3-D position of the junction relative to the head.



Axis	Range	Velocity	Acceleration	Resolution
Pan	$\pm 45°$	$73°/s$	$1600°/s^2$	$0.007°$
Tilt	$\pm 60°$	$73°/s$	$2100°/s^2$	$0.007°$
Vergence	$\pm 45°$	$330°/s$	$5100°/s^2$	$0.03°$

Fig. 6. CAD-model of the EuroHead and its specifications (see Fig. 3 for a picture)

7 2-D feature tracking and pose estimation: V4R

The task of the V4R system is to extract features from the cues of images and relate them to the features provided by C2V. C2V provides geometric features, such as line, arc, junctions, region, and attributes connected to these features. The attributes of a line can be, for example, welded or not welded, chamfered, or rounded. Regions can have attributes such as intensity, texture, or colour.

The V4R software package is a well-tested prototype, which is available to interested researchers from the authors. V4R provides a tool for tracking, using images from video (mpeg), life camera, or image sequences [28]. V4R contains two major components, which can be exploited separately: (1) framework for tracking of features and (2) pose estimation using a model and the features found in the image.

The tracking tool within V4R is capable of following line, junction and ellipse features at field rate. The tracking method is edge based and uses an EPIC scheme to obtain robustness against varying background and continuous illumination changes [26]. The goal of tracking is to be able to follow fast motions. Therefore, a fast cycle rate and a windowing approach have been adopted, using the results of the formal derivations regarding the dynamics of the robot-vision system [21]. ACIN's entire vision system is designed in C++ and presents a generic structure for any model-based vision method [28]. Its development pays tribute to the development of the XVision system by Greg Hager [16].

The pose-estimation tool of V4R uses the object model and the features found in the image to determine an optimal pose. The following features are utilised for pose estimation: line, 2-D point (junction), 3-D point, and surface normal. Outliers are detected, and a least-squares procedure over all remaining features gives a best pose fit.

Figure 7 shows the windows projected into the image. Section 8 shows tracking results.

7.1 Performance evaluation of 2-D feature-finding methods

To track the edges (lines, ellipses), the EPIC method for cue integration , proposed in [26], is used. EPIC uses cues (edge, intensity, colour, texture, etc.) to distinguish object edgels from background edgels. However, the initialisation of the feature

Fig. 7. Tracking windows to search for line features

cannot use this information, and it is, therefore, the most critical step.

For first finding the edge feature, several methods can be used. This section introduces a comparison of methods for finding the edges. During the project, the following methods were evaluated when the model is first projected into the image. One reference method for the evaluation is tracking, where previous information can be taken into account. This is simpler than finding the feature for the first time. The second reference only uses the edge information from a classical edge extractor. For all methods, the warped-image approach of [16] has been used. The methods are:

1. only-edge: edge finding using only an edge filter, as in [16], and least-squares fit to find a line
2. tracking: using information from a simulated previous finding
3. LMedS: using zero crossings to find edgels and a least-median-square regression to fit the line
4. EPIC-centre: integration using the centre location as most likely source to indicate edgels

The methods have been evaluated on real images of characteristic ship sections. Figure 8 shows a typical example. The detection of features has been tested by moving the images in the horizontal and vertical axes. This renders the initialisation inaccurate by a know range covering ± 30 pixels to the correct centre position.

The results of tests of the four algorithms are summarised in Fig. 9. The detection rate gives the percentage of lines found of all lines initialised. Each data point corresponds to 128 lines in two dozen images. Figure 9 also shows the percentage of the detected lines that have been detected correctly. It shows that the only-edge method detects many lines but also many incorrect lines. The LMedS improves the performance, but adding the centre information is better, even if the displacement becomes larger.

A significant problem is the false detection of features due to ambiguities in the image. The introduction of probabilistic likelihood values to evaluate the quality of a line did not prove successful. Lines were found with many edgels along the line. Therefore, other measures have to be taken. The basic approach is to integrate the topological knowledge from the CAD model.

