Winning the RoboCup@Work 2014 Competition: The smARTLab Approach

Bastian Broecker, Daniel Claes(✉), Joscha Fossel, and Karl Tuyls

smARTLab, Department of Computer Science,
University of Liverpool, Liverpool, UK
{bastian.broecker,daniel.claes,j.fossel,k.tuyls}@liverpool.ac.uk

Abstract. In this paper we summarise the approach of the smARTLab@Work team that has won the 2014 RoboCup@Work competition. smARTLab (swarms, multi-agent, and robot technologies and learning lab) is a robotics research lab that is part of the Agent ART group at the computer science department of the University of Liverpool. This team has won the competition for the second year in a row. The team previously competed as swarmlab@Work and changed name after the move of professor Tuyls and his research group from Maastricht University to the University of Liverpool, UK. The various techniques that have been combined to win the competition come from different computer science domains, including machine learning, (simultaneous) localisation and mapping, navigation, object recognition and object manipulation. While the RoboCup@Work league is not a standard platform league, all participants use a (customised) KUKA youBot. The stock youBot is a ground based platform, capable of omnidirectional movement and equipped with a five degree of freedom arm featuring a parallel gripper. We present our adaptations to the robot, in which the replacement of the gripper was the most important upgrade comparing to the version of the robot that was used last year.

1 Introduction

Although fixed industrial robotics has achieved unrivalled productivity in mass production lines executing predefined repetitive tasks, such automation is still unreachable in customised flexible production or assembly situations. Such situations typically involve a high mixture of different types of products or objects that need to be transferred from one configuration into another. For example this could involve a task in a car garage in which robots need to collect specific parts from a bin or cupboard from one side of the workshop and bring these to another part or even put them back in a specific configuration, requiring planning, learning and reasoning capabilities as well as multi-robot and human-robot cooperation.

Recent years have seen new initiatives in this direction by the initiation of new robot competitions that focus on future industrial robot settings [11]. The most famous competition of its type is RoboCup@Work [10], which is specifically targeted to mobile industrial robotics in futuristic settings (the factory of

R.A.C. Bianchi et al. (Eds.): RoboCup 2014, LNAI 8992, pp. 142–154, 2015.
DOI: 10.1007/978-3-319-18615-3_12

the future). The competition, in particular, focuses on the use of robots in various work-related or industrial scenarios, such as navigating to specific locations and moving and manipulating objects. It aims to foster the research and development of innovative industrial robots for current and future industrial applications, where robots cooperate with humans on complex tasks ranging from manufacturing and automation to general logistics. Currently this competition is limited to one single mobile industrial robot (Kuka youBot) that has to achieve a number of predefined challenges. The current single-robot setting still poses important scientific and engineering challenges i.e. navigation, object detection, planning, learning and manipulation from the robotic platform in several industrial settings.

The competition has been founded in 2012 and has matured in the past two years. Some teams (including ours) have made their source code from the last competition publicly available[1].

2 smARTLab@Work

smARTLab is the research laboratory of the Department of Computer Science at University of Liverpool that focuses on designing Autonomous Systems. The general research goal and mission of the lab is to create adaptive systems and robots that are able to autonomously operate in complex and diverse environments. The smARTLab@Work team has been established in the beginning of 2014, consisting of 3 PhD candidates and 2 senior faculty members. The team has won the @Work competitions of the 2013 RoboCup German Open, and the 2013 RoboCup world championship under the name swarmlab@Work. After migrating to the University of Liverpool smARTLab@Work also placed first in the 2014 German Open and the 2014 RoboCup world championship.

The team's mission is (a) to apply smARTLab research achievements in the area of industrial robotics and (b) identify new research challenges that connect to the core smARTLab research areas: autonomous systems, reinforcement learning and multi-robot coordination.

In the remainder of this section we introduce the smARTLab@Work robot platform that was used for the 2014 RoboCup@Work world championship. Especially, we describe the necessary hardware modifications that were made to adapt this standard platform to the winning configuration.

