A System Architecture Supporting Multiple Perception Tasks on an Autonomous Mobile Robot

C. Eberst, D. Burschka, A. Hauck, G. Magin, N. O. Stöffler and G. Färber
e-mail: eberst@lpr.e-technik.tu-muenchen.de

Department of Process Control Computer
Prof. Dr.–Ing. G. Färber
Technische Universität München
Arcisstr. 21, 80333 München, Germany

Abstract. Perception tasks for autonomous mobile robots such as navigation, object- and state-identification require reliable and accurate model information as complete as possible. Early disposal of a closed and prevailingly static description of the environment improves the performance of these tasks. In opposite to these requirements, exploration preliminary contributes incomplete, unrelated and uncertain information partially with significant delay. This paper describes a system structure that handles reliable and static information for localization and other high-level tasks as well as the acquisition, stabilization and filtering of uncertain and dynamic features and their conversion to features on a higher level. We present a system architecture that combines a hierarchical geometric model with multi-stage sensor data interpretation, which allows fast access to relevant and accurate information as well as sensor-based model generation and maintenance.

1 Introduction

The design of autonomous mobile robots (AMRs) that can cope with unexpected disturbances like obstacles or misplaced objects is an active field of research. Such a robot assesses the situation by comparing data from one or more sensors with an internal representation of its environment. Differences between expected and observed information are used e.g. to localize the robot in the world or to update the position of an object that is to be manipulated. Such applications require specialized model representations that allow a fast access to the relevant data for different sensors and tasks. To support sensor-based model generation and update a very general description is needed, though, that serves as a base to derive the sensor-specific representations. Most modeling systems described in literature focus on some aspects of the set of tasks described above, either presenting primarily static models geared to specific sensors and tasks [1] or environments [4], or working on methods to reconstruct environmental models without any a-priori knowledge [2]. In contrast to this we present a system architecture that offers both fast access and sensor-based reconstruction by combining a hierarchical geometric model with sensor- and task-specific information on the one hand with a likewise hierarchical sensor data interpretation process on the other, which results in a highly interconnected, mixed bottom-up/top-down structure. This facilitates perception tasks by offering fast access to relevant and reliable information, at the same time allowing sensor-based model update and generation, with the geometric layer guaranteeing consistency of the different sensor- and task-specific ones.

The paper is organized as follows: The proposed system architecture is described in section 2; subsequent chapters treat the building blocks in more detail.

2 System architecture

Figure 1 depicts the hierarchical system architecture. The system is designed to support a multi-sensor system, but for the sake of simplicity only the parts appropriate for a stereo CCD-camera system are shown. Raw video data is preprocessed to extract video-specific features, in our case 2D line segments (section 3). Those features are combined to generate 3D information and then stabilized by the *Dynamic Local Map* (DLM) (section 4). The *Predictive Spatial Completion* (PSC) (section 5) clusters features and introduces spatial reasoning into the sensor-based data acquisition to generate high level descriptions, from polygons up to objects. This bottom-up processing is supported by top-down feedback on each level: Preprocessing can steer the sensor by specifying regions of attention, while the DLM focuses the attention of the processing stage by predicting 3D line segments. The PSC inserts hypothetical 3D features into the DLM, which are in their turn checked against sensor data, which can enable feature extraction in areas of low contrast and ambiguous scenes based upon context. Additionally, model information is integrated at each stage, thereby allowing for example the DLM to use formerly explored information when the robot reenters a room, and submitting to the PSC the the information necessary for object identification. The *Geometric Symbolic Model* (GSM) reflects this hierarchical interpretation process in a similarly hierarchical model structure, which will be described in more detail in section 6. Perception tasks can access and compare sensor and model information at different levels of abstraction, corresponding to their needs. Robot localization e.g. typically works on 2D feature level, covering the whole area visible for the sensor, while object recognition relies on 3D clusters. In addition, the model can be accessed on a symbolic level, which allows applications to query information on objects by class or name.

