MACS

**FP6-004381-MACS**

**MACS**

Multi-sensory Autonomous Cognitive Systems Interacting with Dynamic Environments for Perceiving and Using Affordances

Instrument: Specifically Targeted Research Project (STReP)

Thematic Priority: 2.3.2.4 Cognitive Systems

# D3.2.1 Multi-sensor Affordance Recognition

Due date of deliverable: June 30, 2006
Actual submission date: August 15, 2006

Start date of project: September 1, 2004 Duration: 36 months

**Fraunhofer Institute für Intelligene Analyse- und Informationssysteme (FhG/AIS)**

Revision: Version 2

| **Project co-funded by the European Commission within the Sixth Framework Programme (2002–2006)** | | |
|---|---|---|
| **Dissemination Level** | | |
| **PU** | Public | **X** |
| **PP** | Restricted to other programme participants (including the Commission Services) | |
| **RE** | Restricted to a group specified by the consortium (including the Commission Services) | |
| **CO** | Confidential, only for members of the consortium (including the Commission Services) | |

EU Project MACS

Deliverable D3.2.1

# Multi-sensor Affordance Recognition

*Erich Rome, Lucas Paletta, Gerald Fritz, Hartmut Surmann, Stefan May, Christopher Lörken*

| | |
|---|---|
| *Number:* | **MACS/3/2.1** |
| *WP:* | *3.2* |
| *Status:* | *version 2* |
| *Created at:* | *July 17, 2006* |
| *Revised at:* | |
| *Internal rev:* | *v7 – August 15, 2006* |

| | |
|---|---|
| **FhG/AIS** | Fraunhofer Institut für Intelligente Analyse- und Informationssysteme, Sankt Augustin, D |
| **JR_DIB** | Joanneum Research Graz, A |
| **LiU-IDA** | Linköpings Universitet, Linköping, S |
| **METU-KOVAN** | Middle East Technical University, Ankara, T |
| **OFAI** | Österreichische Studiengesellschaft für Kybernetik, Vienna, A |

This research was partly funded by the European Commission's 6th Framework Programme IST Project MACS under contract/grant number FP6-004381. The Commission's support is gratefully acknowledged.



**Corresponding author's address:**

Dr.-Ing. Erich Rome
Fraunhofer Institut für Intelligente
Analyse- und Informationssysteme
Schloß Birlinghoven
D-53754 Sankt Augustin, Germany

| | | |
|---|---|---|
| Fraunhofer Institut Intelligente Analyse- und Informationssysteme | Fraunhofer Institut für Intelligente Analyse- und Informationssysteme<br>Schloss Birlinghoven<br>D-53754 Sankt Augustin<br>Germany | Tel.: +49 (0) 2241 14-2683<br>(Co-ordinator)<br><br>**Contact:**<br>Dr.-Ing. Erich Rome |
| JOANNEUM RESEARCH | Joanneum Research<br>Institute of Digital Image Processing<br>Computational Perception (CAPE)<br>Steyrergasse 9<br>A-8010 Graz<br>Austria | Tel.: +43 (0) 316 876-1769<br><br>**Contact:**<br>Dr. Lucas Paletta |
| LINKÖPINGS UNIVERSITET | Linköpings Universitet<br>Dept. of Computer and Info. Science<br>Linköping 581 83<br>Sweden | Tel.: +46 13 24 26 28<br><br>**Contact:**<br>Prof. Dr. Patrick Doherty |
| Middle East Technical University | Middle East Technical University<br>Dept. of Computer Engineering<br>Inonu Bulvari<br>TR-06531 Ankara<br>Turkey | Tel.: +90 312 210 5539<br><br>**Contact:**<br>Asst. Prof. Dr. Erol Şahin |
| Ö F A I | Österreichische Studiengesellschaft<br>für Kybernetik (ÖSGK)<br>Freyung 6<br>A-1010 Vienna<br>Austria | Tel.: +43 1 5336112 0<br><br>**Contact:**<br>Prof. Dr. Georg Dorffner |

# Contents

# 1 Introduction

This deliverable describes a part of the development of a software that supports the perception of affordance-related *cues*[1] by a multi-sensory mobile robot system. As early as possible, we want to point out that the much earlier defined title "Multi-sensor affordance recognition" is considered to be no longer appropriate under the light of the current definitions and terms used in the MACS project. Today, we would rather title it "Multi-sensor affordance perception", since affordances can only be perceived, not recognized.

In this deliverable, we focus on the special properties and possibilities of the variety of sensors available on the robot, namely odometry, cameras, and a multi-modal 3D Laser scanner, to name a few. The main uses of the data produced by the multi-sensor setup are located in two modules, namely the behavior system and the perception module [1].

In the behavior system, multi-sensory data are used by control routines that implement robot behaviors and actions. These behaviors and actions constitute basic "skills" that control the robot while acquiring knowledge about affordance relations, as well as more complex behaviors that can be tuned by other modules in the control architecture.

In the perception module, multi-sensory data are used by feature extractors to provide the features for the learning of affordance relations (bottom-up perception) and features for active perception and search of cues for the presence of affordances (top-down perception).

Both modules, i.e. the behavior system's control routines and the feature extractors, also apply *sensor fusion* methods to produce more reliable data and features.

The remainder of this document is structured as follows. After specifying the related documents, we start the technical part by explaining the concepts and methods we use for bottom-up perception of affordances (top-down perception will be described in another deliverable). We continue with describing the robot sensors and their characteristics. In the next section then describes some of the experiments we have performed in order to illustrate bottom-up perception. We began with applying feature extractors on visualizations of our experimentation environment (the demonstrator scenario setup), for the case of "liftability", i.e., perceiving that a test object is liftable. In order to adapt the methods to real world data, we have built the test objects shown in the visualization and recorded multi-sensory data in real setups. The deliverable is concluded by summarizing the state of work and the next steps.

