ROBVISION
Vision Based Navigation for Mobile Robots

W. Ponweiser[1], M. Ayromlou[1], M. Vincze[1], C. Beltran[2], O. Madsen[3], A. Gasteratos[2]

Abstract—**This paper introduces the system, developed during the Esprit project RobVision (ROBust VIsion for Sensing in Industrial Operations and Needs), that navigates a climbing robot through a ship section for inspection and welding tasks. The basic idea is to continuously generate a robot position and orientation (pose) signal by matching visual sensing information from the environment with predetermined CAD-information. The key for robust behaviour is the integration of two diffierent vision methods, one measuring 3D junctions with a stereo head, the second tracking edge and junction features in a single image. To render tracking robust and fast, model knowledge such as feature topology, object side, and view dependent information is utilised. The pose calculation step then integrates the finding of both vision systems, detects outliers and sends the result to the robot. The real-time capability is important to reach an acceptable performance of the overall system. Presently a pose update cycle time of 120ms is achieved. Due to appearing jerks of the robot accelerometers are used for stabilization. Experiments show that our approach is feasible and reaches the positioning accuracies demanded.**

Keywords— **CAD based vision, visual robot navigation, system architecture.**

I. Introduction

QUALITY assurance and intelligent products are key roads to success in global competition. Supervising and automatically measuring the quality of parts in the production of large structures can reduce work costs by up to 20% (expected by Odense Steel Shipyard Ltd. (OSS), the end user of this project). The inspection of these structures needs automated systems to position the tools necessary in a large environment. For both, quality measurement and positioning of inspection tools, the key is to generate the 3D pose of objects. The task of the EU-funded project RobVision was to navigate a climbing robot into the ship structure and to position it for inspection and welding tasks.

Industry commonly uses CAD systems to design parts or working areas. In RobVision this model knowledge is applied to initialize the vision process. The CAD system provides features to a vision module that tries to find these features in the images. Cameras of a stereo head mounted on the robot deliver the images. Two alternative image-processing techniques are used, a monocular and a binocular approach. The rationale is to take advantage of the redundant features detected. The redundancy is the key to enhance the robustness of the vision process and to make the overall system more reliable. To ensure correct feature detection, image and model cues (e.g. intensity and the topology of features) are integrated [8] [9]. After feature extraction, the pose calculation algorithm integrates the 3D and 2D feature information found by the two vision processes, calculates the current pose and sends it to the robot.

Fig. 1. The 8-legged pneumatic walking and climbing robot.

The tasks of the RobVision partners is as follows: the mobile robot is an 8-legged climbing robot (Figure 1) constructed by the partner Portsmouth Technology Consultants Ltd. (Portech). The second industrial partner, the customer OSS, delivers the CAD data of the ship and defines the trajectory of the robot. The Department of Production of the Aalborg University develops a CAD system that adapts the model to the essential information necessary for the vision process. The Lab for Integrated Advanced Robotics of the University of Genoa (LIRA-Lab) and the Institute of Flexible Automation of the University of Technology in Vienna process images captured using the model data and integrates this information to generate the 3D pose of the robot.

The following section gives an overview of the modules and functions developed during the project. Section III describes the integration of the two different methods to extract image feature and the subsequent pose calculation. In section IV the subsystem dependencies are described by the means of a state machine. Section V gives then an outline about the demonstration of the system and a summary of the results is given in section VI.

II. Sytem Overview

The entire system is shown in Figure 2. It consists of several modules colored in different greyscales. Each subsystem is provided by one of the project partners and fulfills all the functions drawn inside the according module. The

---

[1] Institute of Flexible Automation, Vienna University of Technology {ma,wp,vm}@infa.tuwien.ac.at, www.infa.tuwien.ac.at

[2] Laboratory for Integrated Advanced Robotics, University of Genoa {cbeltran,antonis}@lira.dist.unige.it, www.lira.dist.unige.it

[3] Department of Production, Aalborg University i9om@iprod.auc.dk, www.iprod.auc.dk

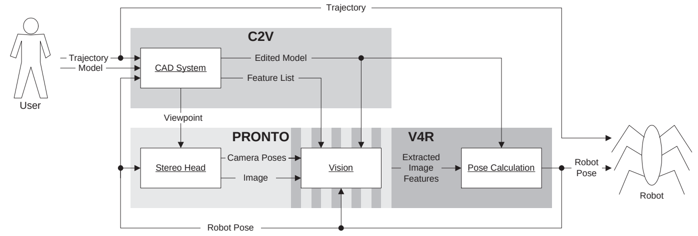


Fig. 2. The subsystems of RobVision.