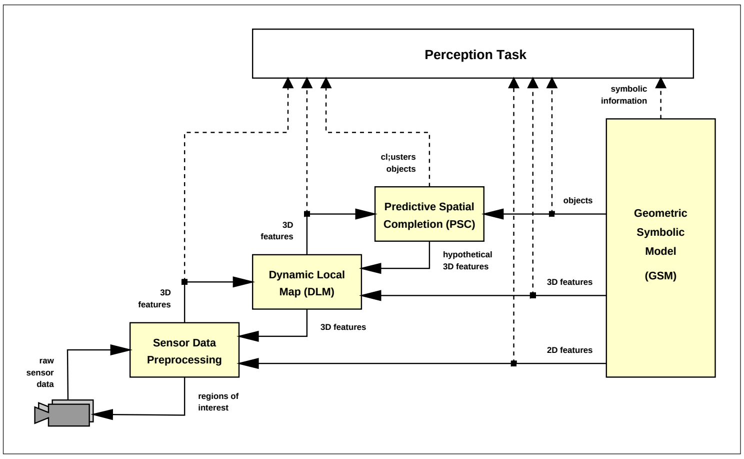


Fig. 1.: System architecture

3 Sensor data processing

3.1 Image processing

The main sensor is a CCD camera system for both binocular and monocular stereo, based on line features. Line feature extraction has to be fast, especially for monocular, motion based stereo. Therefore, we use contour tracing to overcome the time consuming multistage process of smoothing, labeling, thinning and symbolizing the image as it is done in classical edge line extraction schemes. In those schemes all pixels undergo processing with sometimes huge convolution matrices.

Being more selective in applying expensive computations is not a new idea in the image processing domain, however, we go further and combine most steps needed for extracting a contour segment into three small operators, built by 5x5 matrices, whose convolution results are used to determine the next contour spot. This algorithm is self steering, as soon as it has been set on a contour. A tremendous timing gain is reached by applying the convolution matrices only to about 5% of the pixels of an image. The matrices have been originally derived by an electrodynamic analogon, however, it has been found that classical gradient and Laplacian-of-Gaussian (LoG) matrices have delivered similar results. Though 5x5 matrices are much more susceptible to image noise than larger matrices due to missing smoothing effects, high frequency noise reduction is achieved by a kinematic movement model of the contour spot, using mass and accelerations. Searching for starting points efficiently is almost as important as tracing an already found contour.

We use a grid based search scheme, which minimizes the number of pixels to be examined to detect a contour. A contour is assumed at zero crossings of the 2nd order derivative when there is a certain steepness. From that point, the contour is traced in both directions until either another contour or the image margin has been reached, or the contour strength has been fallen below a threshold. To accelerate the starting point search, grid elements are marked as processed, as soon as there has been found a certain percentage of contour spots within that grid element. So no more start points are searched in those grid elements. This is allowed, as line segments have a minimum distance between each other, and the grid size is rather small (16-32 pixels).

A-priori known edges and hypothetically predicted edges are handled by searching starting points in the grid elements touched by the expected line, and by lowering the detection thresholds there. Furthermore, the minimum contour strength, measured by the gradient amount, can be lowered as well to verify even poorly visible edge segments, which have been predicted by the PSC. The grid based search algorithm helps to limit line extraction on regions of interest (ROI) e.g. when identifying a door wing angle (chapter 6): The ROIs are the upper and lower end of the hinge side, where the wing and frame contours meet. As the contour trace algorithm will trace along the cyclic contour, all the edges of the door wing, as well as the door frame are extracted. Other contours traversing the ROIs are extracted as well, however, no starting points are searched outside the ROI. The polygon approximation algorithm is based on calculating the sliding mean value of the secant slope angles between a detected corner point or start point and the current contour points. If the difference between the sliding mean angle and the current angle exceeds a dynamically decreasing threshold angle, a new corner point has been found. The algorithm is geared towards real time operation, as it avoids recursive determination of corner points usually performed in the classical split&merge methods. Symbolic line segment descriptions are gained from the contour chains by least square regression analysis, further limiting the influence of image noise. Edge line intersection finally leads to the precise corner points. By all this we cut processing time by a factor of about 10 compared e.g. to the standard edge extraction algorithm implemented in the image processing software *Vista* [10], thus enabling feature extraction with a frame rate of 10 Hz for 287x736 8bit grey level images on general purpose standard hardware (i860 at 40 MHz).