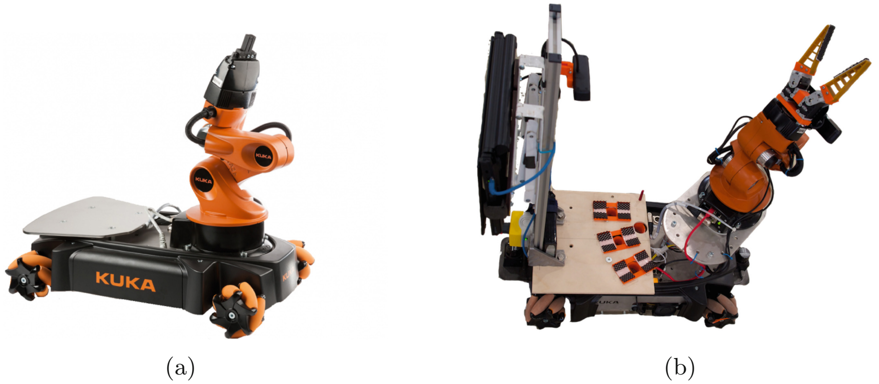
2.1 YouBot Platform

The youBot is an omni-directional platform that has four mecanum [8] wheels, a 5 DoF manipulator and a two finger gripper. The platform is manufactured by KUKA[2], and is commercially available at the youBot-store[3]. It has been

[1] https://github.com/swarmlab/swarmlabatwork.

[2] http://kuka.com.

[3] http://youbot-store.com/.



(a) (b)

Fig. 1. (a) Stock youBot. (b) smARTLab@Work modified youBot for world championships 2014.

designed to work in industrial like environments and to perform various industrial tasks. With this open-source robot, KUKA is targeting educational and research markets [2]. Figure 1a shows a model of the stock youBot.

The youBot comes with a 5-degree-of-freedom arm that is made from casted magnesium, and has a 2-degree-of-freedom gripper. The standard gripper has two detachable fingers that can be remounted in different configurations. The gripper has a stroke of 20 mm and a reach of 50 mm, it opens and closes with an approximate speed of 1 cm/s. All hardware modules of the robot communicate internally over real-time EtherCat [9].

2.2 Hardware Modifications

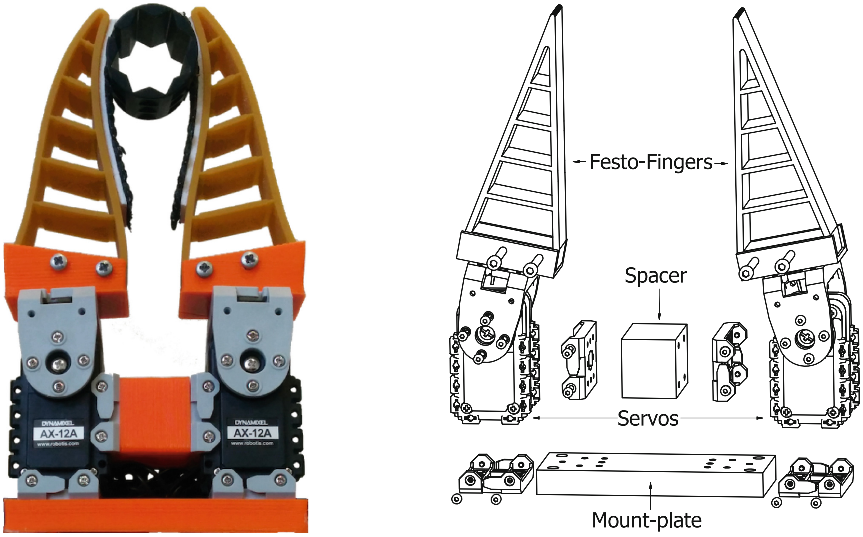
In order to meet the requirements we demand from the youBot platform to tackle the challenges, we made a number of modifications to the robot. In this paragraph we describe which parts are modified and why these modifications are a necessity for our approach. Figure 1b shows the modified smARTLab@Work youBot setup that was used in the world championships in Brazil 2014.

