Towards Scene Analysis based on Multi-Sensor Fusion, Active Perception and Mixed Reality in Mobile Robotics

Denis Klimentjew, Sebastian Rockel, and Jianwei Zhang

TAMS Group, University of Hamburg,
Vogt-Koelln-Straße 30, 22527 Hamburg, Germany
{klimentjew,rockel,zhang}@informatik.uni-hamburg.de
http://tams.informatik.uni-hamburg.de

Abstract. The approach presented shows possible ways of improving scene analysis to achieve more reliable and accurate object recognition in the context of mobile robotics. The centralized architecture combines different feature detectors with active modalities, such as change of perspective or influencing the scene. It opens possibilities for the use of 2D detectors and extends the results to 3D. In combination with mixed reality, it offers the possibility of evaluation of the developed system as well as increased efficiency. The architecture developed and the preliminary results are presented. The work goes a step in the direction of active intelligent perception.

Keywords: Scene analysis, object detection, clustering and recognition, mixed reality, active perception, 3D modeling

1 Introduction

The detection and recognition of stationary objects is necessary in many scenarios and everyday tasks in mobile robotics. Many different algorithms and methods have been developed recently but nevertheless, no approach for stationary-object recognition meets all of the requirements of mobile robotics. Each method has its disadvantages and depends upon boundary conditions. A perfect object recognition system must be able accurately to classify an object in its environment and to deliver the object's exact position and orientation (in relation to the main robot coordinate system).

In search of the "Holy Grail" of scene analysis and object recognition, the idea of combining the use of several algorithms and methods with the possibility of changing the view and/or actively interfering with the object in the scene is the most promising prospect. The authors do not know of any existing architecture or implemented system that has this combination of facilities. There are several approaches that combine different methods, such as the combination of volumetric models and SIFT features in RoboEarth [1]. Another example is the so-called

[1] http://www.roboearth.org/

project recognition kitchen from Willow Garage [2]. The project consists of two different pipelines of 2D and 3D data processing algorithms. Each pipeline combines different algorithms.

Many existing methods for simultaneous recognition of objects try to match features using a database [1][2] and may also integrate stereo cameras, time of flight cameras, thermal cameras, etc. [3]. However all of these methods have limitations; one method, for example, works only with rotationally symmetrical objects. Other algorithms cannot handle unknown objects and yet others work only with unknown objects.

The approach we present tries to extend the existing state of the art and shows a possible way to improve scene analysis and to achieve more reliable and accurate object recognition. This paper presents a novel approach to increasing object recognition efficiency by simultaneously using multiple algorithms and by active interference with scene objects. The possible architecture and the first preliminary results are presented. The work goes a step in the direction of active intelligent perception.

2 Possible Approach And Architecture

While based on well-known object recognition methods, the work presented uses a new architecture to investigate optimal object recognition strategies. It includes novel approaches, such as changing the view and actively interfering with the environment.

Figure 1 presents the object recognition architecture including both new and traditional means of object detection. The upper part illustrates the perception of the complete environment and the regions of interest (ROI). The scenario is limited to an indoor scene, such as an office environment. In this case the ROI's are planar surfaces, places where interesting objects are most likely to be found. There are many existing approaches to 3D environment mapping. We use our own approach [7] that navigates in the 2D environment and automatically creates a 3D map. The advantage of this approach is that it guarantees that all surrounding objects are mapped. On detection of the ROI's the robot will navigate to those locations according to the assigned task. The upper part of the architecture shows the state in which the robot has been placed in front of the ROI.

The lower part of the architecture starts with "sensor fusion". The aim of gaining as much information as possible is achieved by using many different 2D and 3D feature detectors. The approach used is introduced in the section 3 and combines rgb and depth information. According to future requirements, further sensors can also be adopted, for example other cameras or laser range finders. The combination of depth and color information allows many more possibilities for scene analysis. After sensor fusion, planar surfaces are detected and removed as soon as the 3D data is segmented. Further processing steps (to reduce the