# 2 Related documents

**D1.1.2** "Specification of Module Interfaces" [2]
**D2.1.1** "Identification of architectural requirements of an affordance-based control" [3]
**D2.2.2** "Development of an affordance-based control architecture" [1], includes definitions and glossary of terms
**D3.1.2** "Affordance recognition from visual cues" [4]
**D3.1.3** "Saliency detection with visual attention" [5]
**D4.2.1+4.3.1:** "Tentative Proposal for a Formal Theory of Affordances
Tentative Proposal for an Affordance Support Architecture
Prototype: Affordance-Based Motion Planner" [6]

[1]A glossary of terms and definitions are contained in deliverable D2.2.2 [1].

**D5.3.1** "Robotic learning architecture that can be taught by manually putting the robot through action sequences" [7]
**D6.1.1** "Specification of final demonstrator" [8]
**D6.4.1** "Report on experiment design" [9]
**Technical report:** "Specification of a Prototype Behavior System" [10]

# 3 Affordance Perception Concepts

A (robot) affordance has been defined as a possibility for action that the environment offers an agent, where the agent in turn has the capability to act upon it. It is an agent-specific relation between features of the environment that indicate the existence of an affordance and the results of the agent when acting upon it. Definitions and more detailed descriptions of the MACS approach to affordance-inspired robot control can be found in D4.2.1+4.3.1 [6] and D2.2.2 [1]. Here, we briefly summarize the most important concepts for reasons of readability.

A major task of the affordance-inspired control architecture that we proposed in D2.2.2 [1] is the generation of symbolic representations of such affordance relations, i.e. symbolic descriptions of the relation between indicative features in the environment – called *cues* –, a robot's acting upon parts of the environment – called behaviors – and the results of these actions, called – outcome. The basic representation form is that of a triple ([affordance] cue, behavior, outcome).

The cue descriptor contains sensory data – filtered or raw – that support the existence of an affordance. The outcome descriptor contains sensory data – filtered or raw – that describe the outcome as *perceived by the robot* of a prior application of a single or complex behavior. On repeated execution of a behavior, the outcome descriptor can be used to *verify* (→affordance hypothesis verification) the "success" of the execution, that is, whether the actual result matches the expected result. Both cue and outcome descriptors do not necessarily describe a single state, they can also describe time intervals. In terms of architectural control and data flow, affordance representations are produced in the following "bottom-up" way[2]:

The sensors enable the robot to perceive its environment and its internal states, virtual sensors provide software state information, real sensors yield extero- and proprioceptive data. All sensory data are first handled by the *Perception module*. It relays sensory data, extracted features and status information (like active behaviors and their parameters) to the Learning module, Execution module, Behavior System and Deliberation module. It can be configured to look just for certain features that relate to searched affordance support cues. Its *Entity Structure Generation Module* converts sensory data into appropriate data structures for architectural affordance support. The Learning module takes input from the Perception module and generates affordance representations (affordance triples) to populate a data pool called the *Affordance Representation Repository*.

Top-down perception is employed when the robot acts in a goal-oriented way. In this mode, the robot may look for the presence of certain affordances that are required to solve a task or sub-task. Such an affordance is represented as ([affordance] cue, behavior, outcome). The cue descriptor can be used to configure the perception module in such a way that the features that constitute the cue are preferred. The appropriate configuration

[2]Please note that this is only a brief summary of the complete architecture as proposed in D2.2.2 [1].

of the Perception module is initiated by the Event and Execution monitor [1]. Top-down perception and the interplay between Perception module and the Event and Execution monitor will be described in another deliverable. Here, we will describe bottom-up perception.

In the following section we will describe sensing concepts and the sensor configuration of the robot at hand, KURT3D.

# 4 Sensing

## 4.1 Multi-sensor data fusion

Multi-sensor data fusion in the context of the affordance based architecture is the unification and pre-processing of sensor data into a homogeneous overall view of the current situation usable for affordances or cognitive modules. Errors, noise and partial contradictory data are major problems while fusing data from different or bimodal sources. The resulting "situation picture" is the base for the perception, behaviors and learning modules. Further problems while fusing the sensor data have to be considered:

**Information interpretation:** The general idea behind the fusion of different data is the principle: "the whole is more than the sum of the individual parts". A consistent fused situation picture requires the partial interpretation of the different sensor data. On one hand the quality of the data has to be considered, e. g. Laser data are in general more reliable and precise than data from ultra-sonic sensors with the exception of windows and mirrors where Laser data is not suitable. On the other hand it has to be considered how a shift of the information weighting changes the situation picture and which potential danger could be caused when fusing incorrect data. As a principle, contradictory data should be double checked by two other different sensor sources.

**The information aging:** The frequency of the detailed data is differing e.g. 2D Laser data come at a frequency of 75 Hz, camera data at 15/30 Hz and odometry at 100 Hz. The multi-sensor data fusion system must be able to process information of different age. The age of the data plays a role regarding to the questions whether and how relevant it is for the current situation. Furthermore, it shows the need to process different data with different time stamps in order to decide whether the current observations are contradictory to older observations or whether a development is recognizable. For example, a Laser scanner in a pitching mode acquires 2D Laser scans at different horizontal positions. The scans have to be transformed to a more or less consistent 3D situation according to the robot pose measured by the odometry. It has to be decided if data discovered at different positions in different Laser scans belong to the same object.

**Information weighting:** The weighting of the sensor data differs and depends upon the possible interpretation of the sensor and its local position on the robot. On one hand, Laser scan data in front of the robot or in the current driving direction is more important for the navigation. On the other hand, Laser data left and right of the robot e.g. walls are more important for the localization and mapping. Thus, different weighting factors like system equipment, measuring range, scan frequencies and the current positions of a distributed sensor system of a mobile robot have to be considered for the information weighting. On a higher level, this is also done while integrating new data with former already fused data.