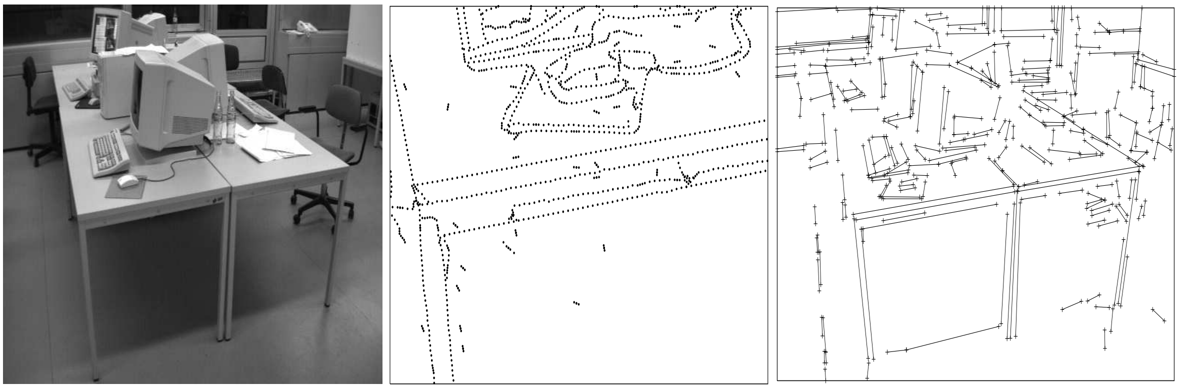


Fig. 2.: *from left to right*: shows a grey level image, the intermediate contour spots as an enlargement of the lower left central part, and the final, length filtered edge segments.

3.2 Three-dimensional reconstruction

The Dynamic Local Map (DLM) stores three-dimensional sensor features. In case of the CCD camera the environment is described by the edge-lines of the objects. The extracted 2D features in the images are processed to 3D lines within the *Sensor Data Processing* unit. We use two different methods to generate these features.

Monocular camera system - The 2D lines from a single camera are tracked in consecutive sensor frames assuming differential image motion. If the stereo base for a tracked line segment is wide enough, the 2D line segment is triangulated to a 3D line segment, which is then stored in the DLM. The stereo base is determined by odometry.

Binocular camera system - The 2D lines from two cameras are matched according to their epipolar restrictions. Often, there are several possible matches for a given line. All these matches are stored in the DLM to be verified from a different position in the environment.

The information stored in the DLM is used to find the already known features in the images. The position of these features is refined. The remaining features are matched and stored with a low confidence value for their real existence. This reduces the computational effort to establish the correspondences and results in a more reliable 3D information. The stored features are verified, utilizing the motion of the robot.

4 The Dynamic Local Map

4.1 Function in the system

The 3D features extracted by the sensor data preprocessing (fig. 1) show a poor *accuracy* of their position and a varying *confidence* in their real existence due to physical and computational limitations of the applied sensor system. The quantization of the perceived sensor features results in a limited accuracy of the reconstructed 3D lines.

Therefore, an important function of the applied DLM is to filter and stabilize the extracted features in a local representation of the environment. The basic idea of our filtering technique is to use not only the data extracted in sensor reading but to combine it with older data and thus locate features more precisely and enhance the probability factor of reappearing features. The possible ambiguities of the correspondence problem are dissolved in consecutive sensor readings from a moving robot. The three-dimensional match of the extracted features with the previous content of the DLM allows a reconstruction of the true objects' dimensions despite of the limited view angle of the applied sensor and it improves their *accuracies*.

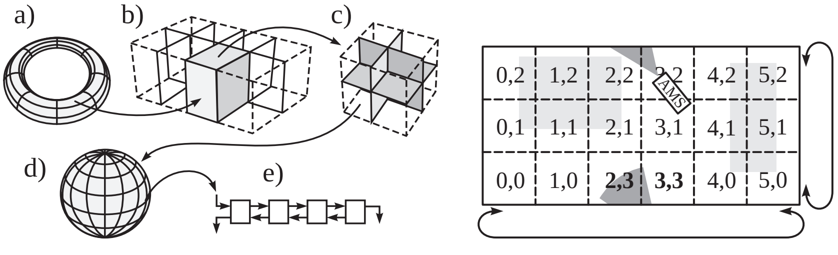
The second function of the DLM in the introduced system is the acceleration and stabilization of the sensor data preprocessing. The data stored in the DLM is obtained from different sources. The most important source is the sensor system, which is capable to register the recent changes in the environment. This source makes it possible for the AMR to operate in an unknown or a varying environment. The second source is the information stored in the GSM, which represents a combination of an a-priori knowledge and earlier exploration results. This information reduces the computational effort and the possibility of mismatches by a supplement of dependable hints to the sensor data processing. The third source are *hypothetical features* computed as a completion of the explored environmental description in the Predictive Spatial Completion module coupled to the DLM. These features fulfill two functions: they stabilize and accelerate the sensor data processing in the same way as the features from the GSM do and they are used to control the path planning during an exploration. Regions containing many of these features are worth of further more detailed exploration.