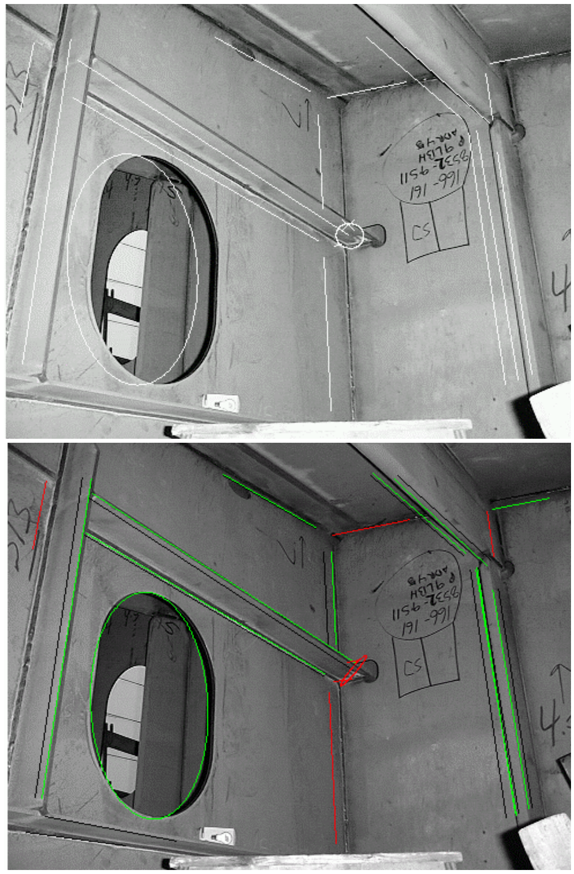


Fig. 8. Initialisation (*top*) and detection results (*bottom*) when horizontally displacing the image of the ship section in 10-pixel steps. The EPIC-centre algorithm is used in this example

Double lines cause severe problems, due to ambiguity. Displacements of junctions on both image planes, due to either wrong estimation of the robot position or bad construction, also cause problems. On the other hand, exactly this information can be exploited to discriminate features: Parallel lines and lines intersecting at a junction must show some specific properties. The use of this topological information is referred to as validation. It helps to disambiguate local ambiguities. This has been implemented and evaluated in detail in [28]. The reader is referred to this article, while this paper focuses on the system aspects. Using the validation, it is possible to robustly track complex scenes, as given here, with a standard Pentium PC 500 MHz at frame rate. In the project, only 120 ms have been used to run several other tools to improve the system performance. Frame or field rate is achieved in standalone applications.

8 Demonstration results

The final demonstration of the RobVision project brought together all of the tools to demonstrate the integrated system at at Odense Shipyard. The goal was to highlight some of the system aspects, and to address the claim that redundant features improve the *accuracy of pose estimation*. Results of an evaluation are shown here and will be followed by a discussion of head stabilisation.

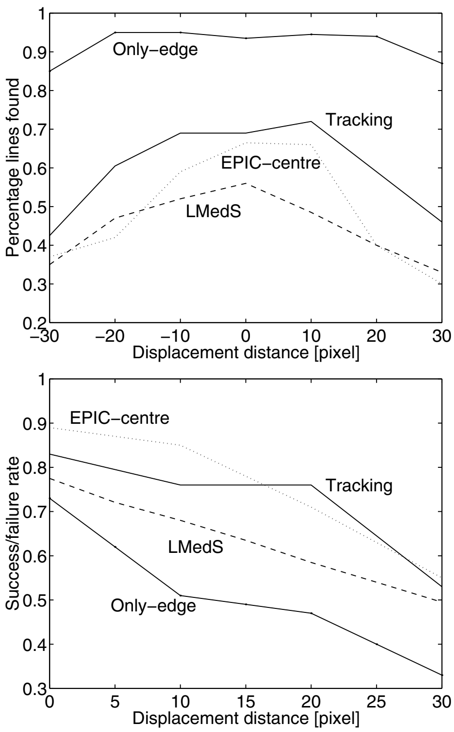


Fig. 9. The detection rate (*top*) and the success/failure (S/F) rate of the only-edge, the LMedS, the EPIC-centre, and the tracking algorithms over the distance, relative to correct localisation. The S/F rate (*bottom*) is given by the relation of correctly versus falsely found features, over all features found