**Not commensurable data sources:** A comprehensive situation overview often requires the integration of sensors and data sources, which are different not only regarding their data structure, but also regarding their content. A set of pre-processing steps is necessary to shift the data to a similar semantic level on which the information is actually combinable. For example, encoder data has to be pre-processed to robot tracks to be able to compare these tracks with predefined target tracks as well as sensor or object interpretation and identification data. Furthermore, Laser and gyro data have to be integrated for reliably generating such a track. Last but not least all data processing and pre-processing has to be done in real-time i.e. with 100 Hz (encoder, gyro), 75 Hz (Laser scanner) and 15 Hz (camera data).

## 4.2 Robot sensors

To fulfill the above criteria, the existing software base has to be reorganized and further modules have to be implemented. In the last working period the following device drivers are implemented and the further developed code is reorganized according to the Coding Style, Programming Languages and Middleware specified in D1.1.3 (Implementation of Software Development Environment). The new base implementation foresees a sensor device driver to grab the data with an interface adapter that can be the device itself, a previously recorded file or the simulator. The file adapter is used to test the functionality of the device under development. All sensors can be configured by configuration files and easily be created due to the applied factory design pattern. Grabbed sensor data is marked with a time stamp and stored at a status board where all data of the current situation is collected. The history of the data and further post-process data has to be integrated with the DyKnow module (Deliverable Development of an Affordance-based Control Architecture D2.2.2). The sensor fusion itself is done with linear and nonlinear functions i.e. (extended) Kalman filter, closed loop controllers and logic functions [11; 12] . The list of device drivers consists of:

**Encoder device:** Encoder values from encoders of the two motors are grabbed at 100Hz from the CAN bus. For the odometry determination several test programs are developed e.g. countticks for the experimental determination of the number of ticks per turn of a wheel, speedtable for the experimental determination of the mapping between PWM values and motor speed (in m/s), and `friction_determination` for the determination of the dynamic friction based on the current weight and ground floor. The speedtable and `friction_determination` programs also use further sensor / actors like the motors and the gyro from the IMU.

**Motor device and controller:** The motor device can be seen both as an actuator as and as a virtual sensor. The controller computes the actual motor values based on the current speed values and the set speed values. It includes a feed-forward PI controller on the base of the determined mapping between speed and PWM values as a feed-forward term (see above). The integration values of the I part of the controller act as an interesting virtual sensor signal because it indicates large differences between set and current values which for example occurs when the robot is physically blocked. So it can be used as one indicator for the detection of the 'pushability' affordance.

**Laser scanner:** The basic Laser scanner device grabs 2D Laser scans with or without remission values.

**Servo device:** The servo device sets different servos for the pan and tilt cameras and the Laser scanner. Since the servo PWM signal has to be very precise, the connected S10 module ignores some signals that have to be retransmitted. The module returns the current position of the servos which is important for the fusion with the 2D Laser data, i.e. the calculation of the 3D data while step-rotating the 2D Laser scanner. For the initialization and calibration a test program determines a servo to angle mapping table based on the servo device and a gyro on top of the scanner. The table is used for the fusion of 2D Laser data and the servo angle to 3D data. Alternatively, a minimum and maximum angle could be measured with a fixed increment while moving.

**Virtual 3D sensor:** Based on the servo position and the 2D Laser scan the 3D Laser data is collected. The 3D data is consistent if the robot stops. While moving each 2D scan is also transformed with the relative pose change i.e. the data has to be fused with the pose from the encoder (odometry) data or pose information based on fusion of encoder and gyro data. Also noise reduction filters, e.g. 'salt and pepper' filter, and a filter for the calculation of floor points are under development. Furthermore, the unification of 3D data to a virtual 2D scan obstacle and a virtual 2D scan environment is under development. These filters are a projection from 3D to 2D data (x, y), whereas the z coordinate is considered in a specific interval, e.g., the height of the robot. The virtual 2.5D data is important for efficient obstacle avoidance / navigation and localization e.g. in a generated or given map. The mapping modules will be described in later deliverables.

**Camera:** A basic device driver for grabbing, storing and (jpg) compression of camera data has been developed.

**Virtual sensor; point of attention:** Positions of some foci of attention are calculated on the base of the grabbed camera data. The fusion of the attended points with the 3D data will be implemented in later deliverables.

**IMU device:** The data of the gyro and acceleration sensors of the IMU device can be grabbed at 100 Hz. A first ve rsion of the fusion of the gyro data with the encoder odometry is under development and being tested respectively according to the examinations of Borenstein et. al. [12].

**Virtual sensor; 2D free space orientation:** An orientation into the free space is calculated on the base of a horizontal 2D scan or a virtual 2D scan. The orientation is important for a "wander around" behavior and is the main part of the reactive drive behavior. This orientation can also be computed based on the fused data of the virtual 2.5D map.

### 4.3 Sensor fusion within the behaviors

The fused sensor data is on one hand used in the behavior system and on the other hand the behavior is a fusion process itself. In the following we describe some of the fusion process within the behavior system. A detailed description of the behaviors can be found in technical report "Specification of a Prototype Behavior System" [10].

**Brake behavior:** The brake behavior fuses the distance to an obstacle on the current way of the robot (measured from the Laser scan or virtual 2D scan) and the current speed of the robot. A higher robot speed leads to higher and earlier brake activation if an obstacle occurs.

**Turn behavior:** The turn behavior fuse the distance to an obstacle, the angular velocity and the orientation to a free space i.e. the robot turns in front of an obstacle until the road is free.

item[Motor behavior:] The motor behavior is basically a controller or filter that should map the set values $v_{set}$ and $\omega_{set}$ (for speed and direction) to the speed of the left and right wheel. The set values itself are from one up to three different sources e.g. odometery, gyro-odometery, laser-odometery or a combination [13].

**Drive behavior:** The drive behavior drives the robot into the direction that is the currently best free way $\alpha_{free}$. It fuses the speed and distance data.