The main adaptations with respect to last years version of the youBot are the replacement of the standard gripper with a custom made gripper and the design of a backplate to allow transportation of multiple objects. Since the new custom gripper is longer than the standard gripper, the arm was elevated by an additional 10 cm to still be able pick objects in a top-down fashion.

The standard gripper is replaced by two FESTO FinGripper fingers[4] mounted on two Dynamixel AX-12A[5] servo motors. This increases the stroke to more than 20 cm and the speed of the gripper to up to 10 cm/s. Also the fingers passively adapt to the shape of the objects as illustrated in Fig. 2. Through the custom

[4] http://www.festo.com/rep/en_corp/assets/pdf/Tripod_en.pdf.

[5] http://support.robotis.com/en/product/dynamixel/ax_series/dxl_ax_actuator.htm.



(a)

(b)

Fig. 2. Figure (a) illustrates the adaptation of the gripper to the round shape of the object. Figure (b) shows a drawing of the separate parts of the new designed gripper in more detail.

gripper is the youBot is now able to grasp every competition object, where the previous standard gripper was limited to the thinner objects due to his maximal opening width of 2 cm.

The new backplate (see Fig. 1b) offers three high friction pockets to place the objects on the robot. These pockets are designed to hold the items in place during transportation.

In order to detect and recognise manipulation objects, an ASUS Xtion PRO LIVE RGBD camera is attached to the last arm joint. This camera is mounted, so that it faces away from the manipulator, as can be seen in Fig. 7a. The main idea behind this mounting position originates from the minimal distance of the RGB-D camera, which is approximately ~0.5 m. Using this special pre-grip pose, we can scan the area and perceive objects from a top-down perspective. An additional camera is mounted on the back of the robot. This camera is used to get a second view of certain objects, i.e. an object is picked up and it is moved by the arm such that it can be seen from a different angle. This helps to resolve confusions that are inevitable when only looking from a top-down fashion with the ASUS RGBD camera.

In this section we described the new hardware modifications needed to optimally execute the required RoboCup@Work tasks, in the next section these tasks will be describes in more detail, as well as the methods that we used to solve the specific tasks.

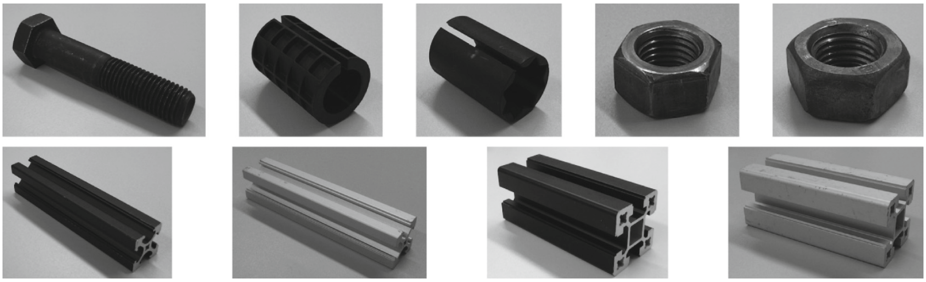
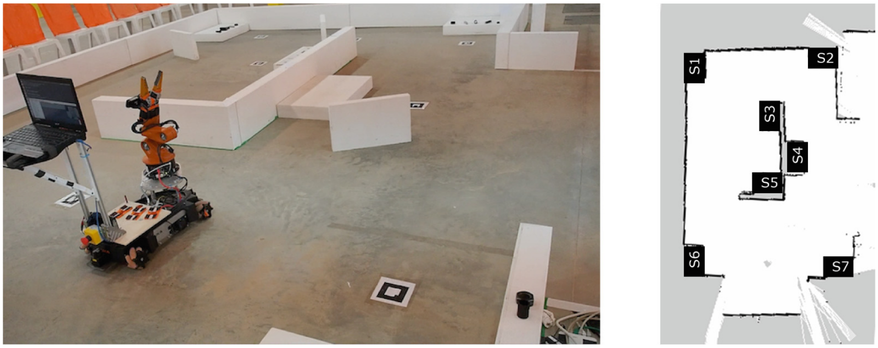


Fig. 3. Manipulation objects, from top left to bottom right: M20_100, R20, V20, M20, M30, F20_20_B, F20_20_G, S40_40_B, S40_40_G.