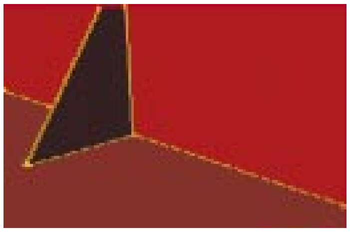


Fig. 3. A view automatically generated by C2V with junction and line features.

next subsections give an overview of the modules and their respective functions.

A. C2V

C2V (Cad to Vision), developed by the Department of Production, Aalborg University, provides the CAD system. The main task is to generate geometrical features for the vision system. These features specify the shape and location of geometrical entities such as lines, junctions and regions that the cameras can expect to see while the robot is moving inside the structure. As it appears from Figure 2 the CAD system has three inputs:

- A *CAD model* that is the geometric model of the structure inside which the robot has to move.
- A *Reference Robot Trajectory* containing a specification of the task or trajectory that the robot has to perform. The user generates this trajectory off-line by using a trajectory planner developed by OSS.
- *Robot Poses* computed by the pose calculation component using the features found by the vision system. For details please refer to section III.

The main output from the CAD system are model and view data containing:

- A *Camera Viewpoint*, which specifies a 3D point (x,y,z) in world co-ordinates towards which the stereo head ought to point.
- A *Feature List* containing a set of robust features that the vision systems can expect to find when looking at the specified viewpoint. Figure 3 shows a typical camera view. The CAD system is sending information about lines and the endpoints of these lines.
- *Model data* containing a specification of relevant information from the underlying geometrical model, such as the feature topology.

The CAD-model and the reference robot trajectory are computed prior to the operation of the RobVision system. An example of a CAD-model and a Reference Robot Trajectory is shown in Figure 4.

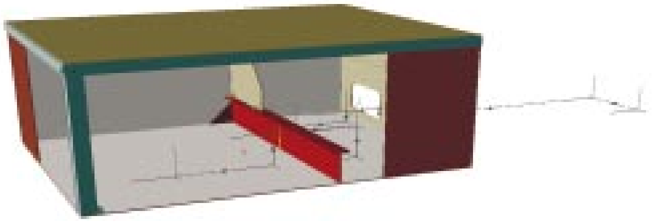


Fig. 4. The input CAD-model and Reference Robot Trajectory.

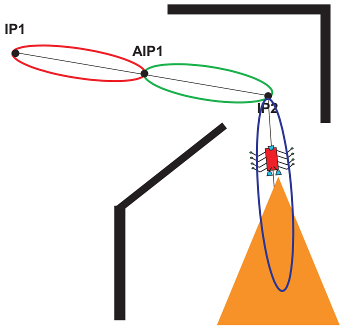


Fig. 5. The ellipses denote areas along the trajectory where a constant set of features is visible.

In this off-line phase of C2V (C2VoffLine) a model of the robot is placed in the starting location of the reference robot trajectory. A viewpoint is determined and the features associated with this view are derived and generated. The robot modeled is then moved forward along the reference robot trajectory while looking at the viewpoint until new robust features become visible and the old robust features become invisible. A new viewpoint is then selected together with a new set of visible features. The robot modeled then continues along the trajectory looking towards the new viewpoint. This approach divides the trajectory into areas where the same viewpoint and features can be used (Figure 5). Figure 6 is an example where the features are the same for two poses on the reference robot trajectory with the same viewpoint.

In the online phase of C2V (C2VonLine) the estimated robot pose generated by V4R is used to identify in which of the areas generated by the off-line system the robot is presently located (Figure 7). The according viewpoint and the features associated with this area are sent to the vision systems.