[2] http://ecto.willowgarage.com/recognition/user.html

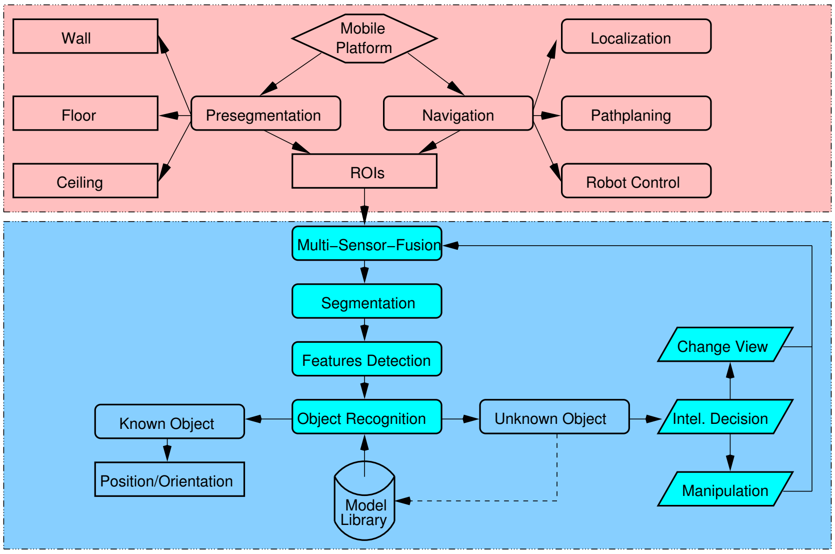

Fig. 1. Visualization of developed architecture.

complexity) operate only on recently detected clusters. In the next step the 2D/3D feature detectors can be applied, singly or in combination; as single components; in combination with other methods of the same kind; and as combinations of different methods. Of course, each additional method increases the possibility of successful scene analysis. However, when no detector succeeds, the system must decide between changing the view perspective and active influence through a manipulator in order to change the spatial relationships between objects under investigation. In either case, data acquisition, fusion and clustering must be repeated before feature detection can be reapplied.

It is clear that the system is powerful and can deliver more usable results than state-of-the-art methods. An open question is the selection of an optimal, efficient strategy within the presented architecture. Success depends upon on many parameters, the number of which grows with the number of methods included and combined. A mathematical calculation of the best capabilities is not given; the problem is NP complete. The authors suggest that the use of physical simulation and mixed reality can help to reduce problem complexity and to find an efficient solution. More details are presented in section 5.

3 Multi-Sensor Fusion

Before defining the object recognition process, the necessary input has to be specified. The more information the input includes, the higher the probability of finding characteristic features in it. We concentrate on the Kinect-like sensors (see section 6), combination of structured light (PrimeSense sensor) and rgb images.

The pseudo-code presented below emphasizes the multi-modal fusion of both sensors. The calibration process uses a calibration body consisting of a colored planar surface and some holes with predefined alignments. The calibration body is shown in figure 2.

Algorithm 1 The calibration algorithm for Kinect-like sensors

```
procedure CALIBRATION(RGB-D)
    Synchronized data acquisition of RGB and depth information.
    RGB:
    Transform to HSV color space.
    Generate a histogram.
    Sorting to white, black, gray, and other pixels.
    Fore- / background separation.
    Reduce to ROI (calibration body in foreground).
    Finding a rgb-holes.
    Sorting a rgb-holes.
    Depth:
    Transform the distances to standard values ( m)
    Generate a binary image (if depth between 0.60 m − 1.2 m then 1 else 0).
    Reduce to ROI through remove the lines (if number of 0 values > 99 %).
    Finding a depth-holes (growing).
    Sorting a depth-holes.
    Both:
    Check and remove of false-positives.
    Calculation of hole centers.
    Calculation of and finding the minimal distance between RGB- and depth hole
  centers.
    Generate the corresponding pairs.
    Calculation of homogeneous transformation matrices (with help of RANSAC,
  7/8 point, and LMEDS algorithms).
    Return the homogeneous transformation matrices.
end procedure
```

After the detection of corresponding points (center of circles in rgb and of holes in depth images) the fundamental matrix can be calculated. The RANSAC, 7/8 Point and LMEDS (Least Median of Squares) algorithms have been implemented and tested with the best results delivered by the RANSAC algorithm. Eventually the user receives a colored point cloud with to each other registered depth and color information.