The DLM is a source for the most recent information about the changes in the environment. This information is stored at a low level of abstraction adapted to the capabilities of the applied sensor system. In case of a line based stereo system the DLM represents the environment in form of single lines describing the object boundaries. The stored information can directly be used in the sensor data processing without any transformations. It is also used for path planning by the navigator. The additional attributes of a feature (accuracy and confidence) in the DLM help to decide its practicability for this task.

4.2 Internal structure

The internal structure of the DLM was designed to handle a strongly changeable information from the applied sensor system. The demand to cooperate directly with the sensor data processing implies a short access time. The access time to the DLM should be negligible. It is supposed to be 10 times faster than the sensor data processing.

The aging of the information and a limited storage space result in a local description instead of a global map. This map consists of a multi-level indexing structure (fig. 3) forced by partially contradictory requirements on the DLM: fast exchange of the stored information, minimal storage space requirements and selective access to the content. As a first level we use an extended grid indexing structure. The location in the "real world" is quantified by the size of a voxel. A so–called *virtual voxel number* is mapped[1] to the size of the



0,2	1,2	2,2	3,2	4,2	5,2
0,1	1,1	2,1	3,1	4,1	5,1
0,0	1,0	**2,3**	**3,3**	4,0	5,0

Fig. 3.: *left picture* The multi–level indexing structure. (a) Torus (b) Grid (c) Octree (d) Direction–Grid (e) Linked List *right picture* Torus mapping: relation between world coordinates and virtual numbers (x, y) in the DLM

[1] The mapping is done using a modulus operation.

grid array resulting in the *real voxel number*. The real number identifies a voxel (Fig. 3b), so every voxel may correspond to different locations in the real world. This mapping mechanism is similar to that of a computer cache, where the information from the main memory (represented by the GSM) can be stored in a fast cache memory (DLM) to allow a fast access to it. The stored information is replaced only in the individual grid elements, when the AMR reaches new regions. The torus mapping provides a localized map which represents an arbitrary region in the real world. The underlying structures are optimized for efficient storage of the features (octree) and fast, selective access to the stored information (direction grid).

4.3 Data formats

Currently, we store line shaped features described by their three dimensional endpoints and additionally their orientation as the angles between them and the z–axis and the xy–plane, respectively. This information is refined in consecutive steps. Each feature in the map is additionally described by its confidence and accuracy. This information is used in navigation tasks to decide, which features should be employed for the localization.

5 The Predictive Spatial Completion

The PSC improves the exploration architecture by extracting high level features such as faces, clusters and objects and by introducing spatial reasoning and knowledge about objects in the extraction and exploration phase. The PSC converts the unrelated feature description, supplied by the DLM to a boundary description. Features are checked for their relationships and affiliation to spatial structures. Finding related features and the underlying structure always implies some uncertainty. Assignments are treated as hypotheses and have to be tested. Therefore, hypothetically missing landmarks are generated and verified in the DLM. The confirmation of the missing landmarks in consecutive sensor readings is tantamount to the confirmation of the underlying hypotheses and the assignments. These hypothetical landmarks can speed up detection and stabilization and can reduce costs for feature extraction and establishing correspondences (see section 3 and 4). The contribution of spatial reasoning in the extraction phase can further reduce the position uncertainty of the robot and improve the quality of the model. The hypothetical landmarks are generated based on three strategies that cover a high rate of common spatial structures:

Structure recognition: Benefiting to the map building as well as for navigation, the PSC clusters unrelated features such as 3D-landmarks of the line based video system and checks them against typical structures. Convex, concave, parallel or symmetrical feature groupings are utilized to build relationships among features. Higher level features are established. Higher extraction levels improve the quality of the extracted information, enable hidden line calculation for improvements in localization and can be used to facilitate the assignment of features to object hypotheses. Incomplete explored structures are completed with hypothetical landmarks. Learned statistics of the environment facilitate the assignment of related features and improves the quality and hit-rate of the generated hypothetical landmarks.