This approach requires that the actual deviations of the robot trajectory are less than the areas generated by C2VoffLine. Simulations show that deviation of 500 mm or more, depending on the situation, are acceptable.

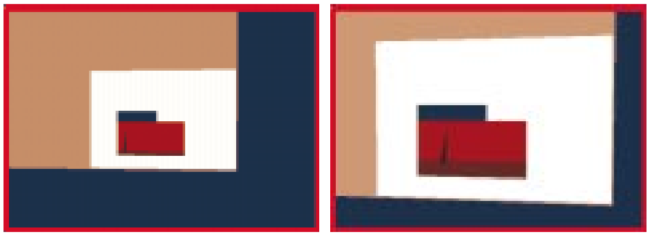


Fig. 6. Two views with the same viewpoint, 1,5 m apart on the robot trajectory

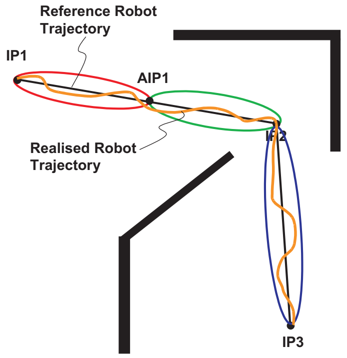


Fig. 7. The Realised- and the Reference Robot Trajectory

B. PRONTO

PRONTO developed by Laboratory for Integrated Advanced Robotics, University of Genoa, is the software/hardware module that is responsible for the stereo head task which in brief consists of:

- Head control
- Head stabilization
- Head calibration

and for the acquisition of 3D feature data suggested by the CAD module (C2V). To perform this task no movements of the robot are allowed. This consents PRONTO to perform high accuracy 3D measurements of the features suggested by the CAD module (C2V). For each of the features PRONTO applies a stereovision algorithm . The main idea of this algorithm is that of measuring the depths of edge junctions. A Hough technique is implemented to extract the lines on the image planes of the stereo pair. A weighted LMS (Least Mean Square) method is used to relate them to the features provided by the CAD system [4]. Then a closed loop method is followed, so that by moving simultaneously the three degrees of freedom of the head the junction is put 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 3D position of the junction relatively to the head. The time used for this measurement process depends on the number and the quality of the features to be detected. During the movement of the robot, the vision algorithm based on tracking (V4R) is used and PRONTO concentrates its computational efforts to stabilize the head and to keep the gaze of the cameras fixated on the view point recommended by C2V. The stabilization is performed using angular accelerometers that react to the movements of the robot. This consents PRONTO to calculate the angular velocity by integrating the acceleration in a 40 ms cycle. This velocity is then multiplied by a predefined factor and directly introduced in the head motors producing a compensation movement.

The gaze of the head is maintained in the direction of the viewpoint in a 240 ms cycle using the robot pose feedback generated by V4R. During this operation the inertial stabilization is deactivated in order not to confuse the inertial sensors. Both, stabilization and gaze orientation have the purpose to guarantee a good image quality to increase the robustness of the tracking algorithm described in the next section.

C. V4R

V4R (Vision for Robotics), developed by Institute of Flexible Automation, Vienna University of Technology consists of two functions. The first task of V4R is the monocular 2D feature search and tracking. That means V4R contains one of the two units of the overall vision system. The second unit of V4R is the pose calculation component to calculate the pose of the robot relative to a reference system.

The emphasis of the vision method is basically to provide robust features in real-time, i.e., keeping the processing time lower than the frame rate of the camera (40ms). To handle the real-time constraint a windowing method is applied to limit the processing time [5]. Robustness can be achieved by including image and model information - so-called cues - into the vision process. The method used is a combination of cue integration strategies (with cues like intensity values, color, texture) and the RANSAC-method (Random Sample Consensus) for Edge finding [1] [9]. V4R is presently able to track edge features like lines, junctions, ellipses and arcs. In near future regions will be included.

For the first search of features the projection of the model into the image indicates an approximate position of the features. Since the model contains the model of the features and their topological relations to each other, checking of the actual geometric relations like, e.g., connectedness of some features can assure that the right feature is found. Furthermore some other model cues like color information of the object can be used to eliminate wrong feature candidates. Once a feature is found for the first time, this feature is then tracked in the next cycle. Additional information of the features found in the image (image cues like intensity) can be stored to facilitate the search for the same feature in the next tracking cycle. An example for a tracking sequence is given in Figure 8.