(a)

(b)

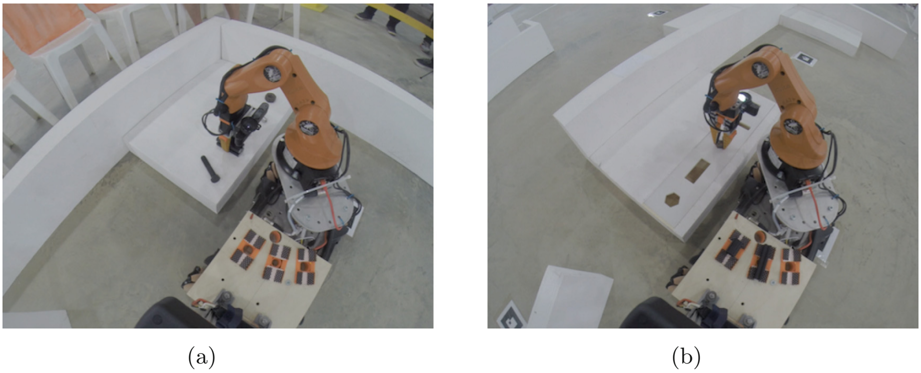
Fig. 4. (a) 2014 arena during final run with extra obstacles. (b) Map of the arena annotated with the service areas.

3 RoboCup@Work

In this section, we introduce the various tests of the 2014 RoboCup@Work world championship. Note that while the tests are slightly different for various levels of difficulty, we only give a description for the difficulty chosen by smARTLab@Work. For a more detailed description we refer the reader to the RoboCup@Work rule book[6]. A list of manipulation objects used in 2014 is given in Fig. 3. A picture of the 2014 arena is shown in Fig. 4a, in which several extra static obstacles are placed. In the arena, there are seven service areas as annotated in Fig. 4b. The term service area refers to a small table from which objects have to be manipulated. Figure 5a shows our robot performing a grasp in one of the service areas.

Basic Navigation Test. The purpose of the Basic Navigation Test (BNT) is testing mapping, localisation, path planning, obstacle avoidance and motion

[6] http://robocupatwork.org/resources.html.



(a) (b)

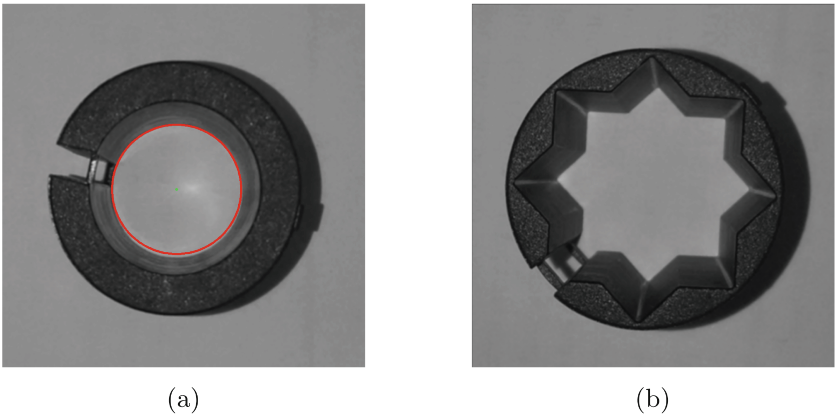
Fig. 5. (a) Grasping the S40_40_B in a service area. (b) A PPT placement into the M20 hole.

control. The robot has to autonomously enter the (known) arena, visit multiple way-points, and exit the arena again. Each way-point has to be accurately covered by the robot for a given duration in a given orientation. Additionally, obstacles may be spawned randomly in the arena, where a penalty is given if the robot collides with an obstacle.