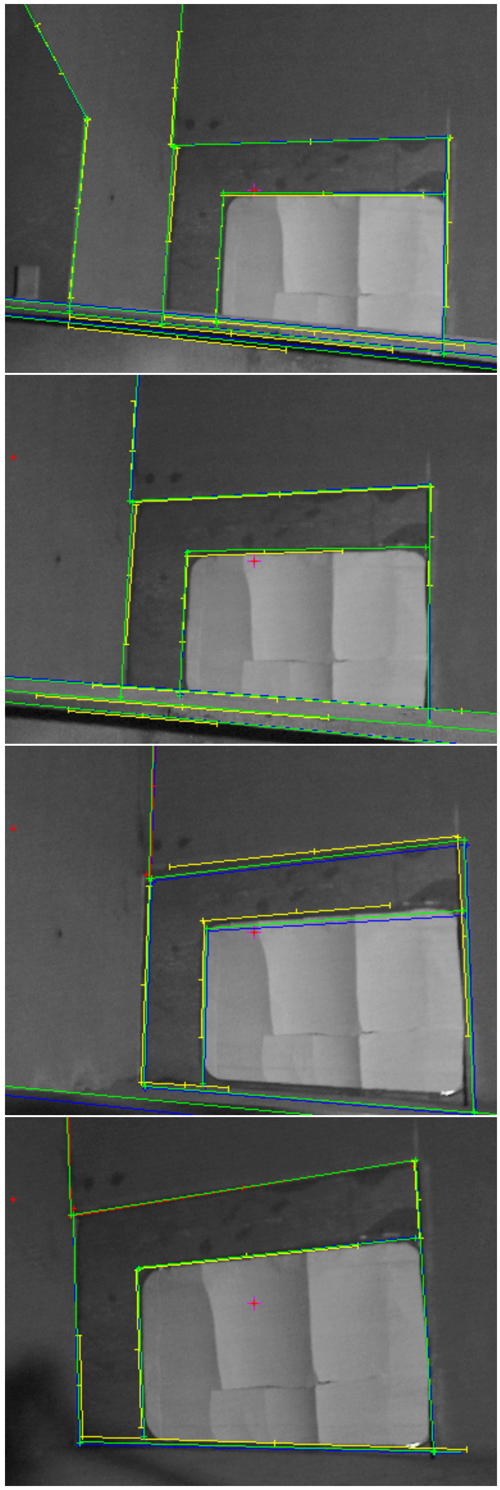


Fig. 10. Four images from a sequence of 309 images of tracking from the moving robot. The *light lines* indicate lines found, while the *dark lines* indicate the re-projection of the estimated pose

An example of tracking is given in Fig. 10. The motion of the robot is depicted in Fig. 11. The jerky motion of the robot can be clearly seen. While tracking is fast and shows good robustness, the maximum-allowed object motion is still restricted. The use of inertial sensors proved effective to compensate fast angular motions. This aspect will be investigated more closely by integrating a full 6-D inertial sensor suit with the optical tracking methods.

Several measurement sets, of eight tests each, were executed with the integrated system, as given in Fig. 2. The head orientation was roughly parallel to the T-truss, which made it easier to take a reference measurement. Table 1 summarises the measurements and gives the standard deviations and the mean error to the reference measurement. The maximum deviations are $+58/-14$ mm and $+1/-0°$. The 3-D standard deviation is 4.64 mm. However, it is observed that the z measurement deviates consistently. The measurements were taken in the centre of the mock-up, and it was observed that the bottom plate of the mock-up bent downwards. The reference measure is biased by this bending of the plate. Considering the results of this set of measurements, a measurement tool to determine the deviations of the mock-up would be helpful. Such a tool is planned as extension of RobVision.

The standard deviation of 4.64 mm reported in these tests is in the range of the pilot goal for the RobVision system ($2\sigma < 10$ mm). The accuracy of the measurement is also seen in the overlay of the features over the image. For the tracking sequence, Fig. 10 shows this overlay and indicates the quality of pose estimation. The same measurements have been made without the 3-D junction information. The standard deviations have been found to be not as good (see Table 1). The 3-D point measurements alone also do not give good confidence values and, for the two or three points measured, the full pose cannot be always calculated.

