Knowledge-based scene analysis using colour and range images

Benjamin Pitzer, Lars Libuda, Karl-Friedrich Kraiss

RWTH Aachen University, Chair of Technical Computer Science, Ahornstr. 55, 52074 Aachen, Germany
benjamin.pitzer@us.bosch.com, {libuda,kraiss}@techinfo.rwth-aachen.de

Abstract

Object recognition from camera images is inherently an ambiguous problem. Even when stereo vision techniques are used, it is difficult to perform robust object recognition. Humans have a broad knowledge about their environment and are able to use this knowledge to reason in unknown environments.

In this paper we present a knowledge based system to analyze single color and range images of indoor scenes inspired by human visual perception. The input images are recursively processed over four layers of abstraction resulting in a semantic scene description. A generic scene model of typical indoor environments is used as a priori knowledge. This model is encoded in semantic networks for explicit knowledge representation. The developed system is applied to images of artificial and real world indoor scenes, where it demonstrates good reconstruction rates.

Figure 1: A $STORM^3$ wheelchair equipped with a stereo vision system used for assisted and autonomous navigation.

1 Introduction

Visual scene analysis is the analysis of contents in images showing spatially limited sections of the real world. The scenes are three-dimensional, but during the capturing process, they are projected onto a two-dimensional image plane. This loss of information renders the reconstruction from images an inherently under-determined problem. Therefore the goal of *knowledge-based scene analysis* is to incorporate additional knowledge on the capturing process and on the scene itself in order to answer at least the two basic questions:

- Where are objects located in the scene?
- What objects are these?

Scene analysis based on color and range images is a important research problem, since such a system is a key component for the automation in many areas like autonomous and assisted driving, industrial manufacturing, traffic engineering, security technologies, medicine, and environmental protection.

One application for the developed scene analysis is VICTORIA [1]. The goal of this project is to create a robotic wheelchair that is able to assist its operator in difficult and unexpected driving situations. Figure 1 shows an experimental wheelchair, equipped with stereo vision and a mobile computer for image processing.

Tasks for such a system include autonomous point-to-point navigation; avoidance of collisions with walls, obstacles, and persons; avoidance of unsecure driving areas such as stairs; and assisted navigation in narrow passages such as doors. This requires the detection of at least the *floor* as drivable area, *walls* as boundaries of the floor, *doors* as passage to adjacent rooms, and the detection of static and dynamic *obstacles*.

In this paper, we address these problems and propose a theoretical and practical framework for knowledge-based scene analysis. The result of this scene analysis system is a semantic description of the scene, perceived by the camera mounted on the VICTORIA wheelchair. Our approach is fundamentally based on the principles of the human visual perception proposed by David Marr [17] and Stephen Palmer [21]. The processing is recursively grouped in different layers of abstraction and each layer depends on a certain amount of a priori knowledge. Our framework covers all layers with different processing methods. We successfully applied this framework to the problem of indoor scene analysis by using generic a priori knowledge on the architectural structure of interiors for all processing layers. The knowledge is encoded in semantic networks.

2 Previous Work

The problem of assisted and autonomous navigation in unknown environments is complex, because a large number of scientific issues are involved in this task. One main issue is the environment mapping. To navigate on drivable areas and to avoid collisions with static and dynamic obstacles a robot has to create a map of its vicinity to plan further actions. Classically this problem is solved by simultaneous localization and mapping (SLAM) of the environment. Research over the last two decades has led to impressive results; see [24] for a recent survey. Several successful algorithms emerged, among them Relaxation [5], CEKF [11], SEIF [24], FastSLAM [18], MLR [9], and TJTF [22]. Nearly all state-of-the-art methods are probabilistic and most of them are robust to noise and small variations of the environment. Despite significant progress in this area, it still poses great challenges. At present, we have robust methods for mapping environments that are static, structured, and of limited size. Mapping unstructured, dynamic, or large-scale environments remains largely an open research problem.