Basic Manipulation Test. In the Basic Manipulation Test (BMT), the robot has to demonstrate basic manipulation capabilities such as grasping and placing an object. For this purpose three target and two decoy objects are placed on a service area, and the robot has to detect the three actual target objects and move them to another service area that is close by. All three objects are allowed to be carried at the same time. Implicitly this also tests whether the object recognition works sufficiently well. Since navigation is not a part of this test, the robot may be placed at any location before the test starts and is allowed to end anywhere in the arena.

Basic Transportation Test. The Basic Transportation Test (BTT) is a combination of the BNT and BMT. Here, six objects have to be picked up from various service areas and transported to specific destination areas. Similar to BNT and BMT both decoy objects and obstacles are placed on the service areas and in the arena, respectively. The robot has to start and end the BTT outside of the arena. In addition to the challenges individually posed by BNT and BMT, the robot now has to approach multiple service and destination areas within a certain time window, hence for optimal performance determining the optimal route and payload configuration is necessary. Only three objects are allowed to be carried at any given point in time.

Precision Placement Test. The Precision Placement Test (PPT) consists of grasping objects (see BMT), followed by placing them very accurately in object-specific cavities. Every object has one specific cavity for vertical, and one specific



(a)

(b)

Fig. 6. (a) R20 object with detected circle. (b) V20 object with no detected circle.

cavity for horizontal placement. Each cavity has the same outline as the object plus 10 percent tolerance. Figure 5b shows the PPT destination area with some sample cavities.

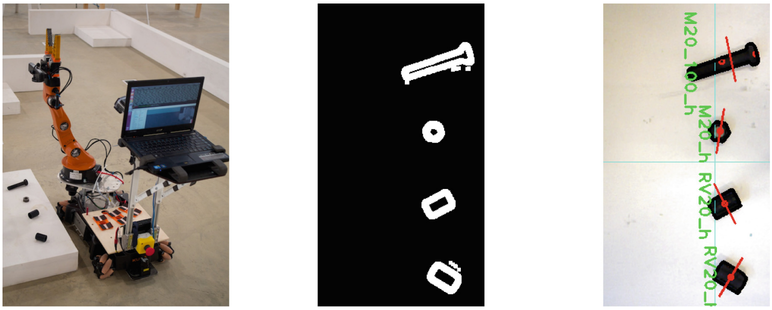
Final Test. The final tests consists of a combination of all the tests described above. More specifically, in the arena various unknown static obstacles are present. There are six objects distributed over three service platforms with other decoy objects present. Of these objects two have to be placed precisely into the corresponding holes as in the PPT and the rest has to be placed on different service platforms as in the BTT.

4 Approach

In this section, the different techniques used to tackle the above mentioned tests are explained. We developed different modules for many different capabilities, e.g. basic global navigation, fine-positioning, object recognition, inverse kinematics of the arm, etc. We also explain our various recovery methods. By combining these capabilities in state-machines we are able to solve the tasks specified above.

The main adaptation with respect to last year, is the addition of the possibility to transport objects on the backplate of the robot. This allows us to transport more than one object at a time and actually planning the task becomes an important factor in the execution of the test. Furthermore, the object recognition was improved by combining the depth information together with the RGB information for the features. Also, the additional camera on the back was used to distinguish the R20 and V20 object, since the object shapes are very similar as shown in Fig. 3.

Thus for the descriptions of the approaches for mapping, localisation, inverse-kinematics and fine-positioning, we refer to [1].



(a) (b) (c)

Fig. 7. (a) Pre-grip scan position. (b) Pre-processing of the image. (c) Detected objects, classification and grasp position of the closest object.