Evaluation of the calibration method is difficult. On the one hand, the information content of one voxel grows with distance; on the other hand, the perspective change (baseline) has negative character (parallax effect). In practical applications, the quality of the method has been verified and shows great potential. The algorithm is fast and simple to use: the user just has to place the calibration

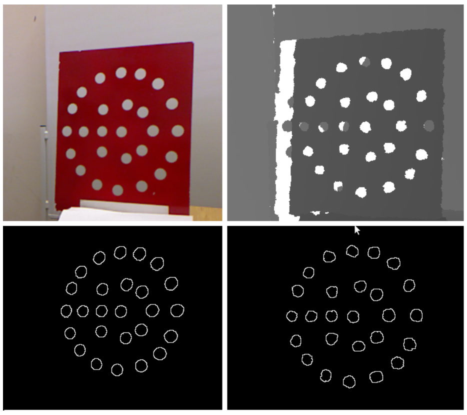

Fig. 2. Upper images shows the rgb and depth image respectively. The lower images visualize the found holes in rgb and depth image.

body $0.60\,m - 1.2\,m$ in front of the sensor and start the application. Hence calibration process is completely automated and can easily be repeated on demand. The resulting homogeneous transformation matrix is automatically updated and the platform is ready to use.

4 Approach

In the work presented, the 3D image will be seen as a one dimension expanded 2D image. Consequently the same model is used for 3D point clouds as is used for an image. So an image I can be seen as the sum of several components:

$$F_{i,f}(\overrightarrow{x}) = \alpha \cdot BG_{i,f}(\overrightarrow{x}) + N_{i,f}(\overrightarrow{x}) + T_{i,f}(\overrightarrow{x}) \qquad (1)$$

or simplified

$$F_i = T_i + BG_i + N_i \qquad (2)$$

where F denotes the image, T_i the foreground region, BG_i the background, and N_i the camera noise at frame number i. The advantages of this approach are the retention of the relation between neighboring voxels and the possibility of using the know-how of 2D image processing and the many existing pertinent algorithms [4].

4.1 Preliminary work

The calibration procedure described below delivers colored point clouds. The results can be seen in the middle image of figure 3. To reduce complexity and increase efficiency, planar surfaces are recognized and removed in the first step. Afterwards, cluster segmentation is applied to the remaining voxels. The methods based on calculation of Euclidean distance shows good results and excellent performance. Such found clusters are used for further processing. Of course, the clusters can be under segmented and hence can contain more than one object, specifically in case of occlusion. Nevertheless, the architecture allows the use of any suitable segmentation algorithm. The authors see pre-segmentation as a precondition for applying the feature detectors, because the clusters offer an excellent basis for further processing.

Naturally, the characteristics of the sensors in use affects the course of object recognition. In the present case the depth resolution of the PrimeSense sensor is only $\frac{1}{4}$ of its color resolution. That is a reason for the many black pixels in the middle image of figure 3. This circumstance prevents the application of some image processing algorithms or can falsify their results. To compensate for this disadvantage, the method presented transfers the bounding boxes of each cluster to the color image. The results are the ROI's from the color image.

For the calculation, the homogeneous and projecting matrices are used. The procedure consists of the following steps: At first, under assumptions of ideal conditions, the so-called pinhole camera model is used. The focus lies on the projection of an object point $(x, y, z)^T$ to the position $(u, v)^T$ in the image frame. The procedure starts with the transformation of an object point $(x, y, z)^T$ in the camera frame to the point $(x', y')^T$ in the image frame with the focal length of $f = 1$. With this, $x' = \frac{x}{z}$ and $y' = \frac{y}{z}$. Thereby the emitted brightness of the object point $(x, y, z)^T$ will be mapped to the position $(x', y')^T$.

The same layer will also be used for the calculation of lens distortion. In order that the point at position $(x', y')^T$ is transformed to $(x'', y'')^T$.

$$x'' = x'(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 x' y' + p_2(r2 + 2x'^2) \tag{3}$$

$$y'' = y'(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2y'^2) + 2p_2 x' y', \tag{4}$$

where $r^2 = x'^2 + y'^2$; (k_1, k_2, k_3) are the coefficients of radial and (p_1, p_2) of the tangential distortion.

Finally the points of the distorted image are mapped, with the help of intrinsic parameters, to position $(u, v)^T$ of the 2D camera sensor.

$$u = f_x x'' + c_x \tag{5}$$

and

$$v = f_y y'' + c_y, \tag{6}$$

where (c_x, c_y) are coordinates of the principal point and (f_x, f_y) is a principal distance in pixel.