The second unit of V4R is pose calculation which is descibed in the next section.

III. Feature Data Integration (Pose Calculation)

The movement of a mobile robot, and especially of a walking robot, is very unprecise. This is the result of the sliding wheels or feet on the ground. Therefore the navigation of a mobile robot needs the feedback of its own 3D position relative to the environment. In contrast to other location sensors (e.g. laser ranger, sonar) vision is able to generate a full 3D position, which is needed when stepping over trusses or when climbing the walls. The difficulty when using vision sensors is a robust sensor behaviour. To obtain a more robust sensor system two different vision methodes were implemented and integrated (see section II-B and II-C).

The result of the binocular vision algorithm of the Pronto subsystem are 3D positions of edge junctions in stereo head co-ordinates. To obtain accurate junction positions two line edges are intersected. In contrast, the monocular vi-

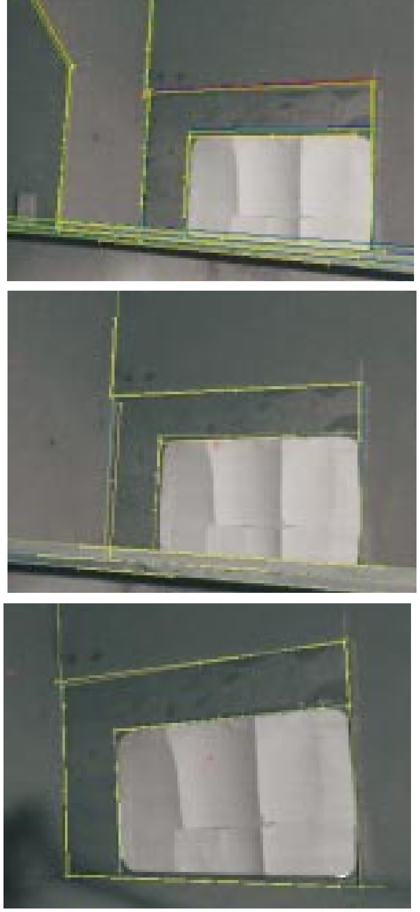


Fig. 8. A tracking sequence along the robot trajectory: Projection of the model and result of feature finding

sion method used in V4R is able to deliver 2D positions of lines and junctions in the two dimensional image co-ordinate system. If available the surface normal as found from tracking a cricle (seen as ellipse in the iamge) can be also used for pose estimation. To be able to integrate these different types of position information an accurate calibration of the camera and the stereo head was developed [4].

During the first step of the pose calculation both 2D and 3D feature positions are converted into the camera co-ordinate frame. In the second step the feature data is fitted iteratively to the model data received from the CAD system. The algorithm employed is based on a method proposed by Wunsch [10]. This algorithm was chosen because it scales linearly with the number of feature, it is fast and it is therefore well-suited for object tracking. The method has then been modified for providing accuracy estimation of the calculated pose as well as outlier detection. Evaluation of measurements and detection of outliers is important to eliminate errors from false tracking and to enable the re-initialisation of these features via a new projection into the image at the next image processing cycle.

Ourlier detection is based on using the weight-correlation matrix to find the normalised corrections for each individual measurement [6]. We proceed by eliminating the measurement with the largest error above 2.5 times the normalised standard deviation. With this procedure outliers can be iteratively eliminated during pose estimation.

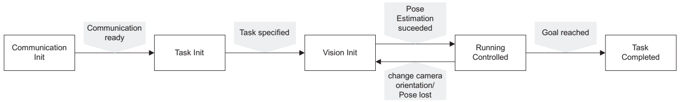


Fig. 9. System States of the RobVision system.

IV. System States

The two different vision modules as well as the CAD system have their special requirements. Especially the initialisation of the image processing techniques requires a fixed robot, and maybe several trials at different viewpoints, to generate accurate and reliable position data. To work out this dependencies of the different modules the whole system is coordinated by system states. Each state defines a special running mode for each subsystem where it gets state dependent input data and should deliver specified output data.