Table 1. The standard deviation (std) of the measurements and the mean distance to the reference measurement for the integrated and the monocular case. A reference for the roll axis was difficult to obtain, so this measurement is not considered. The last column presents the 3-D position standard deviation, with respect to the mean position

Measure	x [mm]	y [mm]	z [mm]	roll [deg]	pitch [deg]	yaw [deg]	position 3D [mm]
Uncertainty using monocular and stereo features							
Std	5.49	3.29	8.48	0	0.92	0.35	4.64
Mean	4.88	8.62	51.25	0	0.62	0.13	52.49
Uncertainty using monocular measurements							
Std	22.87	22.83	19.53	0	1.16	0.46	37.33

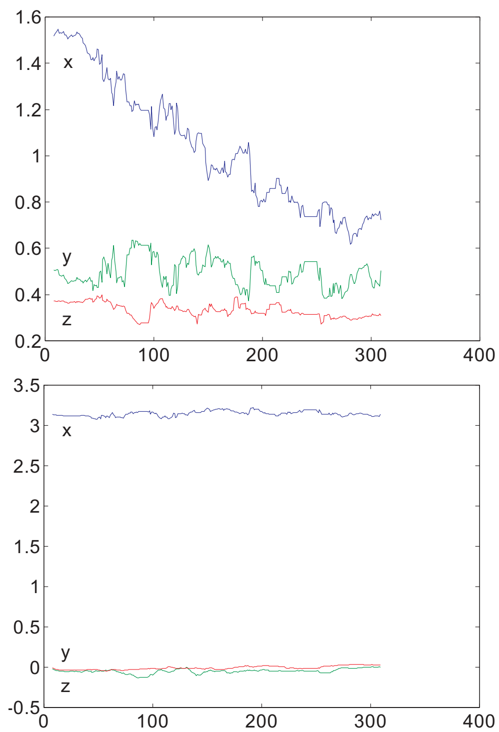


Fig. 11. The result of pose estimation during the tracking sequence of Fig. 10 of the walking robot

The measurement sets indicate that the system reports, in the worst case, a standard deviation of 35 mm. However, the reference measurements are difficult to obtain and seem biased. Repeated performance indicates a standard deviation of 4.64 mm. The features were at a distance of 1–3 mm from the camera(s). On average, about 10 line features and 2–3 3-D junctions were used. Using all features results in the more accurate pose estimates. Hence, the redundancy of using two visual methods to find and measure features has shown its advantage.

This result compares favourably with the need of the ship building application to place the robot within ±10 cm. However, the target goal of ±1 cm could not be quiet reached. On the other hand, the systematic errors (e.g. of the floor plate) indicate the need to measure the actual dimensions of the mock-up. To obtain a measurement of the planar surfaces projected light or depth, imaging is proposed.

Tracking in Fig. 10 did not use *head stabilisation*. Adding the stabilisation of three inclinometres mounted on the Euro-Head slightly improves the pose signal in the Fig. 11. The two horizontal inclinometres are directly used to actively compensate the robot-body motion. The loop is closed at a higher rate than the visual loop (20 ms). The third, vertical, body rotation cannot be compensated by the head. However, the signal of the inclinometre is transmitted to V4R, which uses the signal to adjust the reference-body frame accordingly. The main effect is better reliability of tracking, since the jerky motion is reduced and the likelihood of loosing track is reduced. This is best reported by the percentage of features that is not found for comparable sequences. For the sequence in Fig. 10, the percentage of features not found decreased from 21% to 11%, mainly due to the fact of the lower image motion. One edge (in the back left) is not found during most of the sequence, due to bad contrast. The performance of tracking, using EPIC and validation, is given in more detail in [28].

In summary, tracking operates at a cycle rate of 120 ms when using a full line and junction 3-D wire-frame model of the mock-up with about 20 visible lines. Head stabilisation considerably improves the reliability of tracking. The accuracy obtained is ± 1 cm/m distance to the ship structure.

9 Lessons learned

The final demonstration of the project showed the feasibility of navigating a walking robot through part of the ship's structure. While the goal of building a concept demonstrator has been fulfilled, several system aspects need reconsideration to obtain a system that complies with industrial and commercial needs.