Navigation. A necessary capability of the robot is to navigate in the known environment without colliding with obstacles. The map is created with gmapping [6] and AMCL [4] is used for the localisation in the map. The global path is computed by an A* algorithm and is then executed using a dynamic window approach [5] trajectory rollout for the local path planning. This planner samples different velocities and ranks them according to the distance to the goal and the distance to the path, while velocities that collide with the environment are excluded. We tuned the trajectory rollout in such a way that it incorporates the omni-directional capabilities of the robot and also included scores for the correct destination heading, when being in close distance to the goal. The ROS implementation of the planner only takes the goal position into account and disregards the goal orientation until the robot has reached the goal position. The robot then performs a turn in place in order to achieve the correct orientation. Additionally, the robot tends to drive diagonally instead of straight, since the magnitude of the velocity can be higher and velocities with non-negative y values are sampled first. To tackle these problems, two additional scoring functions were added to the evaluation for each velocity sample:

$$c_p = |y| * w_p \tag{1}$$

$$c_h = |\theta_r - \theta_g| * w_h \tag{2}$$

where c_p is the penalty for considering velocities that contain a y component. c_h is the costs for the difference of the current heading θ_r in comparison to the goal heading θ_g. w_p and w_h are the corresponding weights for the two cost values. Equation 1 ensures that the robot prefers driving straight instead of diagonally. Equation 2 is applied if the robot is in proximity to the goal position, i.e. 1.5 m. Thus, the robot rotates towards the target heading while driving and it does not need to turn in place.

Object/Hole Recognition. Besides the navigation tasks, object detection and recognition is crucial to be able to interact with the environment, i.e. picking up objects and placing them in the correct target locations. We use the openCV-library[7] to detect the objects. We use two different detection pipelines, one is based on the depth image and the other is based on the RGB image. The depth image has the advantage that it is mostly independent of light conditions, while the RGB image is more accurate and allows to distinguish between black and silver objects.

An adaptive threshold filter is applied to the input image. Afterwards the image is converted into a black and white image and this is used to detect the contours of the objects as shown in Fig. 7b. We use various features of the detected objects, e.g., length of principal axis, average intensity and area, and use a labeled data-set that we created on the location to train a J4.8 decision tree in weka [7] for the recognition of the objects. This decision tree is then used for the online classification of the objects. Figure 7c shows the detection in a service area.

In order to distinguish between the R20 and the V20 object, we use an additional camera on the back. The objects have the same shape from the outside, but inside there different, i.e. round or star-shaped as shown in Fig. 6. This means that we have to pick the object up and hold in such that the additional camera can look inside the object. We use the Hough circle detection method [12] in order to determine if the inside of the object is round or star-shaped.

Recovery Methods. Of course, in robotics many things can go wrong, so we include various recovery methods. The navigation can easily get stuck, since the robot is tuned to drive very close to obstacles. Therefore, we implemented a force field [3] recovery. More specifically, the robot moves away from every obstacle that is picked up by the laser scan for a couple of seconds and the navigation is restarted. The servos in the arm tend to overshoot sometimes, especially when they get hot. Since the new fingers are compliant, it is not too problematic to slightly hit the ground, but in some cases it can lead to an overload of the arm. Thus, we measure the voltages in the arm, and as soon as a certain voltage is exceeded, we recover the arm to an upwards position and back up a couple of states in the state machine to be able to try again.

4.1 State Machine

For the different tests, we combine the various capabilities explained in the previous section in different state machines to complete the tasks. For the BNT, we do not need any of the manipulation techniques, so we have a very small state machine that combines the navigation and the fine-positioning. More specifically, we use the navigation to get close to the position and then the fine-positioning

[7] http://opencv.org.