In Figure 9 states are written in boxes and the arrows define the transitions from one state to another. The triggering event according to these transitions is written in gray boxes. The control over the state changes is part of the responsibility of the supervisor.

We introduce the 5 main states that are passed through under regular conditions (neglecting system failure handling) :

- **Communication Init**: During this state a defined communication protocol starts up all communication links required.
- **Task Init**: In Task Init C2V delivers the task specifications to the other subsystems. Especially the entire model of the steel structure is sent to PRONTO and V4R.
- **Vision Init**: The goal of this state is to set up the vision and to find an initial robot pose. To ensure a correct initialization, no movements of the robot are allowed. So the first step is to fix the robot where the stereo head is mounted. Now C2V has to select a camera viewpoint for the actual robot position. This camera viewpoint determines the camera image and the features, which are visible in this image. After receiving all information necessary, PRONTO starts to search and measures all available junctions and V4R starts feature tracking. If the vision isn't able to find enough image features for the pose calculation, it informs C2V to select an alternative camera viewpoint with different image features for the same view.
- **Running Controlled**: This is the running state desired. V4R has already found the initial robot pose in the previous state and continuous feature tracking and robot pose determination is happening. The Robot, C2V and PRONTO get this robot pose as feedback and the robot is moving towards its target position. There are two possibilities for switching the system back to the Vision Init state: C2V computes a new camera viewpoint for the current robot pose with new robust image features. Secondly the vision fails to deliver a robot pose and another view is requested.
- **Task Completed**: This is the exit state when the target position of the robot is reached successfully.

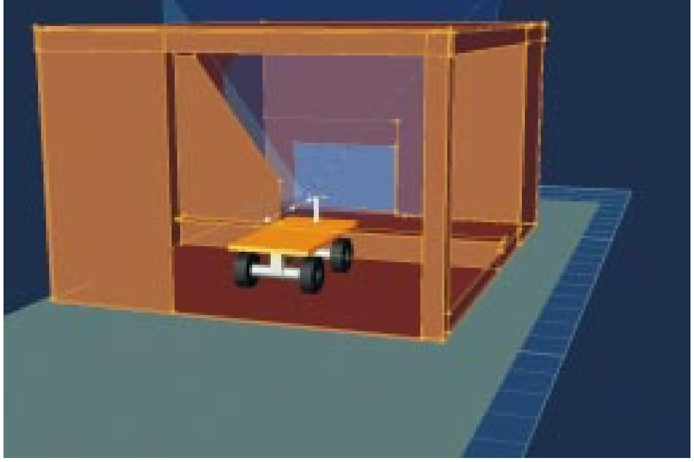


Fig. 10. Model of the mockup where the cameras are looking at a viewpoint

V. DEMONSTRATION

This section gives an overview of the tests for evaluating the whole system. As seen in Figure 10 a mock-up - a typical test environment - was built with structures that can be found in a big ship environment.

Several trajectories were examined to test the robustness of the system and its accuracy. A big challenge is to find a large number of "good" views to enable a successful task completion. Generally spoken a good view is a view containing a large number of robust features. The importance hereby is not only to generate good views along the path but also good alternative views for one robot pose to be on the secure side if pose determination with the first view fails. Malfunction can happen since the mock-up is made of big metal plates welded together, which is a big defiance for the vision. Some welding causes irregular edge features, contrast is sometimes poor and additional features on the plate surfaces can affect pose calculation results negatively. That's why feature redundancy is a basic issue. Figure 11 shows an example for a view. C2V generates the features seen in the image to deliver it to the vision subsystems (Figure 11a). PRONTO is able to extract the 3D coordinates of the junctions (Figure 11b) and V4R the 2D position of edges and junctions in the image (Figure 11c). All this features extracted are integrated in one co-ordinate system to calculate a robot pose. Figure 12 gives an overview of the pose calculation recorded along a path.