Consequently the procedure finds the corresponding ROI inside the original color image, providing that the parameters of intrinsic calibration are correct. The work on the ROIs reduces the run-time and increases the efficiency. Figure 3 visualizes the results of clustering, depth and color segmentation. On the left side the original image is presented. The middle image shows the depth segmentation with the projected color information on it. On the right is the corresponding

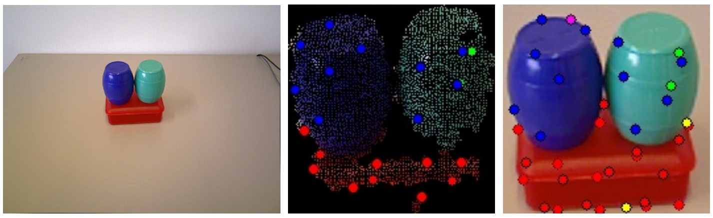

Fig. 3. Visualization of results for clustering, depth and color segmentation. The left image shows the original scene. The image in the middle visualize the depth segmentation with the registered color information. Rights the corresponded ROI taken from the original color image.

ROI from the original color image. With the naked eye it can be seen, that the two images contain more information than only one of them. The single colored circles visualize color information found at the corresponding positions.

The presented calculation projects a bounding box from the point clouds to a corresponding box in the image. Unfortunately, there is no possibility to reconstruct depth information from the 2D image. Nevertheless for the use of the results from the 2D image processing algorithm this possibility is strictly needed. To provide the depth information, a structure like $[x_1, y_1, z_1, r_1, g_1, b_1, \ldots, x_n, y_n, z_n, r_n, g_n, b_n]$ is created and filled using the bounding box transformation. This allows each feature detector to switch permanently between 2D and 3D images and to achieve the best possible results. For example the 2D edge detector is applied to the image. This kind of algorithms is robust and very fast. Nevertheless the methods search for image brightness changes and produce a lot false-positives. To improve the results the edges detected can be evaluated with the help of depth information. In this case the fast 2D algorithm is used as a surface based method. The 3D method can be applied only on interesting regions, thus the complexity can be reduced. Moreover at each step of the architecture the manipulator can be included to enable dedicated manipulation inside the scene.

In almost the same manner as using 2D and 3D information the system needs a centralized place for all object relevant information. To provide the necessary information a database is needed. This database must hold the following attributes per item: 3D mesh, images, color and other necessary data belonging to an object. The resulting system shall have the ability not only to read the data from the database, but also to complete the data for one object or insert the new objects into the database during run-time.

4.2 Single components / features detectors

One of the difficulties with Euclidean distance based methods applies when the distance between objects is below a specified threshold. An over sized threshold results in under segmentation. On the other hand, the threshold depends on the distance between sensor and object and must be not too low. One extreme example of another difficulty represents an occlusion, in this case the clustering is useless and the employment of other 2/3D feature detection algorithms is necessary. In resent years two 2D feature detection algorithms have been widely used. The SIFT [5] and the SURF [6] algorithms are robust and fast and deliver practical results. Both feature detectors are similar to each other. The difference between them is that the SURF Detector uses a 2D Haar wavelet that makes it possible to use the integral images. Therefore the SURF is faster compared to the SIFT detector and has constant run-time.

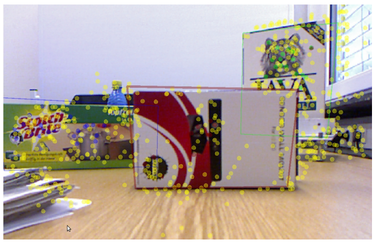

Fig. 4. Object detection with the SURF/SIFT feature detector.

Figure 4 visualizes object recognition with the SIFT feature detector. The methods can compensate partial occlusions under the condition that enough features are found. Three objects are recognized, the recognition of the bottle in the background fails, because the insufficient features are found. The bounding boxes are correctly transformed to the input image despite partial occlusions. Objects with complex texturing are excellently suited qualified for this kind of method.

Fundamentally, all possible algorithms can be included and used with the developed architecture. The algorithms considered are based on edge and corner detectors, color recognition, silhouettes, principal component analysis (pca), etc. Most of these methods are well-known, efficiently implemented and fast. Through ROS [3] as a common framework the integration of methods is simple and consistent. The big advantage lies in the possibility to use algorithms with different input information such as color, texture, size, shape, and so on. So information about different object properties can be obtained and fused. An overview of different 2D methods with significant comparison can be found in [8].