Preparation of diverse sensor data: The strategy of the DLM – verifying and fusing features by each detection – allows multi-sensor-fusion with low costs. Since the DLM fuses sensor data of one sensor type, readings of different types have to be converted by the PSC. Based on the structure recognition described above, the PSC extracts plausible structures from the detected features (semantical fusion). As sensor readings give a clue about the underlying structure, the expected structure gives a clue about the sensor readings for each sensor. These equivalent landmarks are calculated and inserted into the DLM. Correctly generated landmarks are fused with the detected ones. Different sensor ranges lead to the fact that generated landmarks are often inserted earlier into the DLM as the detected ones. This means that the position uncertainty of the robot, caused by the intermittent drive, until the generated features are in the sensor range of the equivalent target sensor, can be reduced. Since the structures are based on combination of landmarks, the generated ones can often reach a higher precision than the detected landmarks. The primitive structures are then related to each other to assemble more meaningful structures. This shall be extended to trigger the recognition of a subgroup of objects. The known object description allows to predict landmarks more complete and accurate.
Currently, laser-radar is applied to check for primitive structures and combination of structures. Hypothetical landmarks are assigned to video data. CCD features that comply to these structures are generated.

The missing dimension can often be regained (partially with delay) by checking the height of neighboring CCD features or estimated by utilizing the statistics of the environment. Applied on laser and video sensors, the generation of CCD landmarks, based on laser readings showed to be more suitable, since the laser has a higher precision and range and structures can be found with fewer computational costs (of combining landmarks) and artifacts. Since the active process of the laser-radar is further less affected by the prevailing conditions than the CCD sensor, hypothetical landmarks can be generated even under poor conditions when they gain the highest advantage.

Continuous object recognition in 3D: Knowledge about objects is commonly used for scene interpretation if the robot has to accomplish tasks on a higher level than localization or obstacle-avoidance. Compared to unrelated features, the knowledge about the presence of objects implies a valuable amount of information concerning hidden line calculation for video based localization, knowledge about solid bodies for obstacle avoidance and the association of features to objects that are important for manipulation or navigation tasks. It is desirable to identify objects as soon as possible with low computational costs.
The presented identification during the exploration presumes an approach with low computational costs that does not require separate time-consuming sensor-preprocessing. The information supplied by the sensors for the purpose of localization, exploration and obstacle avoidance is used. The costs of object recognition shall be partially regained by reducing the costs of feature extraction, correspondence calculation and map construction. A geometry-based object identification is used since 3D landmarks are available from the DLM. The recognition is geared to objects and groups of objects that are relevant for localization, obstacle avoidance and map construction that stay in the pose of their natural use. Discrimination between objects is essential since objects in indoor environments often show a similar appearance from at least one point of view or resemble a combination of other objects. Therefore, beyond their own descriptions each object implies features showing differences to similar objects. In case that these features are detected, the plausibility of the hypothesis is reduced. This speeds up the discrimination between object hypotheses and reduces the number of ambiguities to be handled until the complete identification is fulfilled.
The object description, geared to the requirements of the identification is supplied by the GSM. A common *"hypothesize and test algorithm"* is extended to the handling of groups of objects. The common prediction of 2D sensor readings (see [5], [9]) is replaced by a generation of hypothetical 3D landmarks and their insertion into the DLM independent of their visibility from the current position of the robot. Verification of these hypothetical 3D landmarks is accomplished by matching them with detected ones, stored in the DLM. Verified, corrected and falsified hypothetical landmarks are used to draw conclusions about the correctness of the object hypothesis and to refine the pose estimation. Utilizing the robot's motion for the purpose of object-recognition by taking different views of the object into account, even if the robot's behavior is not geared to this task, does not imply additional costs since the generation of hypothetical landmarks avoid recalculations of the sensor readings.