The outcome of classical SLAM approaches is a two dimensional map showing the accessible area. This is sufficient for simple navigation tasks. But when it comes to more complex navigation tasks, such as finding certain objects in an unknown environment or the interaction with humans, more sophisticated descriptions of the environment are required. We come to the conclusion that rather than simply sensing for obstacles and adapting to changes, we need technologies for understanding environments and the dynamics of such environments. This is only achievable if we incorporate additional knowledge for the mapping problem.

Earlier work done, for example by Kosaka and Kak [14] suggests to use Bayesian methods that combine visual cues from a monocular camera with some a priori knowledge about the geometry of a scene. The described navigation algorithm incorporates a CAD model of a building and tracks the robot position by associating visual features in the camera images, such as lines and corners with the configuration in the prior model. However, this approach fails in new environments in which a CAD model is not available beforehand.

More generic a priori knowledge has to be incorporated which is beyond the geometric information about the different objects. For example, the relationships of the detected objects to each other must be considered. The use of this knowledge finally leads to a semantic description of the scene. Liedtke and Ender demonstrated in [16] the concepts of knowledge-based object recognition. The problem was to recognize different workpieces in gray scale images. To solve this problem, a high-level symbolic description of geometric primitives, such as edges and circles and the relation between these primitives, was used to distinguish between objects. A similar approach was used by Grau in [10] to reconstruct the three dimensional structure of building exteriors and by Tönjes [25] to reconstruct landscapes from aerial photos.

3 Conceptual Model

The approach we use is based on the human visual perception. According to recent research, it is clear nowadays that we have a tremendous amount of knowledge about objects that surround us, and that we are able to use this knowledge for recognition and reconstruction in visual cues.

One theoretical model to describe the visual perception of humans is based on the ideas of Marr [17]. Palmer [21] improved Marr's theories and came up with the model displayed in Figure 2. This model constitutes the foundation for our scene analysis system.

Palmer's model is organized in four layers and

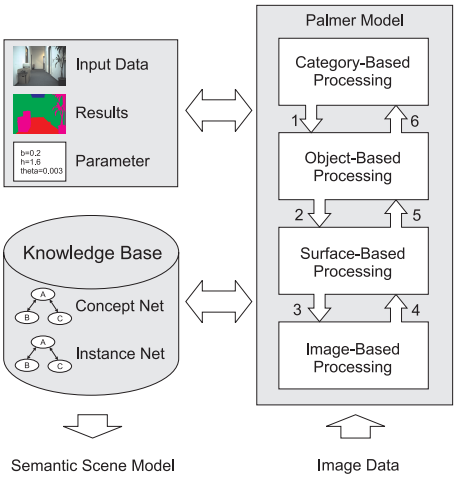


Figure 2: Image based scene analysis system.

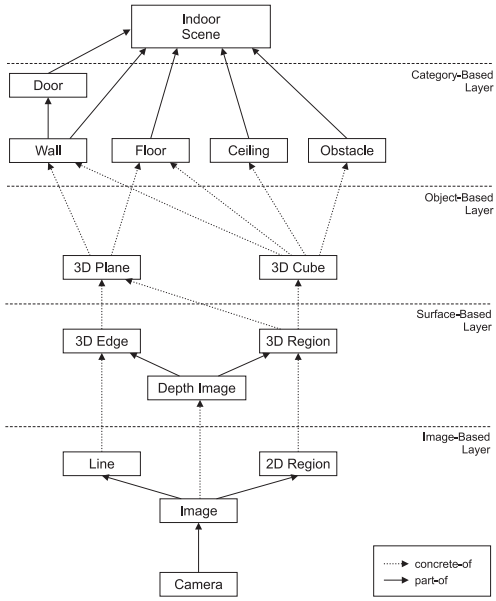


Figure 3: A subset out of the semantic concept net used for indoor scene analysis.

the degree of abstraction is increased in each layer. The retinal image of the perceived scene is the input for the *image-based processing layer*. This first stage registers the views of both eyes and extracts low-level image primitives, such as corners, edges, and homogeneous patches. These primitives are directly grouped into useful topologies like straight lines, polygons, etc. The *surface-based processing layer* provides the transition from two dimensional images to a three dimensional surface by combining multiple views. Marr denotes the result of this layer a *2.5D sketch*, because only the visible surface of objects is analyzed. A substantial three dimensional representation is created in the *object-based processing layer*. The surface of objects is complemented with assumptions on occluded parts of the scene. The *category-based processing layer* finally sorts the extracted objects into groups according to their appearance, their spatial relation, or other properties. This layer incorporates high-level knowledge on not visible properties of objects like functionality and condition. The outcome of this layer is a high-level symbolic description of the perceived scene.