The *accuracy* of the head is sufficient; however, at larger distances, 3-D point-measure accuracy degrades. Points further than 3 m do not improve pose accuracy. This is fine for the small sections of the ship, but a wider baseline is necessary to obtain better accuracy when the distance to the object is greater.

A practical aspect is to improve the *range of the head axes* to turn the head farther to the left or right and, hence, to be able to follow features a longer time. In general, a range of at least ±90° is advisable for a system that can actively select view points for navigation.

The foremost improvement was and still is the *reliability* of sensing. In particular, the jerky robot motions ask for a countermeasure. The posture control of the robot eventually makes up for drastic jerks (e.g. when the robot lifts a leg) and visual tracking is regained. A more robust approach was indicated by the use of inclinometres to actively stabilise the cameras. The next prototype will be equipped with a full 6-D inertial system. The fast response of inertial sensors and the ability to measure high accelerations of the inertial system complements the more accurate image-based tracking. Visual sensing is used to correct the typical drift of the inertial sensors. The stabilisation using inclinometres for three axes has already improved performance. It is expected that the full 6-D compensation will further improve the tracking results and that it is also advantageous over many fixed but not actively compensated cameras. This result agrees with the findings from air-vehicle [12] and helicopter steering [2]. Integrating 6-D inertial and vision sensing has been shown in simulation studies in [24]; however, a lot of experimental work is needed to achieve reliable system performance and accurate interplay of the sensors. The results indicate that a head combining 6-D inertial sensing with visual tracking should be able to provide fully autonomous navigation in indoor settings.

While the work on RobVision improved *robustness* of tracking considerably, more needs to be done. For better pose estimation, the distance of features should be taken into account. The distance of features could be also used to adapt the size (by changing the resolution) of the search window. The biggest problem that has been encountered is that of not finding edge features, due to poor contrast. However, from manually investigating these cases, it seems that pyramid or scale-space approaches would adapt detection, depending on relative significance. This line of work will be pursued in future projects.

Finally, a lesson that has been confirmed is the known fact that *integration* is full of surprises and that it is time consuming. The communication tool described in 4.3 turned out to cut down the on-site integration time considerably. It enables testing of the operability of command exchange and reduces the on-site time used on problems of physical connection and the dynamic system aspects.

10 Summary of results

The objective of the RobVision project is to navigate a walking robot into ship sections using visual feedback. The components of the system consist of the components walking robot, CAD-system, and two redundant vision systems that provide the feedback to steer the robot. The system demonstrated the following.

- A modular walker is able to operate with 6 or 8 legs without any hardware or software changes and can carry a work package of 50 kg into the cell.
- C2V can use the CAD model to evaluate and deliver significant features that the cameras on the robot should see along the path of the robot.
- The redundancy of two vision systems is useful for improving the reliability of finding and tracking features.
- Features, such as lines, 2-D and 3-D points and junctions, ellipses, or surface normals, can be integrated with one algorithm to determine the pose of the object.
- Image processing is executed in 120 ms (and can be improved to obtain frame rate 40 ms), which allows fast real-time operation.
- The approach is ready to be applied to locate and measure any object that has been described with a standard CAD system.

This technique opens up other potential applications. The model-based approach enables us to measure any modelled object and to feed back the measurement data directly into the CAD system. The impact of this capability is manifest. The industrial partners in the ship building and the construction industry can supervise the production quality and consequently reduce production time.

Acknowledgements. This work has been mainly supported by the RobVision project Esprit 28867 and is partly supported by the Austrian Science Foundation (FWF) under grant P13167-MAT and Project GRD1-1999-10693 FlexPaint.