The prevailing conditions during the exploration affect the speed, accuracy and completeness of the feature detection. This results in a varying influence of the hypothesis generation. The faster and closer to completeness the features are explored, the lower are the costs of hypothesis generation and the more accurate is the clustering and object recognition. On the other hand the value of the hypotheses, expressed by the the information gain, is higher if exploration delivers features incomplete and with delay.

6 The Geometric Symbolic Model

The Geometric Symbolic Model is the global knowledge base of the system; it stores the current, reliable information about the environment in an object oriented manner. Information is accessed on various levels of abstraction by the exploration modules DLM and PSC and further perception tasks. Those tasks include e.g. localization relative to mission relevant objects or state identification for objects with kinematic degrees of freedom.
A hierarchical structure has been developed that enables access at different levels of aggregation, ranging from single unrelated features up to object class descriptions which enable generic object recognition. Because it is neither necessary nor possible to describe the complete environment of a robot in terms of distinguishable objects, a pseudo-object called *background* is introduced; it encompasses all world elements that need not or could not be assigned to any known object class.

Objects are built up recursively (see fig. 4a). They can contain so–called *member–objects*, which are connected by a joint which exhibits exactly one rotatory or translatory degree of freedom, following the conventions used in manipulator kinematics. Each object or member–object has its own coordinate system (*frame*), whose relation to that of the parent–object is described by a homogeneous transform matrix. The possible positions of a joint are normalized to the unit interval allowing a unified treatment of joint–states; additionally there exists a state called *unknown*. To deal with unknown states during a prediction, the space potentially being occupied by a moving member–object is stored as an additional boundary, called *mask*. Each branch in the object–tree carries its own boundary and feature description.

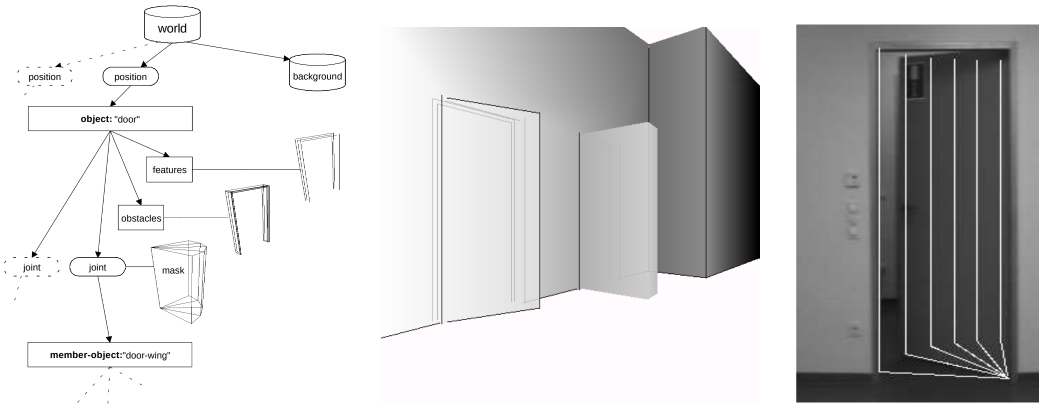


Fig. 4.: a) Model structure b) Prediction for localization c) Prediction for state identification

The most important reading access is the request of a feature prediction. Figure 4b shows the view of a corridor scene, consisting of two doors of the same class and some walls which are part of the background, as predicted for the task of video-based robot localization. Black lines denote video-specific features that are visible, grey lines hidden ones. The state of the right door wing is unknown, so the mask is predicted instead, hiding features on the wall. The localization task corrects the robot position by minimizing the distance between sensor and model features using a least squares regression method.

Feature requests can be limited to certain sensor- and task-specific feature types. This guarantees that only relevant and reliable information is predicted, which facilitates the matching process enormously. A perception task called *state identification* for example, which is charged with determining the joint states of articulated objects, focuses on so-called *rotary corners*, an aggregated feature type that contains all the information necessary to determine the joint state and which is modeled as a junction of two line segments. Figure 4c shows the result of subsequent views of such a feature for different opening angles of a door; the predicted features directly serve as starting points for the contour tracker described in section 3.1, which results in fast and robust matching and interpretation.