**Go_To_Pose behavior:** The Go_To_Pose behavior approaches a given location that is nearby the robot e.g. (less than 2 m). It does not require higher level path planning. The behavior will eventually stop at that pose having the previously specified heading. It combines pose information to robots linear speed and angular velocity [14].

**Path following behavior:** Similar to the Go_To_Pose behavior the path following behavior combines (fused) pose information to robots linear speed and angular velocity. The reference path could be in local or global coordinates not limited to a nearby region [15].

Further behaviors like `Wander Around` are high-level behaviors which combine some low level behaviors like Drive, Brake, and Turn to move in the environment.

## 5 Feature Extraction for bottom-up Perception

### 5.1 Introduction

In this section, we describe some of our experiments that best illustrate feature extraction for (affordance) cue perception. The chosen example is that of the affordance 'liftability', i.e. perceiving that an object can be lifted by the robot's single dedicated manipulator, the crane arm. Figure 1 shows a scene in a visualization of our demonstrator setup [9], and as a photo of the real experimentation setup.

(a)

(b)

Figure 1: (a) Test objects and robot KURT3D in visualization. (b) Similar photo in real experimentation setup.

### 5.2 An Example: Recognizing Cues for Liftability

We performed specific experiments with the purpose to *investigate the potential of 3D information for providing affordance cues* in robot scenarios. In deliverable D3.1.2 [4] we presented a methodology to learn cues for the perception of affordances, in particular, for the affordance ‘liftability’. In this sample scenario, we used cues from 2D information to prove that – in contrast to the assumptions made by Gibson who completely focused on visual motion and information from 3D cues – affordances can be predicted from any available visual information, only based on the constraint that the information would be relevant to predict the potential of interaction with particular aspects of the environment. Here, we perform experiments with the goal to compare the efficiency of using cues from 2D information [4] with that of using 3D information. Another aspect of investigation is how far affordance perception could take advantage of *fusing information* from *various modalities*. To verify experiments of the simulation environment in real world we build a test scenario with different kind of objects. These objects can or can not be manipulated with the robot crane. Some of the cylindrical objects are magnetic some not or have spherical or flat covers. Figure 2 show the experimental scene with four different objects. The objects are produced according the guidelines in color, shape and size of the simulation environment. The left image shows the depth image, e.g. the z-coordinate of the 3D point cloud is coded as a gray color (black near, white far). The picture is distorted since the laser scanner has an opening angle of 180 degree. The middle image is also distorted and shows the among of reflected light of the laser beam. The right image is a picture of the left robot pan and tilt camera in vga (640x480 resolution). The vga picture shows only the objects in the scene since its opening angle is 40 degree which is a small part of the 180 degree opening angle of the laser scanner. Further pictures can be found in the appendix A.

The following paragraphs describe the experimental setup for the investigation of predictive cues from 3D information, the used methodology, and the proof of concept acquired from the investigation, and an outlook on further experiments of ongoing work:

**Experimental Setup:** According to the experimental setup used for learning the vision based prediction of the affordance ‘liftability’ in deliverable D3.1.2 [4], we position

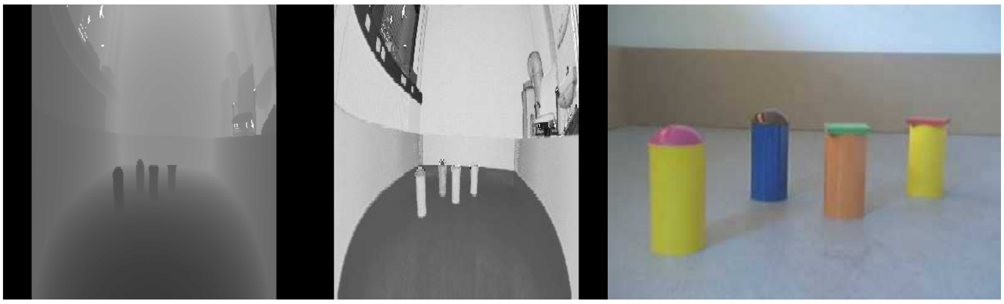
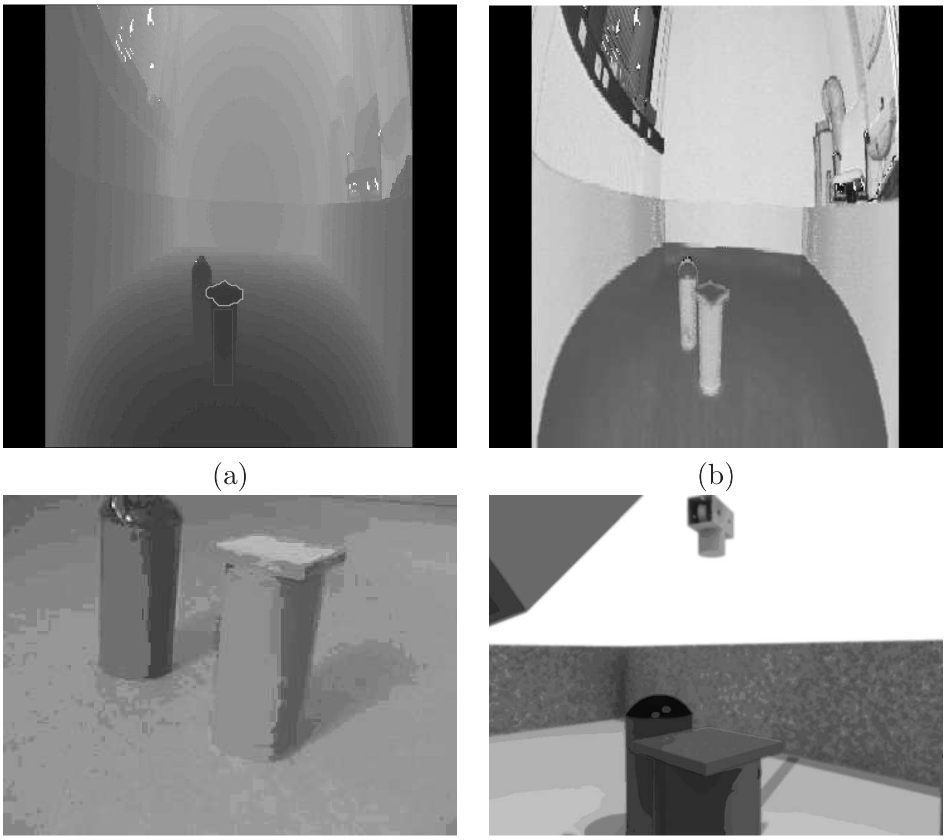