Object recognition in 3D point clouds is a relatively new research field. The input data from the pure point cloud are voxels and the neighbor relations between them. Therefore after the clustering, the size, curve and shape can be compared. The changes between normal vectors helps to determine information about curve and shape as well as possible edges and corners. There are many algorithms to compare the similarity between point clouds. The common methods are based on the so-called Iterative Closest Point (ICP). An examination of different ICP based algorithms can be found in [9].

[3] http://www.ros.org

The more detectors that are implemented, integrated and used, the higher the probability of the successful scene analysis. Nevertheless the developed architecture opens a possibility to go a step further through the combination of different methods. It starts with edge and corner detection in 2D and verification of the resulting hypothesis in 3D. But more complex combination are thinkable. Also the manipulator may be integrated. So the recognized object can be removed in the scene. Recursive application is promising and is under study.

The next interesting point is the handling of unknown objects. The fused sensor information and the mobile platform can be used to create the necessary information for the database. The robot can move around the table or rotate the object with the manipulator. The resulting data, such as a registered 3D model or images from different perspective, can be stored directly in the database. The absent information, such as the category, can be applied from the user or a learning algorithm.

Anyway these possibilities are not utilized. What can be done if despite the large number of detectors and their combinations, no useful information deliver? The architecture provides two further possibilities. The first is a change of perspective, after which the scene analysis can be restarted. The second is active influence of the scene to change the correlation between objects. Re-application of the analysis algorithms is promising, too.

5 Scene Analysis and Mixed Reality

When the robot meets unknown scenes it needs a strategy for further analysis. In this work we define that a scene contains a region of interest (ROI), i.e. a robot with sensors, a table and an unknown amount of objects on it. To further specify the domain we restrict objects to be of a-priori known models in a data base. Even with that knowledge, trial and error to find the best view on the scene, manipulate it or test multiple feature detectors is expensive in the real environment. Expensive here can be seen in terms of time and electricity for the robot. To reduce these 'costs' we combine in this work a simulation of the scene with the real-word environment and thus employ a mixed reality scenario [10] where the simulation is updated by real-world sensing data and simulation results are employed in the real environment.

In this section multiple approaches of exploiting the simulation in order to gain an efficient strategy for the scene analysis will be introduced. Finally the approach used will be introduced.

5.1 Generation/Extraction of difficult Scenes for Evaluation

With the help of simulation trails and depending on specific criteria, 'difficult' scenes will be selected out of randomly generated situations. Only these difficult scenes will then be evaluation in the real environment with the presented system.

Difficult scenes include situations where objects occlude each other or are stacked on top of each other.

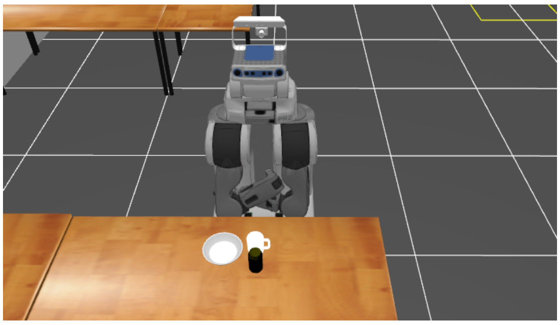

Fig. 5. Example scene in the simulation. Here a special situation is shown: object occlusion

5.2 Generation of a Scene Memory

With the a-priori defined objects in the database, scenes with random combinations of objects are created. Each scene is stored in a 'global memory' and this memory is then online-searched for certain criteria to find appropriate scenes in the memory fitting the currently observed situation in the real environment.

5.3 Finding the next best view

Often scenes with unknown objects on top of a table cannot sufficiently be analyzed because of the perspective, i.e. the view does not allow the scene to be completely segmented. Finding the next best view (NBV) is a problem that many scientists in the graphics and robotics community have faced. Here based on the volumetric model any approach to solve the NBV can be applied. Therefore the 3D point cloud will be used as an input to define the volume. The next best views will then be calculated according to specific criteria. In order to validate these views each has to be tested with the robot. A view includes a goal position for the robot and a position is constrained by the robot shape and its environment, e.g. a table. Instead of giving the robot the goal position of the next best view we validate it first in the simulation. If for some reason the navigation cannot proceed to this position, e.g. because it is too close to an obstacle, it will be adapted and tested again. Eventually a valid goal position will be found and passed to the real robot in order to find a trajectory and proceed to the next best view. The NBV problem is a local optimization problem and NP-hard. It is often solved approximately by an algorithm selecting the view that maximizes a view metric [11]. Whereas in state-of-the-art NBV solutions, it is a challenge to evaluate the view metric for a large set of views, in the simulation each view can be evaluated by testing the planned view according to the feature detectors applied. The domain of possible views has to be restricted nonetheless. Defining NBV algorithms is not the focus of this work.