References

1. Alferez R, Wang YF (1999) Geometric and illumination invariants for object recognition. IEEE Trans PAMI 21(6):505–536
2. Amidi O, Kanade T, Miller R (2000) Vision-based autonomous helicopter research at Carnegie Melolon robotics institute (1991–1998). In: [27]
3. Bräutigam CG (1998) A model-free voting approach to cue integration. Dissertation, KTH, Stockholm
4. Beis JS, Lowe, DG (1999) Indexing without invariants in 3D object recognition. Pattern Anal Mach Intell 21(10):1000–1015
5. Beveridge JR, Riseman EM (1997) How easy is matching 2D line models using local search? Pattern Anal Mach Intell 19(19):564–579
6. Christensen HI, Kragic D (1999) Robust vision for manipulation and navigation. In: Vincze M (ed) Proceedings of the Conference on Robust Vision for Industrial Applications, 23rd Workshop, AGM/AAPR, pp 29–48
7. Hager GD, Christensen HI, Bunke H, Klein R (eds) (2001) Sensor based intelligent robotics international workshop, Dagstuhl, Germany, October 15–20, selected revised papers. Lecture notes in computer science, vol 2238. Springer, Berlin Heidelberg New York, Seminar 0421
8. Deliverables of the RobVision project (2000) Available from robvision.infa.tuwien.ac.at
9. Dickinson SJ, Wilkes D, Tsotsos, JK (1999) A computational model of view degeneracy. IEEE Trans PAMI 21(8):673–689
10. Eberst C, Barth M, et al (2000) Robust vision-based object recognition integrating highly redundant cues for indexing and verification. IEEE ICRA Int. Conf. on Robotics and Automation, pp 3757–3764
11. Rencken W, Feiten W, Soika M (2001) Large consistent geometric landmark maps. In: *[7]*, pp 164–183
12. Fuerst S, Dickmanns ED (1999) A vision based navigation system for autonomous aircraft. Rob and Autonom Sys 28:73–184
13. Gasteratos A, Sandini G (2000) On the accuracy of the Eurohead. LIRA-TR 2/00
14. Gonzalez de Santos P, Armada MA, Jimenez MA (2000) A four-legged mobile platform increases productivity and improves quality and working conditions for industrial naval applications. IEEE Rob Autom Mag 7(4):35–43
15. Grimson, WEL (1990) Object recognition by computer. MIT Press, Cambridge
16. Hager G, Toyama K (1998): The XVision-system: A portable substrate for real-time vision applications. Comput Vision Image Underst 69(1):23–37
17. Harris C (1992) Tracking with rigid models. In: Blake A, Yuille A (eds) Active vision, pp 59–73, MIT Press, Cambridge
18. Kim DS, Nevatia R (1998) Recognition and localization of generic objects for indoor navigation using functionality. Image Vision Comput, SI 16(11):729–743

19. Kosaka A, Nakazawa G (1995) Vision-based motion tracking of rigid objects using prediction of uncertainties. ICRA Int. Conf. on Robotics and Automation, pp 2637–2644
20. Kragic D, Christensen HI: Cue Integration for Manipulation. In: [27], pp 1–16
21. Krautgartner P, Vincze M (1999) Optimal image processing architecture for active vision systems. In: Proceedings of the International Conference on Vision Systems, Gran Canaria, Spain, 13–15 January, pp 331–347. LNCS 1542. Berlin Heidelberg New York, Springer
22. Nagel HH, Müller T, Gengenbach V, Gehrke A (2000) Spatially-adaptive filtering in a model-based machine vision approach to robust workpiece tracking. In: [27]
23. Pirjanian P, Christensen HI, Fayman JA (1998) Application of voting to fusion of purposive modules: an experimental investigation, Rob Autonom Sys 23:253–266
24. Rehbinder H, Ghosh BK (2001) Multi-rate fusion of visual and inertia data. MFI, Baden-Baden
25. Thompson RL, Reid ID, Munoz LA, Murray DW (2000) Providing synthetic views for teleoperation using visual pose tracking in multiple cameras. In: Proceedings of the IEEE SMC, Part A 31(1):43–54
26. Vincze M, Ayromlou M, Kubinger W (1999) An integrating framework for robust real-time 3D object tracking. In: Proceedings of the International Conference on Vision Systems, Gran Canaria, Spain, 13–15 January, pp 135–150. LNCS 1542. Berlin Heidelberg New York, Springer
27. Vincze M, Hager GD (eds) (2000) Robust vision for vision-based control of motion. IEEE Press, Piscataway, N.J.
28. Vincze M, Ayromlou M, Ponweiser W, Zillich M (2001) Edge projected integration of image and model cues for robust model-based object tracking. Int J Rob Res 20 (7): 533–552
29. Wunsch P, Hirzinger G (1997) Real-time visual tracking of 3-D objects with dynamic handling of occlusion. ICRA Int. Conf. on Robotics and Automation, pp 2868–2873
30. Zisserman A, Forsyth D, et al (1995) 3D object recognition using invariance. AI 78:239–288