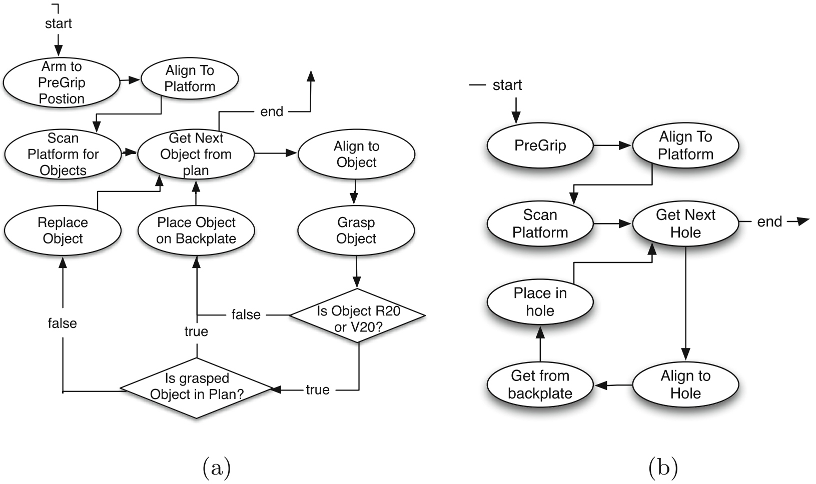


Fig. 8. (a) The simplified subtask-state-machine that is used for grasping objects at a service area. (b) The simplified subtask-state-machine that is used for dropping objects into the holes for PPT.

moves the robot exactly to the asked position. If the navigation fails, the force-field recovery is activated. Afterwards it retries to move to the current location. When it fails more than a fixed number of times, the current goal is skipped.

In principle the BMT, BTT, PPT and the final tasks do not differ very much. For instance, the BMT is basically a BTT, in which only one source and one target location are used and the PPT is a BMT in which the objects have to be placed in a specific location instead of anywhere on the platform. Thus for all these other tests we created so-called subtask state-machines that are shown in Fig. 8. For sake of simplicity, the recovery connections are left out. The subtasks are for example an behavior sequence for aligning to a platform a grasping all objects that need to be taken according to the task specification and placing them on the backplate of the robot as shown in Fig. 8a. Before any object is taken, a full scan of the service platform is performed. This enables us to reason about the objects, if there are any classifications that are uncertain. For instance, if there is only one R20 or V20 found on the platform, and the specification says that we have to get one R20 from that service area, we know that the object should be a R20 and the extra check does not need to be performed. Another subtask is placing all objects that need to be precisely placed at this location, shown in Fig. 8b. This task is very similar to the previously described behavior, since it is the reversed idea. The last subtask is the normal place of objects at target location. This is also similar to the PPT place, but since the objects can be placed anywhere on the platform, we only need to move once to the center of the platform and use the inverse kinematics of the arm to place the objects.

These subtask state machines are combined by a decision state that plans where to go and what to do next. This state gets all the necessary global and

Table 1. Scores for the different tests.

Place	Team	BNT	BMT#1	BMT#2	BTT#1	BTT#2	PPT	Final	Total
1	smARTLab@Work	420	0	250	400	187.5	500	1100	2857.5
2	b-it-bots	420	187.5	150	275	196.875	112.5	468.75	1810.625
3	Lyon CPE	187.5	93.75	112.5	275	84.375	225	0	978.125
4	LUHbots	210	50	100	75	300	150	0	885
5	WF Wolves	200	37.5	75	0	84.375	75	168.75	640.625
6	Robo-Erectus	0	0	0	0	0	0	0	0
7	Robotica UBM	0	0	0	0	0	0	0	0

local information, i.e. which items are currently on the backplate, which items are still located in the arena, where is the robot located, etc. By looking at this information the best next location is chosen, i.e. to pick up more items, since there is space left on the backplate, or to go to a target area to (precisely) place items there. This state is called each time the subtask-state-machines are finished. If there is no item left on the backplate and also nothing more specified in the task description, the robot knows it is finished and moves out of the arena.

5 Competition

Table 1 shows the final scores for the 2014 RoboCup championships.