As can be seen the pneumatic walking robot produces many jerks. Especially fast changes in the orientation of the robot cause big deviations of the feature positions in the image. Because of the fast but spatially restricted windowing technique such features are then lost. This fact reduces the reliability of the system. The head stabilization implemented lower the optical flow and increases the robustness of feature finding. In Figure 13 the motion of a trolley is compared to the motion of the robot. The smoother movement causes much less deviations of the poses along the trajectory since features can be tracked easier. The resulting robot poses are therefore continuous.

The tests executed show the system benefits from the re-

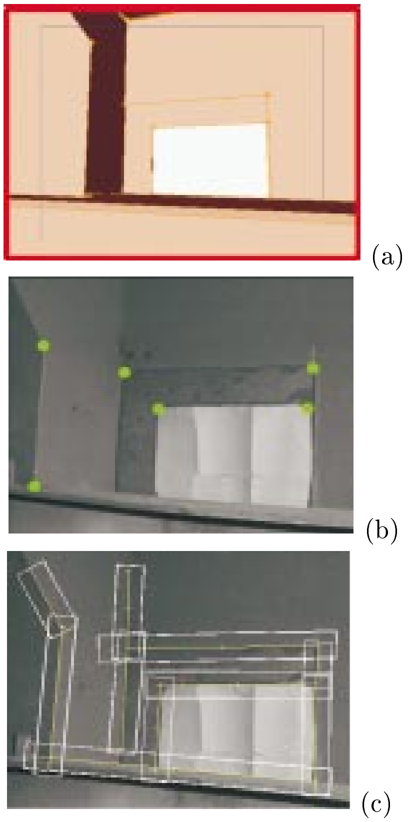


Fig. 11. (a) View with features generated by C2V, (b) 3D junctions detected by PRONTO, (c) lines in their searching windows generated by V4R

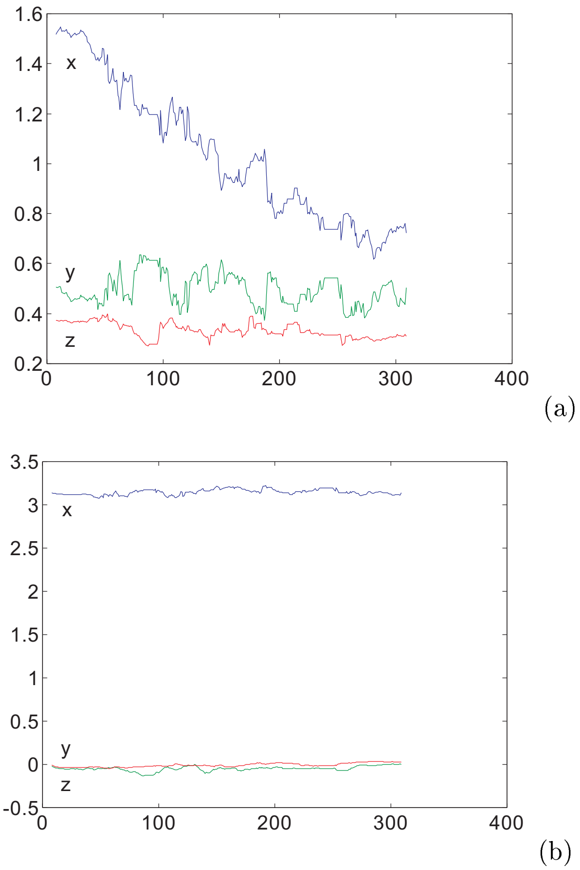
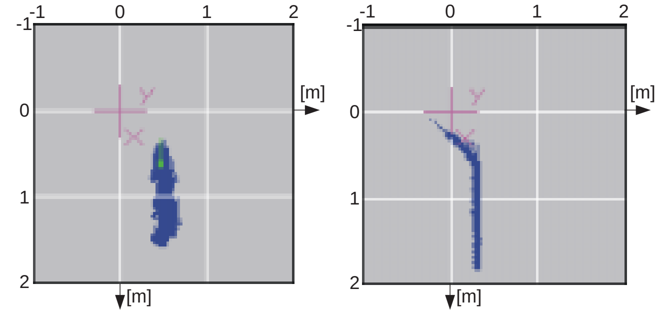


Fig. 12. Robot pose output along the frame cycle of a trajectory, (a) position in [m], (b) orientation in [rad].