Writing access allows a continuous maintenance and even generation of the model in the case of dynamical and partially or completely unknown environments. The hierarchical structure of the GSM supports a gradual refinement of the environmental description as well as the insertion of newly explored information on all levels of abstraction, which fits the iterative aggregation of this information during the exploration. Inserted information can range from purely sensor-specific features, e.g. markings on objects that form reliable video landmarks, up to geometric structures that have been recognized as the instance of an object class. Structures that could not be matched to a known object class are inserted as part of the background.

Whenever a geometric boundary representation could be reconstructed by a perception task, sensor-specific features for other sensors are calculated from the boundaries using the corresponding sensor model. For a video sensor, such a sensor model is difficult to obtain because of its dependencies from various factors like color and illumination. Therefore, in a first step potential features are calculated as a set of line–segments that are based on the same vertices as the boundaries but do not necessarily coincide with boundary edges. In a second step they are compared with a set of images. Only those features that can actually be detected by the sensor are kept in the model, along with an attribute describing their detectability quantitatively in terms of how good they could be fitted to image data. This combination of geometric and sensor–specific information allows a compact representation that is easy to generate and at the same time ensures that the stored features can actually be detected by the sensor. This method is also used for model generation from CAD data.

7 Conclusion and future work

As a part of the interdisciplinary research project SFB 331 an experimental framework implying the environmental model and various perception tasks dedicated to robot localization, exploration, object registration and recognition has been realized (see [6], [7], [8]). The supported tasks are implemented as separate modules that operate in a quasi-parallel way.

The main aspect of this paper was the presentation of the underlying architecture, supporting multiple perception tasks. We presented the hierarchical sensor data interpretation process consisting of modules dedicated to fast image-processing, spatio-temporal filtering, spatial reasoning, object recognition and multi-sensor fusion, and the likewise hierarchical global knowledge base, which contains geometric as well as sensor- and task-specific representations of the environment of the robot. The two parts complement each other and thereby allow fast access to relevant and reliable information as well as sensor-based model update and generation. The modules of the system as well as their performance have been described. Future work will focus on the quantitative evaluation of the entire system as well as on the realization of applications like path planning in partially known, dynamic environments, visually guided manipulation and initial localization relative to known objects.

Acknowledgment

The work presented in this paper was partly supported by the *Deutsche Forschungsgemeinschaft* as a part of an interdisciplinary research project on “Information Processing in Autonomous Mobile Robots”(SFB331).

References

1. M. Buchberger, K.W. Jörg, and E. von Puttkamer. Laserradar and sonar based world modelling and motion control for fast obstacle avoidance of the autonomous robot mobot-iv. In *Proc. IEEE In. Conf. on Robotics and Automation*, 1993.
2. H. Bulata and M. Devy. Incremental construction of a landmark-based and topological model of indoor environments by a mobile robot. In *Proc. IEEE In. Conf. on Robotics and Automation*, pages 1048–1053, 1996.
3. D. Burschka and C. Eberst. Exploration of unknown or partially known environments. *2. Asian Conference on Computer Vision*, pages (II) 727–731, December 1995.
4. E.D. Dickmanns. Active vision through prediction-error minimization. In *Active Perception and Robot Vision*, pages 71–90. Springer Verlag, 1992.
5. P. Grandjean, M. Ghallab, and E. Dekneuvel. Multisensory scene interpretation: Model-based object recognition. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 1588 – 1595, April 1991.
6. A. Hauck and N. O. Stöffler. A hierarchic world model supporting video-based localisation, exploration and object identification. In *2. Asian Conference on Computer Vision, Singapore, 5. – 8. Dec.*, pages (III) 176–180, 1995.
7. J. Horn and A. Ruß. Localization System for a Mobile Robot based on a 3D-Laser-Range-Camera and an Environmental Model. *Proceedings International Conference on Intelligent Vehicles, Paris*, 1994.
8. G. Magin, A. Ruß, D. Burschka, and G. Färber. A dynamic 3d environmental model with real-time access functions for use in autonomous mobile robots. *Robotics and Autonomous Systems*, 14:119–131, 1995.
9. L.H. Pampagnin and M. Devy. 3d object identification based on matchings between a single image and a model. *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 1580 – 1587, April 1991.
10. A.R. Pope and D.G. Lowe. Vista: A software environment for computer vision research. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition*, pages 768–772, 1994.