The main difference between Palmer's model of the human perception and commonly used models for scene reconstruction is, that Palmer's model is a recursive model with *bottom-up* and *top-down* processing directions. This allows the human perception to make use of knowledge from higher processing layers to create low-level assumptions.

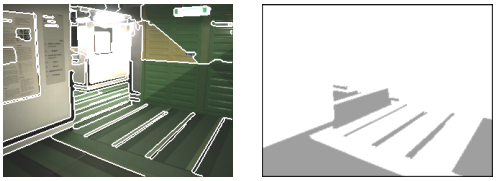
4 Knowledge-Based Scene Analysis

The developed scene analysis system is based on the conceptual model developed by Palmer. Figure 2 shows the structure of our knowledge-based scene analysis system. The *semantic scene model* is the result of the highest processing layer.

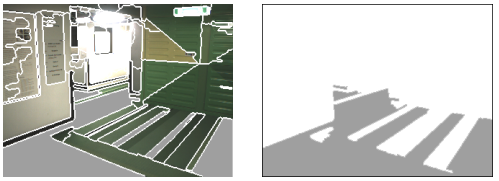
The system is based on declarative and procedural knowledge. The *knowledge base* stores and manages the declarative knowledge in semantic networks, while the procedural knowledge is used to process the image data. The following sections will outline the system's functional parts and how data is processed in each abstraction layer.

4.1 Semantic Concepts

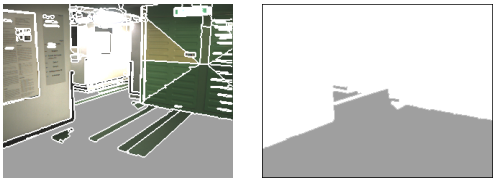
As discussed earlier, abstract a priori knowledge has to be incorporated in the scene analysis process. In our system, the knowledge is encoded in a semantic network called a *concept net* (Figure 3). The results of each processing layer are represented as nodes in this network. To connect the nodes, sev-



(a) First run without floor hypothesis.



(b) Second run with floor hypothesis.



(c) Forth run with floor hypothesis.

Figure 4: Image segmentation with and without top-down hypothesis.

eral edge types are used. The *part-of* edge decomposes a model in different parts (wall, floor, etc.), while the *concrete-of* edge provokes an organization of the model in abstraction layers. Spatial relations of scene objects are defined by edge types such as *above*, *under*, *parallel*, etc. A subset of the used network is displayed in Figure 3.

The *concept net* is a structured generic scene model, because only general knowledge on indoor scenes is used.

4.2 Semantic Instances

During the analysis process, instances are generated out of the concepts. Instances represent a connection between the conceptual model and the analysis data. The final goal is to create a consistent *instance net* with connections in all abstraction layers.

4.3 Processing

The scene analysis is a recursive process as illustrated in Figure 2. Two data flow directions are differentiated: results from lower abstraction layers are propagated to higher abstraction layers in a *bottom-up* data flow and hypotheses from higher abstraction layers are handed down to lower abstraction layers in a *top-down* data flow.

The scene analysis starts with the node *indoor scene* (Figure 3). A top-down phase (Figure 2: 1–3) creates an instantiation path for one concept, of which *indoor scene* is dependent. For example, one possible path would be:

$$Indoor\ Scene \rightarrow Floor \rightarrow 3D\ Plane \rightarrow \ldots \rightarrow Camera$$

A second phase walks through the instantiation path bottom-up (Figure 2: 4–6), and creates instances for all concepts in the path. In order to create instances, instantiation methods are assigned to each concept. For example, the concept *camera* calls a method to trigger the image acquisition and the concept *2D region* applies methods to perform an image segmentation. The result of this phase is a partial scene model. The two phases are repeated until either a consistent model is found or a certain number of iterations is exceeded.