5.4 Finding a strategy to successfully manipulate the scene

In order to grasp an object or to move it the robot situation has to fulfill some pre-conditions. First it has to be in reach of the object which is defined by the robot arms' workspace. If the object is not within reach, no manipulation can be performed. The problem can be reduced to that of finding a suitable manipulation pose. The simulation allows the testing of different robot poses and hence the choice of the optimal pose.

6 Implementation and Evaluation

The work presented benefits from the use of ROS (robot operating system) as a framework. For the evaluation the PR2 [4] robot from Willow Garage is used. ROS provides algorithms for object detection based on the RANSAC method, for example table detection, as well as Euclidean distance using segmentation. Other advantages of using ROS are the integrated image processing library, OpenCV [5], and the 3D processing library PCL [6]. Parts of the algorithms are used directly, while others have been adapted or integrated as part of this work. The ROS structure allows all feature detectors and other modalities to be started in parallel, limitations arising only from the available computational capacity.

In this work we concentrate on Kinect-like sensors (ASUS Xtion Pro Live in the case of our PR2). The advantage is the high acquisition rate of 30 fps for color and depth information simultaneously. Fusion is implemented in C/C++ for hole detection and rgb-images. For the calculation of the fundamental matrix the OpenCV library is used. After calibration, the results are immediately available inside the ROS structure.

The database is realized in PostgreSQL [7], with versions 8.4 and 9.2 integrated and tested. It is possible to connect, extend and modify the database during runtime. Inside the database, the size, color, and if available, the weight and balance information, as well as the 3D model, images taken from different perspectives and SIFT/SURF bag for each object are available attributes.

The first evaluation idea is to show that the implemented part of the architecture already goes beyond state of the art methods. Therefore three different scenarios are created to show the possibilities of the work. All of the scenarios have partial occlusion and the first segmentation delivers an under segmented cluster. In the first scenario one of the feature detectors recognized an object as a part of the cluster. The manipulator is used to grasp the object and put it away. After re-applying the developed system the rest of the objects can be successfully recognized. Figure 6 visualizes the individual steps of the first scenario.

The second evaluation scenario used the possibility to change the perspective. The colored point cloud after the robot has moved allows individual objects to

[4] http://www.willowgarage.com/pages/pr2/overview
[5] http://opencv.willowgarage.com/wiki/
[6] http://pointclouds.org/
[7] http://www.postgresql.org/

Fig. 6. The left upper image shows the under segmented cluster, the recognition of single objects is not possible. The SIFT/SURF detector recognized the bottle (bounding box in the right upper image). In the left lower image the manipulation with the PR2 is visualized. The robot grasps a bottle and removes it from the scene. After re-applying the segmentation and feature detection the rest of the objects can be recognized.

Fig. 7. The left image shows the partial occluded scene. The segmentation delivers an under segmented cluster. After the perspective change the recognition can be applied successfully. The right images show the visualization of a successful recognition in the image (top) and the point cloud (down).

be segmented and recognized using icp-based comparison with the database, as shown in figure 7.

For the third scenario, no practical information can be obtained from the feature detectors. As a last resort the manipulator influences the scene and changes the relation between the objects. After re-applying segmentation and feature detectors the objects can be recognized. Figure 8 shows the individual steps as well as the results.

These experiments show clearly that the developed architecture has great potential. Of course, many detectors and/or segmentation algorithms can and must still be integrated and the system can be improved. The right hand image

Fig. 8. The left image shows the under segmented scene, the feature detectors deliver no useful information. The recognition is not possible. The middle image visualizes the influence in the scene with the manipulator. After re-applying the segmentation and feature detectors the objects can be recognized.

in figure 8 shows that the orientation of the left cup is inaccurate. Consequently, it is possible that the calculation of grasp can fail. Nevertheless, the system opens enormous possibilities for mobile robotics.