Figure 2: Left: Depth image of a experimental scene with four different objects. Middle: Remission image of the scene. Right: Camera image of the left robot camera in vga resolution.

the robot in front of characteristic test objects. Within a distance of about 40 cm from the test object (within reachability of the gripper), a wide area based laser scan and an image capture were initiated with the purpose to extract 3D and 2D information, respectively. Test objects were cans constructed by a cylindrical base volume and a top part that represents either spherical or planar structure. The crucial issue in predicting the affordance 'liftability' is to recognize the structure of the top part of the test object: planar structure physically means that the test object could become magnetized from a magnetic gripper lowered to the top of it, and spherical structure means that the test object could not get magnetized. From the capability to extract the structure of the top part, and from the potential to discriminate top parts from bottom, i.e., cylindrically structured parts, we would derive the power to form predictive cues for the perception of affordances. Figures 3, 4 and 5 depict (a) the depth coded image from the laser scan of the test object, (b) intensity coded laser scan images, (c) the camera view of the test object and (d) the corresponding image captured from the simulated scenario.

**Methodology:** In order to extract and classify 3D information, we need to identify segments of the depth coded images that would correspond to bottom or top parts of the test objects. Here we start with a color segmentation of the camera based image resulting in image regions that correspond to the object parts that were color coded. In Figures 3, 4 and 5 (a) we are able to identify the boundaries of the regions in the depth coded image that correspond to extracted color segmented regions in the camera based imagery. In the next step we extract the depth information within the segmented regions, compute the local depth gradients, and receive histograms of the distribution of the orientations within the complete respective regions. We selected various histogram bin sizes and concluded that the orientation interval of $[0, 36]^\circ$ would result in informative histograms. We normalized these histograms into probability distributions on the selected orientation intervals. Removing noise from these distributions by suppressing intervals with confidences lower than 10% we finally obtained discriminative distributions (Fig. 6). The entropy in these distributions gave us the means to classify into planar and spherical (circular in 2D terms) structure: distributions with entropies below 0.9 were classified as *planar* and spherical otherwise. This classification proved to be accurate in all test cases.



(a) (b) (c) (d)

Figure 3: Frame 216 with (a) depth coded, (b) intensity coded, (c) real world and (d) simulation based image.

**Proof of Concept:** The preliminary experiments described above demonstrate that there is a fundamental potential in 3D information (from depth coded laser scan images) for the prediction of the affordance 'liftability'. While we assume that the action and the effect associated to the affordance can be detected by arbitrary sensing and interpretation, we were mainly interested in whether 3D information would provide efficient features for affordance cueing. The classification of orientation histograms is highly discriminative in the prediction of the affordance relation 'liftability' since it enables to categorize segmented structures into planar and spherical characteristics. According to the conclusions made in deliverable D3.1.2 [4] the recognition of these structures is crucial for the prediction. Hence we regard the results of these preliminary experiments as proof of concept for the efficiency of 3D information to provide informative cues for affordance relations.

**Outlook:** More extensive experiments on the affordance relation 'liftability' will show whether 3D information are even more informative than 2D based cues, or whether any kind of information fusion from 2D and 3D features could result in more robust results. In order to investigate these issues, we plan to feed both 2D and 3D based features into an integrated feature vector and would proceed with feature selection