In further processing phases, the partial scene model is used as top-down hypothesis. Figure 4 demonstrates how this helps to improve the analysis results. The first example (Figure 4a) unveils that the image is only segmented poorly (left). Consequently, only few portions of the image are correctly categorized as floor (right). In a second instantiation phase (Figure 4b) the detected regions are excluded from processing and the segmentation parameters are adjusted (left), resulting in additional floor regions (right). After four iterations almost all floor regions are correctly identified.

4.4 Image-Based Processing Layer

The *image-based processing Layer* is the first layer in the bottom-up processing. Its task is to extract the relevant image features for further processing. For the concept network shown in Figure 3, region and edge features are analyzed in this layer.

A fundamental step in all image processing applications for object recognition is to transfer the high dimensional raw image data into descriptions that are better suited for pattern recognition. Compact descriptions in terms of region and edge information are used in our model. The former is represented by the concept *2D region* and the latter by the node *line*, as shown in Figure 3.

To create instances of the *2D region* concept, an edge-oriented segmentation algorithm called *Color Watershed with Adjacency Graph Merge (CWAGM)*, proposed by Alvarado in [2], is used. This method allows us, to limit an over-segmentation of the image in large scales by means of adjustable adjacency graphs.

The second method used in the image-based processing layer is a method which extracts straight lines based on the CWAGM regions. The creation of instances of the node *line* is based on the fast line extraction algorithm by Kim et al. [12]. This method is used as an alternative to the well-known Hough-Transform [6], which suffers from complexity, coarse resolution, and lack of locality. The algorithm extracts small line segments and groups them into four categories according to their direction. A second step combines segments from the same category to straight lines.

The results of the image-based processing layer are instances of the concepts *line*, *2D region*. Also instances are created for the concept *camera*, which represents the image acquisition and for the concept *image*, which comprises the color pixel data.

4.5 Surface-Based Processing Layer

The *surface-based processing layer* is the second step towards a semantic scene description. The results of the previous layer are taken and complemented with depth information. The goal of this stage is to extract surfaces that belong to individual objects. These surface patches are represented by the concept *3D region* as shown in Figure 3.

The processing in this layer starts with an instantiation of the concept *depth image*. A block-based stereo algorithm [13] is used, to generate a disparity map. With the disparities and intrinsic camera parameters, a depth map is created. Since the depth information is sparse and noisy, several filter techniques [19, 23] are used to enhance the data.

The result of the image segmentation in the previous layer is most likely an over-segmentation of the scene, even when advanced segmentation techniques like the proposed CWAGM algorithm are used. One reason for this is that ambiguities resulting from the projection can only be resolved by incorporating additional knowledge. In this layer, we add three-dimensional information to find *2D regions* belonging to the same object, and we group them together. The used method is the density-based clustering algorithm *DBSCAN* [7]. The basic idea is to find elements that are connected through a certain density in a neighborhood.

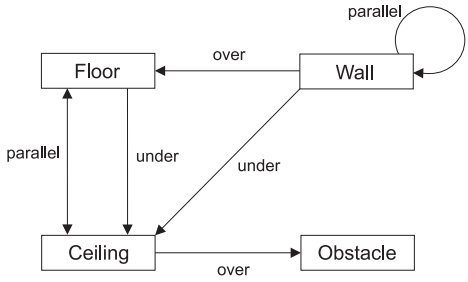


Figure 5: A subset of the Concept Net describing the semantic relations between scene objects.

4.6 Object-Based Processing Layer

For the *object-based processing layer* we have to incorporate more specific knowledge about the geometry of a scene. For indoor environments we use a “manhattan world” assumption [4] which means that the environment contains mainly orthogonal planes. In our model from Figure 3 we distinguish between two basic objects: *3D plane* and *3D cube*.