It is conceivable that with the growing number of detectors as well as segmentation the system becomes complicated. Not all components can be run in parallel, because not enough computation power can be provided. The same problem results from the possible combination and chaining of single components. From this point of view, the search for the best possible strategy for scene analysis is one of the open problems, as already described in section 5. The next open questions are the evaluation and its required metrics. The authors think that the use of a physical simulation and the mixed reality approach can help an efficient solution to be found. At first the system can be evaluated with or without involving any hardware components. It can be examined with ideal sensors as well as with noisy real sensor data. Moreover, mixed reality can be seen as an omniscient concept. According to this, a meaningful scene with interesting relationships between objects can be selected. Using this, the optimal/best possible strategy can be found and verified under realistic conditions. The evaluation approach, using mixed reality, improves during run-time. It can be used also to evaluate similar architectures. Moreover the evaluation concept can be applied to compare different systems or implementation approaches.

7 Conclusion And Future Work

The paper presents a new approach to the improvement of scene analysis and object recognition. The architecture developed allows different modalities and components to be combined in a centralized manner. Of course, one of the next steps is to produce a well integrated system. Furthermore, the mixed reality approach can be used to evaluate the system and to make it more efficient. The final step is to create a global evaluation metric to compare this kind of system to others. The first step is now done and the authors are working hard to realize all of their proposed ambitions.

The first results show promising data and the authors hope to achieve the next step in the direction of active intelligent perception.

Acknowledgment

The authors would like to thank William Morris for his structural feedback and Robert Wieczoreck for supporting the implementation of the calibration procedure.

This work has been conducted as part of RACE, funded under the European Community's Seventh Framework Programme FP7-ICT-2011-7 under grant agreement n 287752 [8].

References

1. R.B. Rusu, G. Bradski, R. Thibaux, J. Hsu, Fast 3D Recognition and Pose Using the Viewpoint Feature Histogram, Proc. IROS, 2010
2. Sun, Xu, Bradski, Savarese, Depth-Encoded Hough Voting for Joint Object Detection and Shape Recovery, Proc. ECCV, 2010
3. R.B. Rusu, N. Blodow, Z.C. Marton, M. Beetz, Close-range Scene Segmentation and Reconstruction of 3D Point Cloud Maps for Mobile Manipulation in Domestic Environments, Proc. IROS, 2009
4. D. Klimentjew, J. Zhang, Adaptive Sensor-Fusion of Depth and Color Information for Cognitive Robotics, In Proceedings of the IEEE International Conference on Robotics and Biomimetics (IEEE-ROBIO), Phuket, Thailand, December 7-11, pp. 957-962, ISBN: 978-4577-2137-3, 2011.
5. D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision, vol. 60, pp. 91-110, 2004.
6. H. Bay, T. Tuytelaars, L. Van Gool, Speeded Up Robust Features, Proceedings of the 9th European Conference on Computer Vision, Springer Verlag, 2006.
7. G. Michalicek, D. Klimentjew, J. Zhang, A 3D Simultaneous Localization And Mapping Exploration System, In Proceedings of the IEEE International Conference on Robotics and Biomimetics (IEEE-ROBIO), Phuket, Thailand, December 7-11, pp. 1059-1065, ISBN: 978-4577-2137-3, 2011.
8. P.M. Roth, M. Winter, Survey of appearance-based methods for object recognition, Technical Report ICG-TR-01/08, Graz University of Technology, Austria, 2008.
9. S. Rusinkiewicz, M. Levoy, Efficient Variants of the ICP Algorithm, Third International Conference on 3D Digital Imaging and Modeling (3DIM 2001), Quebec City, Canada, 2001.
10. S. Rockel, D. Klimentjew, J. Zhang "A Multi-Robot Platform for Mobile Robots - A Novel Evaluation and Development Approach with Multi-Agent Technology", In Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), University of Hamburg, Hamburg, Germany, September 2012.
11. Kok-Lim Low, Lastra, A. "Efficient Constraint Evaluation Algorithms for Hierarchical Next-Best-View Planning" 3D Data Processing, Visualization, and Transmission, Third International Symposium on , vol., no., pp.830-837, 14-16 June 2006, doi: 10.1109/3DPVT.2006.52

[8] http://www.project-race.eu/