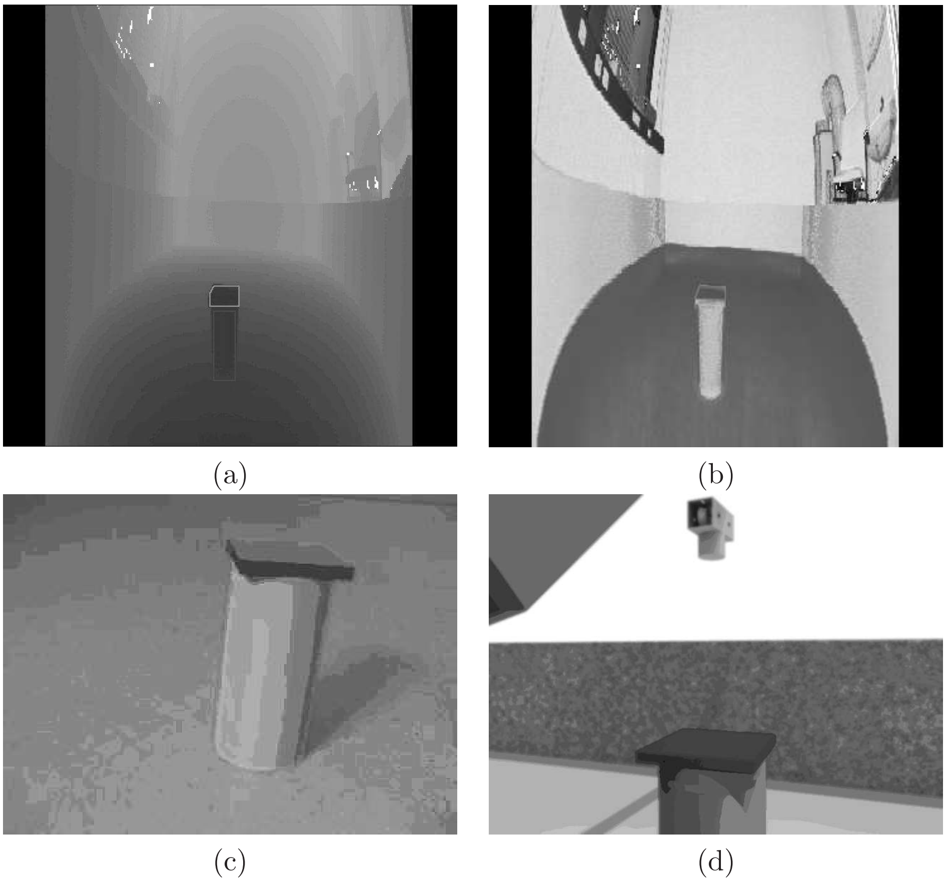
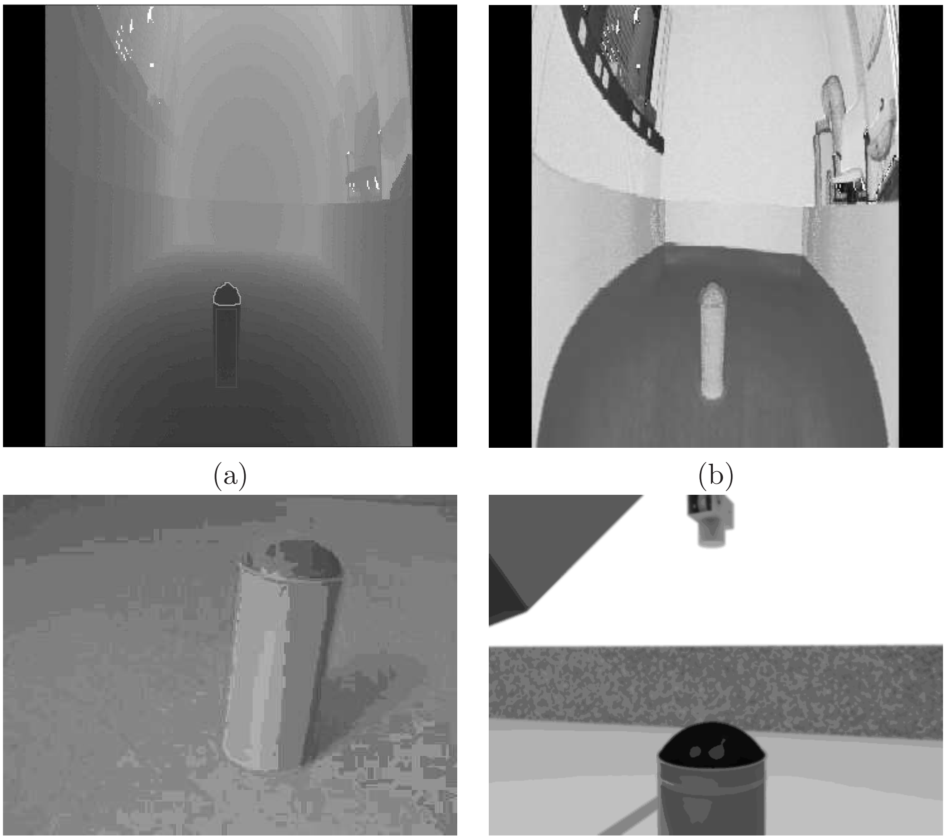


Figure 4: Frame 354 with (a) depth coded, (b) intensity coded, (c) real world and (d) simulation based image.

for the extraction of those features that would be most informative for the prediction of affordance relations, as outlined in previous work [16].



(a) (b)

(c) (d)

Figure 5: Frame 564 with (a) depth coded, (b) intensity coded, (c) real world and (d) simulation based image.