For the detection of planes a robust fitting algorithm based on the *RANSAC* [8] procedure is used. This procedure is applied to all *3D region* instances. The algorithm is composed of three successive steps:

1. **Model-Hypothesis**. Create a plane hypothesis for each *3D region* instance by randomly picking a subset of points.
2. **Model-Evaluation**. Evaluate how many other points from this region are corresponding to this hypothesis. Drop and create new hypothesis, if the number does not exceed a threshold.
3. **Model-Refinement**. Refine the hypothesis within the found set of points.

Real world environments most likely contain other, non-planar objects. Since we cannot provide more specific knowledge for such objects, they are simply modeled as a three-dimensional bounding box with instances of the concept *3D cube*.

4.7 Category-Based Processing Layer

The results of the *category-based processing layer* are finally instances of the concepts *floor*, *wall*, *ceil-*

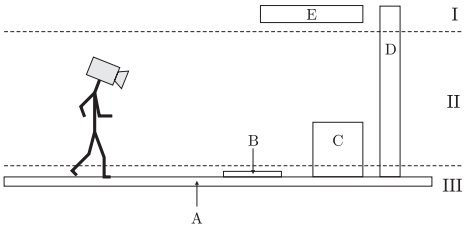


Figure 6: Categorization of scene objects by their spatial position. Only objects that are completely or partially in region II are navigation obstacles.

ing, and *obstacle* as shown in Figure 3. Algorithms and a priori knowledge for the concept *door* is momentarily not implemented in our system.

The category-based layer constitutes the highest layer of data interpretation. The classification done in this layer comprises generic architectural knowledge and specific knowledge on entities of the real world. Based on the ideas of Grau [10], this knowledge is included in the semantic concept network. Two methods are used to categorize the objects extracted in the previous layer.

The first method is similar to the work of Nüchter et al. in [20]. The relationships between the objects (i.e., *above*, *next-to*, *parallel*, *orthogonal*) are encoded using different connections in the concept net. The programming language Prolog is then used to implement and externalize the semantic net in Horn clauses. Prolog's unification algorithm attempts to assign a consistent labeling to all objects based on the clauses. Possible labels for the Prolog classification are *floor*, *ceiling*, *wall*, and *obstacle*.

All objects that couldn't be labeled by Prolog are classified with a second, rule-based approach [3]. The space is partitioned into three horizontal regions (Figure 6). Objects completely inside region III are assigned the *floor* class (A,B), while objects completely inside region I are assigned the *ceiling* class (E). Objects that are completely or partially in region II are navigation obstacles (C,D) and are assigned the *obstacle* class for simplicity.

5 Experiments

To evaluate our automatic scene analysis system we use two different environments. For the first experiment we use a textured virtual reality model to create undistorted input for the system. A stereo camera system and real office environments are used in the second experiment.

The recognition rate is calculated in both experiments by back-tracking the detected scene instances and color-labeling the corresponding image regions. The results are then compared with manually labeled images.

The *concept recognition rate* for the concepts $C = \{floor, wall, ceiling, obstacle\}$ is defined as

$$\xi_i = \frac{n_i}{N_i}\ ,\ i \in C \tag{1}$$

with n_i being the number of correct image pixels for a concept $i \in C$ and N_i being the number of all image pixels occupied by this concept. The *overall recognition rate* is then defined as:

$$\xi_{all} = \frac{\sum_{i \in C} n_i}{\sum_{i \in C} N_i}\ . \tag{2}$$

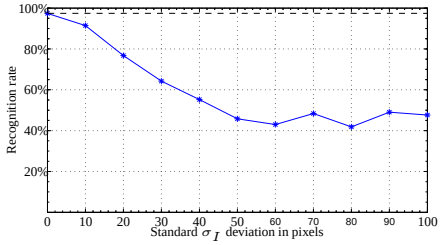
As control architecture, we use the rapid-prototyping tool IMPRESARIO [15]. The different components of our scene analysis system are realized as modules for IMPRESARIO and connected within a graph structure to generate the data flow.