(a) (b)

Fig. 13. Projection of (a) robot poses (b) trolley poses along a trajectory into the x/y plane

TABLE I

Average position accuracy results

X [mm]	Y [mm]	Z [mm]	3D position [mm]
5.49	3.29	8.48	10.62

dundancies implemented. As long as enough features can be tracked, the system is able to re-find lost features. Also wrong detected image features are filtered out from the pose calculation. The stabilization of the pose increases by the integration of 2D and 3D features and the continuous update of new image features from the CAD database makes the overall system robust and reliable.

VI. Summary

During the RobVision project a prototype of a visual navigation tool has been built to guide a walking robot through a big steel construction. It can be shown that applying CAD-model information in combination with visual sensing input enables the extraction of continuous robot poses. The model delivered by the user is adapted to allow a selection of features desired. Two separate vision systems utilizing different approaches guarantee enhancement through redundancies. The required sensor data integration is done by the pose calculation. Furthermore synchronization between all the subsystems are managed to achieve an pose update cycle time of 120 ms. The jerks of the pneumatic walking robot is a big challenge for the vision but is partly compensated by including accelerometer data. The average position accuracies reached are summarized in Table 1.

As a conclusion it can be stated that CAD based vision is a potential position-sensor in normal industrial environments. To reach the reliability demanded requires multiple views, redundancies in image processing strategies, sensor data integration and feature validation techniques. Still enhancements are necessary to achieve an overall reliable system behaviour.

Future work includes the integration of linear accelerometers and inclinometers to the existing gyros, in order to perform a more efficient and accurate image stabilization in all 6 DOF. Moreover, since the robustness of the pose estimation method relies on the redundancy of the features measured, more 3D features extracted by stereoscopic vision, such as lines in 3D [2] and the normal of surfaces, are still under investigation. This should enhance the pose estimation in areas poor of features. Extensions of feature validation techniques to reach more robustness of the vision is another important issue. Together with the improvements of the vision capabilities, future work is also necessary to enhance error recovery and fault tolerance behaviors. Applictions for in-door navigation seem then possible as indicated in Figure 14. Due to the high modularity of the system, the integration with extra sensors can be considered feasible and easy to implement.

Fig. 14. Visual navigation using V4R in an office environment.

VII. 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] Fischler M.A., Bolles R.C.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography; Communications of the ACM Vol.24(6), pp.381-395 (1981).

[2] Gasteratos A., Beltran C., Finding the 3D Orientation of a Line Using Hough Transform and a Stereo Pair, Technical Report, TR 3/00, LIRA-Lab, DIST, University of Genoa, Genoa, 2000.

[3] Gasteratos A., Martinotti R., Metta G., Sandini G., Precise 3D Measurements with a High Resolution Stereo Head. IWISPA 2000, Pula, Croatia, 2000, 171-176.

[4] Gasteratos A. , Sandini G., On the Accuracy of the Eurohead, LIRA - TR 2/00, July 2000.

[5] Hager G. and Toyama K. (1998): The XVision-System: A Portable Substrate for Real-Time Vision Applications, Computer Vision and Image Understanding, 69 (1), 23-37.

[6] Kraus K.: Photogrammetry Volume 2, Advanced Methods and Applications, Dümmler/Bonn, pp.205-215, 1997.

[7] Panerai F., Metta G., Sandini G., Visuo-inertial Stabilization in Space-variant Binocular Systems, Robotics and Autonomous Systems, 30 (2000) 195-214.

[8] M. Vincze: Robust Tracking of Ellipses at Frame Rate; Pattern Recognition 34(2), 487-498, 2001.

[9] M. Vincze, M. Ayromlou, W. Kubinger: An Integrating Framework for Robust Real-Time 3D Object Tracking; Int. Conf. on Vision Systems, Gran Canaria, S. 135-150, January 13-15, 1999.

[10] Wunsch P., Modellbasierte 3-D Objektlageschätzung für visuell geregelte Greifvorgänge in der Robotik, PhD Thesis, Munich University of Technology, pp.37-47, 145-149, 1997.