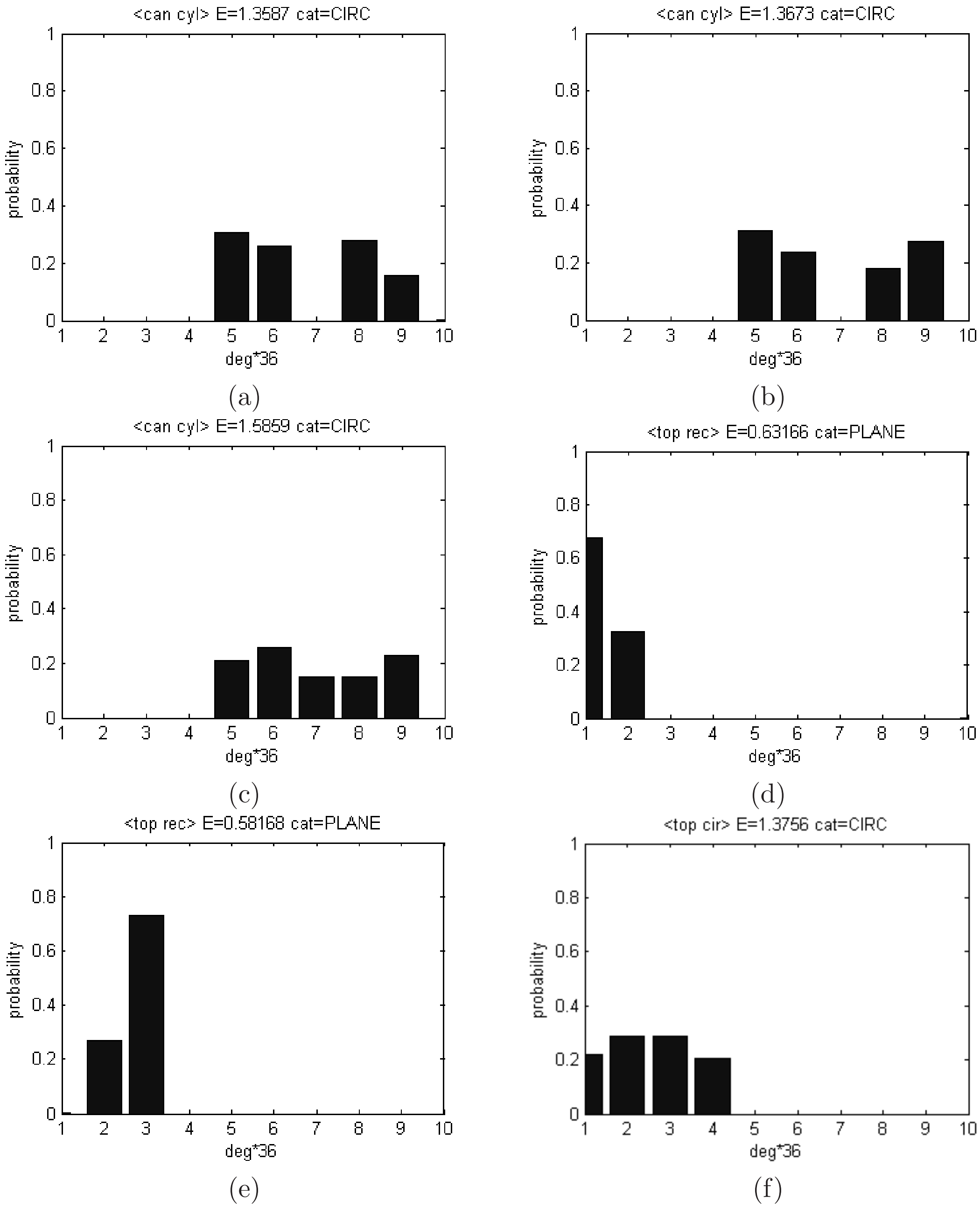



Figure 6: Probability distributions on orientations and classifications into planar and circular surface structures from 3D information for (a) (b) (c) (d) (e) (f).

# 6 State of Work and Outlook

## 6.1 State of work

As described above two main modules use multi data sensor fusion. We have currently implemented basic c++ classes for all of the different sensors on the robot namely:

- The encoder device
- The motor device and controller
- The laser scanner in 2D and 3D mode
- The servo device
- The camera device
- The virtual 3D sensor including free space orientation

Based on the current implementation we have done experiments to test the behavior system as well as the perception modules.

## 6.2 Outlook

To consider the information aging of the different sensors the next steps will be the integration of a 3 level data logging concept. On the top level all sensor or virtual sensor data is send to the DyKnow implementation. The DyKnow software runs on a server connected via WLAN and CORBA with the robot. The update rate is around 2 hz. Faster sensors send average values. Typical examples are pose information, robot speed, activation and state values of actions and behaviors but also planning date and pre and post conditions. The stored data can be used by the learning and planning modules. On the middle level every sensor will use a round robin queue to store data locally for around 2 to 5 seconds. Behaviors and actions will directly use the data to control the robot. On the bottom level simple data processing e.g. integration and derivation of the sensor data is used to fed closed loop controllers e.g. the feedforward PI controller of the left and right wheel chains.

Furthermore we will continue the implementation of:

- Entity Structure Generation Module (ESGM) using the DyKnow software)
- Perception module, including Feature extraction
- The behavior system

We will add more detailed specifications of

- 3D information based feature extraction from laser based depth coded imagery, such as, evaluating SIFT descriptors for structure characterization
- Information fusion components for the feature and the decision level of fusion, such as, Bayesian decision fusion methods (naive Bayes and conditionally dependent probabilistic fusion schemes)
- Stereo based 3D information recovery for the support of rapid and more extensive estimation of depth information

# 7 Conclusion

In this deliverable, we have focused on the special properties and possibilities of the variety of sensors available on the robot, namely encoders, servos, odometry, gyros, cameras, and a multi-modal 3D Laser Scanner. The main uses of the fused data are located in the behavior system module and the perception module. In the behavior system, multi-sensory data are used by control routines that implement robot behaviors and actions. These behaviors and actions constitute basic "skills" that control the robot while acquiring knowledge about affordance relations, as well as more complex behaviors that can be tuned by other modules in the control architecture. In the perception module, multi-sensory data are used by feature extractors to provide the features for the learning of affordance relations and features for active perception and search of cues for the presence of affordances. Both modules, i.e. the behavior system's control routines and the feature extractors, also apply sensor fusion methods to produce more reliable data and features. Furthermore we described the state of current the implementation and the experiments already evaluated with the robot sensors.

# References

[1] Erich Rome, Erol Şahin, Ralph Breithaupt, Jörg Irran, Florian Kintzler, Lucas Paletta, Maya Çakmak, Emre Uğur, Göktürk Üçoluk, Mehmet R. Doğar, Piotr Rudol, Gerald Fritz, Georg Dorffner, Patrick Doherty, Mariusz Wzoreck, Hartmut Surmann, and Christopher Lörken. Evaluation of existing control architectures for using affordances. Deliverable MACS/2/2.2 v1, Fraunhofer Institut für Intelligente Analyse- und Informationssysteme, Sankt Augustin, Germany, 2006.

[2] Rainer Worst, Claus Hoffmann, Björn Wingman, Maya Çakmak, and Martin Hülse. Specification of module interfaces. Deliverable MACS/1/1.2 v2, Fraunhofer Institut für Autonome Intelligente Systeme, Sankt Augustin, Germany, 2005.

[3] Erol Şahin, Ralph Breithaupt, Erich Rome, and Patrick Doherty. Identification of architectural requirements of an affordance-based control. Deliverable MACS/2/1.1 v3, Middle East Technical University Dept. of Computer Engineering, Ankara, Turkey, 2005.

[4] Lucas Paletta, Gerald Fritz, Erol Şahin, and Manish Kumar. Affordance recognition from visual cues. Deliverable MACS/3/1.2 v1, Joanneum Research Institute of Digital Image Processing Computational Perception (CAPE), Graz, Austria, 2005.