5.1 Virtual Reality Environments

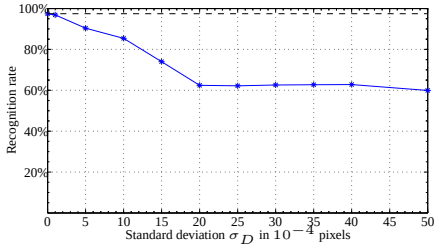
The results for scene analysis using scenes from the virtual reality environment are shown in Table 1. In this table the overall recognition rates of 12 individual scenes are compared to the individual recognition rates of the concepts *floor*, *wall*, *ceiling*, and *obstacle*.

The resulting recognition rates demonstrate that the scene analysis works very well on artificial test data; an example is shown in Figure 8. The recognition rate of obstacles (70.03%) is rather low compared to the rates of other concepts. We belive the reason is that the a priori knowledge used for this concept is vague, because the characteristics of obstacles are very diverse.

In a second experiment we add Gaussian noise with variable standard deviations to the image (σ_I) and the depth data (σ_D), in order to evaluate the robustness of our system against noisy input data. In Figure 7, the results for the overall recognition rates are shown. Expectedly, the recognition rates



(a) Noisy camera images



(b) Noisy depth images

Figure 7: Scene analysis results of virtual reality scenes with distorted color and depth images.

scene	all	floor	wall	ceil.	obst.
V_1	96.75	99.97	95.29	99.25	34.97
V_2	98.16	99.99	98.02	98.17	36.36
V_3	97.94	99.99	99.42	98.49	85.34
V_4	98.12	97.41	98.34	98.87	78.89
V_5	96.64	99.98	93.93	98.57	87.92
V_6	98.21	99.87	98.13	98.43	76.73
V_7	98.22	99.87	98.22	98.99	79.50
V_8	94.99	99.10	87.19	99.82	91.00
V_9	79.70	98.95	98.80	99.25	32.21
V_{10}	93.45	97.84	95.81	98.98	53.52
V_{11}	95.28	99.32	84.81	99.41	100.0
V_{12}	93.48	95.19	78.67	99.95	94.69
mean	**95.08**	**98.96**	**93.88**	**99.01**	**70.03**

Table 1: Scene analysis recognition rates of virtual reality scenes as percentage to manually labeled images.

scene	all	floor	wall	ceil.	obst.
S_1	90.86	95.38	87.69	$*$	87.92
S_2	78.42	94.04	61.53	$*$	40.96
S_3	74.58	98.00	43.42	91.06	46.86
S_4	76.68	95.77	69.64	87.73	$*$
S_5	79.70	93.95	72.00	81.91	$*$
S_6	72.56	95.21	61.19	$*$	82.06
S_7	72.17	83.90	65.06	$*$	86.15
S_8	75.06	98.93	71.48	$*$	34.33
S_9	68.87	91.22	54.19	39.26	46.93
S_{10}	71.64	93.71	58.96	31.52	63.73
S_{11}	65.87	94.25	61.94	$*$	67.51
S_{12}	81.24	96.58	94.48	$*$	8.22
mean	**75.64**	**94.24**	**66.80**	**66.30**	**56.47**

Table 2: Scene analysis recognition rates of real world scenes as percentage to manually labeled images. Categories with $*$ entries are not available in the data and are not considered.

are descending for higher noise levels. However, an interesting observation is that they above a certain level. This demonstrates that our system is not dependent on a special segmentation or classification algorithm. For example, even with very distorted depth information ($\sigma_D = 50 \cdot 10^{-4}$), the system is able to produce good results with just the color image.

5.2 Real Indoor Environments

For further experiments we are using real world scenes of typical office environments. For image capturing, a Videre MDCS stereo camera system [13] is used. The recognition rates for these scenes are presented in Table 2 in the same way, as in the previous section.

The overall recognition rates are significantly lower, than the recognition rates for virtual environments. One reason for this is obviously the sparse and erroneous depth information from the stereo vision algorithm, as demonstrated in Figure 10. The second reason is that the real world scenes are far more complex than the ones modeled in virtual reality. For example, scene S_1 (Figure 9) consists of many different objects, such as the big closet, the mirror, and the wash-basin. These objects are not covered with our simple semantic model. The recognition rates for the concept *floor* are very high for all scenes (94.24%). This indicates that the used a priori knowledge sufficiently covers the concept's properties.