Markus Vincze graduated in 1988 in mechanical engineering from Vienna University of Technology (VUT) and in 1990 as M.Sc. from Rensselaer Polytechnic Institute. He finished his Ph.D. at VUT in 1993. With a three-year grant from the Austrian Academy of Sciences he worked as guest researcher at HelpMate Robotics Inc. and at the Vision Laboratory of Gregory Hager at Yale University. Presently he is with the Automation and Control institute at VUT, where he leads national and European projects in the area of robust sensing for service robots and industrial applications. With Gregory Hager he edited an issue on Robust Vision for IEEE and is (co-)author of over 100 papers. His special interests are vision techniques for real-world robotics solutions.

Minu Ayromlou received the M.Sc. degree in computer science from the Vienna Universtity of Technology, Vienna, Austria, in 1996. In the same year she joined the Institute of Flexible Automation of the Electrical Engineering Faculty as research assistant. In 1998 she was a visiting scientist at the Computer Science Department at Yale University, CT. Her main research interests include computer vision in robotic applications, in particular robust feature tracking in real-time.

Carlos Beltran was born in Valencia, Spain on 7th February 1972. He received the Diploma and the BSc degrees both in Computer Engineering from the University of Valencia in 1998 and from the Polytechnic University of Valencia in 1996, respectively. Currently he lives in Italy where he works towards his PhD in Electronics and Computer Science at the Laboratory for Integrated Advanced Robotics, Department of Communication, Computer and Systems Science, University of Genova. His research interests include computer vision, robotics, machine learning and realtime robotic architectures.

Antonios Gasteratos was born in Athens, Greece on 10 March 1972. He received the Diploma and the Ph.D. degrees both from the Department of Electrical and Computer Engineering (DECE), Democritus University of Thrace (DUTh), Greece, in 1994 and 1999, respectively. During 1999-2000 he was a postdoc fellow in the Laboratory for Integrated Advanced Robotics, Department of Communication, Computer and System Science, University of Genoa, Italy, with a TMR grant. Currently he is a visiting Lecturer at the DECE, DUTh teaching the course of Automated Electronic Measurements. His research interests include robot vision, fuzzy and non-linear image processing, sensor and data fusion and digital VLSI design. He is a member of the IEEE, the Technical Chamber of Greece (TEE) and the Hellenic Society for Artificial Intelligence (EETN).

Simon Hoffgaard obtained his M.Sc. in Production technology with focus on robotics from Aalbrog University in June 2000. He was research Assistant at Aalborg University from August 1999 to December 2000 for the EU Esprite Project RobVision with responsibility for generating models from CAD to help a spider robot navigate in large ship structures. Since 2001 he is system engineer at Photronic A/S with responsibility for the System Construction Department, with focus on programming and controlling robots for handling and quality assurance.

Ole Madsen holds a M.Sc. (1988) and a Ph.D. (1993) from the Department of Production, Aalborg University, Denmark. Since 1989 he has worked at the Department of Production, Aalborg University, where he now is working as associate professor in the area of automatic planning and control manufacturing systems. His main interests are robotics, sensor based control, CAD/CAM and automation.

Wolfgang Ponweiser received the M.Sc. degree in electrical engineering from the Vienna Universtity of Technology, Vienna, Austria, in 2000. He started his research of component based programming for sensors in the field of robotic as a research assistant at the Institute of Flexible Automation of the Electrical Engineering Faculty. In 2001 he made a three month sojourn at the Center for Autonomous Systems at the KTH Stockholm.

Michael Zillich received the degree of M.Sc. in Mechatronics from Linz University in 1998. He then joined the Automation and Control Institute as a PhD researcher, working on 3D object tracking and pose determination. He spent six month as a guest researcher at the Computer Vision and Active Perception Laboratory in Stockholm, Sweden, working on appearance based object recognition using support vector machines. After finishing one year of alternative military service he is again at the Automation and Control Institute continuing his thesis on indexing based 3D object recognition.