[5] Simone Frintrop, Martin Hülse, Erich Rome, and Lucas Paletta. Saliency detection with visual attention. Deliverable MACS/3/1.3 v1, Fraunhofer Institut für Autonome Intelligente Systeme, Sankt Augustin, Germany, 2005.

[6] Patrick Doherty, Torsten Merz, Piotr Rudol, and Mariusz Wzorek. Tentative proposal for a formal theory of affordances
tentative proposal for an affordance support architecture
prototype: Affordance-based motion planner. Technical Report MACS/4/2.1 v1, Linköpings Universitet, IDA Group, Linköping, Sweden, 2005.

[7] Georg Dorffner, Jörg Irran, Florian Kintzler, and Patrick Pölz. Robotic learning architecture that can be taught by manually putting the robot through action sequences. Deliverable MACS/5/3.1 v1, österreichische Studiengesellschaft für Kybernetik (öSGK), Vienna, Austria, 2005.

[8] Ralph Breithaupt, Simone Frintrop, Joachim Hertzberg, Erich Rome, and Bernd S. Müller. Specification of final demonstrator. Deliverable MACS/6/1.1 v2, Fraunhofer Institut für Autonome Intelligente Systeme, Sankt Augustin, Germany, 2004.

[9] Ralph Breithaupt, Simone Frintrop, Erol Şahin, Joachim Hertzberg, Patrick Pölz, Piotr Rudol, Emre Uğur, Patrick Doherty, Erich Rome, and Bernd S. Müller. Report on experiment design. Deliverable MACS/6/4.1 v1, Fraunhofer Institut für Autonome Intelligente Systeme, Sankt Augustin, Germany, 2004.

[10] Christopher Lörken and Hartmut Surmann. Specification of a prototype behavior system. Technical Report Draft Version 0.5, Fraunhofer AIS, March 2006.

[11] J. Borenstein and L. Feng. Gyrodometry: A New Method for Combining Data from Gyros and Odometry in Mobile Robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '96)*, pages 423–428, April 1996.

[12] J. Borenstein and L. Feng. Measurement and Correction of Systematic Odometry Errors in Mobile Robots. *IEEE Transactions on Robotics and Automation*, 12(6):869–880, December 1996.

[13] Kai Lingemann, Andreas Nüchter, Joachim Hertzberg, and Hartmut Surmann. About the Control of High Speed Mobile Indoor Robots. In *Proc. 2. European Conference on Mobile Robotics ECMR '05*, pages 218–223, Ancona, Italy, September 2005.

[14] Giovanni Indiveri. Kinematic Time-invariant Control of a 2D Nonholonomic Vehicle. In *Proceedings of the 38th Conference on Decision and Control, (CDC '99)*, Phoenix, USA, December 1999.

[15] Giovanni Indiveri and Maria Letizia Corradini. Switching linear path following for bounded curvature car-like vehicles. In *Proc. of the 5th IFAC Symposium on Intelligent Autonomous Vehicles, IFAC-IAV04*, Lisbon, Portugal, July 2004.

[16] Gerald Fritz, Lucas Paletta, Manish Kumar, Georg Dorffner, Ralph Breithaupt, and Erich Rome. Visual Learning of Affordance based Cues. In *Proc. International Conference on the Simulation of Adaptive Behavior, SAB*, pages 52–64, Rome, Italy, LNAI 4095, Berlin, Germany, September 2006. Springer-Verlag.

# A Real world pictures

Enclosed we show further pictures from the real world robot scenario which will be used in further experiments.

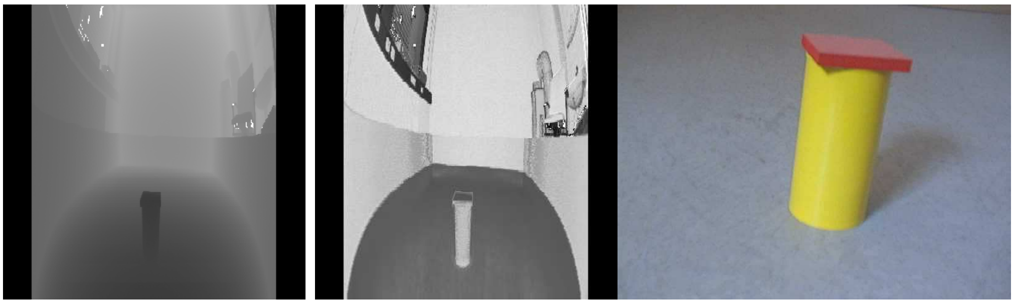


Figure 7: Left: Depth image of a experimental scene with one liftable objects. Middle: Remission image of the scene. Right: Camera image of the left robot camera in vga resolution.

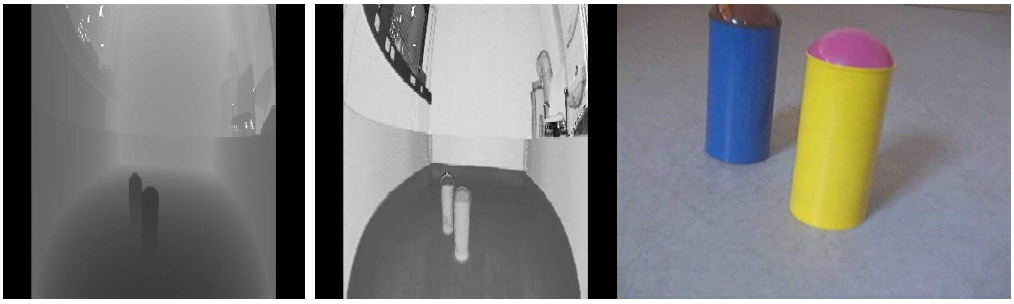


Figure 8: Left: Depth image of a experimental scene with only non liftable objects. Middle: Remission image of the scene. Right: Camera image of the left robot camera in vga resolution.