6 Conclusion and Future Work

In this paper we have presented a framework for knowledge-based scene analysis of indoor scenes.

The developed system is capable of deriving a semantic scene description from image and range data with the use of scene independent, generic a priori knowledge. The resulting description distinguishes between the four scene elements *floor*, *wall*, *ceiling*, and *obstacle*.

We found that the system works extremely well on artificial data from virtual reality models. The overall reconstruction rate for these scenes is above 95%. This indicates that our approach points to the right direction. For real indoor scenes the overall reconstruction rate drops to 75%, which is still reasonable considering only a single point of view issued for reconstruction.

However, this evaluation demonstrates that the quantity and quality of the used knowledge is a crucial factor for the scene analysis problem. The used concept net must be extended to cover more entities of real environments.

In future work, we would like to explore an even more generic knowledge-based approach. A viable goal in the future is to create a system for diverse environments (indoor and outdoor), where only the knowledge base is exchanged according to the application. Additionally, SLAM-like mapping algorithms based on semantic descriptions would be interesting for further research.

Acknowledgements

We would like to thank the members of the Robert Bosch Corporation Research and Technology Center in Palo Alto for supporting this publication.

References

[1] http://www.techinfo.rwth-aachen.de/Forschung/MSR/Victoria/.

[2] J.P. Alvarado. *Segmentation of color images for interactive 3D object retrieval*. PhD thesis, RWTH Aachen, 2004.

[3] D. Burschka, S. Lee, and G. D. Hager. Stereo-based obstacle avoidance in indoor environments with active sensor recalibration. In *IEEE International Conference on Robotics and Automation*, pages 2066–2072, 2002.

[4] J. M. Coughlan and A. L. Yuille. Manhattan world: orientation and outlier detection by bayesian inference. *Neural Comput.*, 15(5):1063–1088, 2003.

[5] T. Duckett, S. R. Marsland, and J. L. Shapiro. Fast, on-line learning of globally consistent maps. *Autonomous Robots*, 12(3):287–300, May 2002.

[6] Richard O. Duda and Peter E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15(1):11–15, 1972.

[7] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. pages 226–231, 1996.

[8] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. In *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*, pages 726–740. Kaufmann, 1987.

[9] U. Frese, P. Larsson, and T. Duckett. A multigrid algorithm for simultaneous localization and mapping. *IEEE Transactions on Robotics*, 21(2):1–12, 2005.

[10] O. Grau. *Wissensbasierte 3D-Analyse von Gebäudeszenen aus mehreren frei gewählten Stereofotos*. ibidem Verlag, 2000.

[11] J. E. Guivant and E. M. Nebot. Improving computational and memory requirements of simultaneous localization and map building algorithms. In *IEEE International Conference on Robotics and Automation*, pages 2731–2736, 2002.

[12] E. Kim, M.i Haseyama, and H. Kitajima. Fast line extraction from digital images using line segments. *Systems and Computers*, 34(10):76–89, 2003.

[13] K. Konolige. Small vision systems: Hardware and implementation. http://www.ai.sri.com/ konolige/svs/svm.htm.

[14] A. Kosaka and A. C. Kak. Fast vision-guided mobile robot navigation using model-based reasoning and prediction of uncertainties. *Computer Vision, Graphics, and Image Processing*, 56(3):271–329, 1992.

[15] L. Libuda. Impresario – a graphical user interface for rapid prototyping of image processing systems. In K.-F. Kraiss, editor, *Advanced Man Machine Interaction – Fundamentals and Implementation*, pages 423–452. Springer Verlag, 2006.

[16] C.-E. Liedtke and M. Ender. *Wissensbasierte Bildverarbeitung*, volume 19 of *Nachrichtentechnik*. Springer-Verlag, 1989.

[17] D. Marr. *A computational investigation into the human representation and processing of visual information*. Freeman, New York, 1982.

[18] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. Fastslam: a factored solution to the simultaneous localization and mapping problem. In *18th National Conf. on Artificial Intelligence*, pages 593–598, 2002.

[19] H. Moravec. Visual mapping by a robot rover. In *6th International Joint Conference on Artificial Intelligence*, pages 599–601, 1979.

[20] A. Nüchter, H. Surmann, K. Lingemann, and J. Hertzberg. Semantic scene analysis of scanned 3d indoor environments. In *8th International Fall Workshop Vision, Modeling, and Visualization*, pages 215–222, 2003.

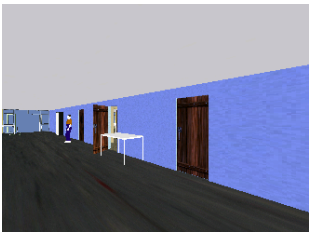
[21] S. E. Palmer. *Vision Science: Photons to Phenomenology*. The MIT Press, Cambridge, Massachusetts, 1999.

[22] M. A. Paskin. Thin junction tree filters for simultaneous localization and mapping. In *18th International Joint Conf. on Artificial Intelligence*, pages 1157–1166, 2003.

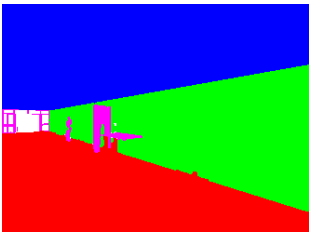
[23] L. Di Stefano, M. Marchionni, S. Mattoccia, and G. Neri. Dense stereo based on the uniqueness constraint. *16th International Conference on Pattern Recognition*, 3:657–661, 2002.

[24] S. Thrun. Robotic mapping: A survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002.

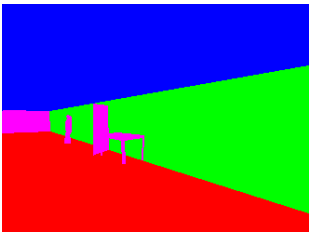
[25] R. Tönjes. *Wissensbasierte Interpretation und 3D-Rekonstruktion von Landschaftsszenen aus Luftbildern*. Number 575 in 19. VDI Verlag, 1999.



(a) Camera image of scene V_1.

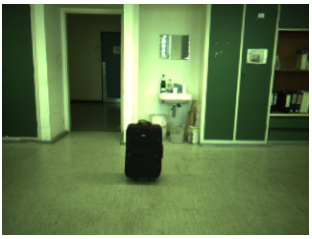


(b) Scene analysis results.



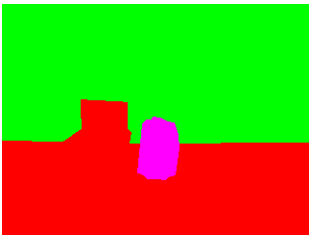
(c) Reference color labels.

Figure 8: Scene analysis of the virtual reality scene V_1. The 2D regions corresponding to the detected objects are color-labeled in *(b)*. *(c)* shows the manually labeled image used as reference. The overall recognition rate for this scene is 96.75%.



(a) Camera image of scene S_4.

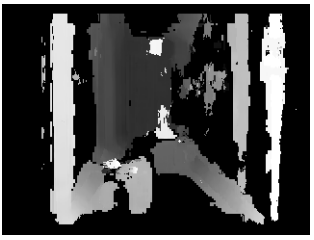
(b) Scene analysis results.



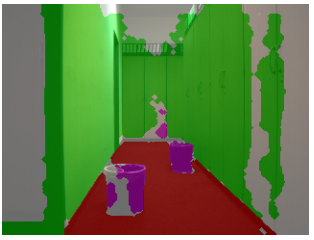
(c) Reference color labels.

Figure 9: Scene analysis of the indoor scene S_4. The regions are labeled in the same way, as in the previous example. The overall recognition rate for this scene is 76.68%.

(a) Camera image of scene S_{11}.



(b) Depth image.



(c) Reconstructed 3D scene.

Figure 10: Scene analysis of the virtual reality scene S_{11}. The depth image in *(b)* is sparse, due to homogeneous textures on the floor and walls. With our knowledge-based approach, we are able to partially reconstruct those areas as demonstrated in *(c)*. The overall recognition rate for this scene is 76.68%.