The team scored full points for almost every test, except for BMT#1 and BTT#2. During the first run of the BMT tests, the robot tried to align to the target service area using the ASUS RGBD camera. Since the arm was in a fully tucked position before the tests, and the joint to which the camera is attached tends to overshoot heavily, no line was found in the image to which the robot could align to. The behaviour that was implemented as solution, was to “blindly” drive in the assumed direction of the platforms for a couple of centimeters. However, since the robot was already very close, it crashed into the platform and the run had to be aborted.

This issue relates to the problem that we had in the second BTT run. There we used part of the program to initialise the arm to a different position, but by doing that we disabled a part of the state machine that we forgot to enable again. Thus the state-machine stopped prematurely, and the run was stopped after grasping all necessary objects from one platform and placing them at another platform.

The rest of the tests went flawlessly, even in the final test, the combination of all of the above, we got the maximum points that were achievable in the difficulty level that was chosen. A video of the final run can be found here: http://youtu.be/FzVgDqrWiOY.

6 Future Work

The smARTLab@Work code and hardware designs will be released on Github[8]. For future competitions we plan on further improving the object recognition by

[8] http://github.com/smartlab-liv.

adding state of the art techniques, such as deep learning algorithms in order to be even more robust against changing light conditions and also to be able to quickly adapt to new objects. Furthermore, we are currently working on a learning by demonstration approach to allow for learning new arm trajectories on the fly. This might enable grasping various 'more difficult to grasp objects', e.g. glasses. It will also help to cope with future RoboCup@Work tests, which are yet to be introduced, e.g. assembly tasks or picking up moving objects.

Additionally, this year we were almost able to compete on the highest difficulty level for the tasks, which includes the detection and grasping of upright standing objects. However, as the object recognition currently did not perform sufficiently reliable for this case, we opted to not tackle grasping upright standing objects in the competition.

Hardware-wise, we plan on mounting the arm on a rotating extension plate, so that the arm can be turned towards the optimal grasp position. Additionally, the backplate can be optimized, maybe even by including servos to hold different kinds of objects in place during transportation.

Lastly, we are looking into an open demonstration that involves the cooperation of many robots, e.g. some Turtlebots[9] for transportation and the youBot to grasp and place the objects on the Turtlebots.

References

1. Alers, S., Claes, D., Fossel, J., Hennes, D., Tuyls, K., Weiss, G.: How to win robocup@work? In: Behnke, S., Veloso, M., Visser, A., Xiong, R. (eds.) RoboCup 2013. LNCS, vol. 8371, pp. 147–158. Springer, Heidelberg (2014)
2. Bischoff, R., Huggenberger, U., Prassler, E.: Kuka youbot - a mobile manipulator for research and education. In: 2011 IEEE International Conference on Robotics and Automation (ICRA), pp. 1–4, May 2011
3. Chong, N. Y., Kotoku, T., Ohba, K., Tanie, K.: Virtual repulsive force field guided coordination for multi-telerobot collaboration. In: Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA 2001) (2001)
4. Fox, D., Burgard, W., Frank D., and Thrun, S.: Monte carlo localization: efficient position estimation for mobile robots. In: Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI 1999) (1999)
5. Fox, D., Burgard, W., Thrun, S.: The dynamic window approach to collision avoidance. IEEE Robot. Autom. Mag. **4**, 23–33 (1997)
6. Grisetti, G., Stachniss, C., Burgard, W.: Improved techniques for grid mapping with rao-blackwellized particle filters. IEEE Trans. Robot. **23**, 34–46 (2007)
7. Hall, M., Eibe, F., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. SIGKDD Explor. Newsl. **11**(1), 10–18 (2009)
8. Hon, B.E.: Wheels for a course stable selfpropelling vehicle movable in any desired direction on the ground or some other base. US Patent 3,876,255 (1975)
9. Jansen, D., Buttner, H.: Real-time ethernet: the ethercat solution. Comput. Control Eng. **15**(1), 16–21 (2004)

[9] http://www.turtlebot.com.

10. Kraetzschmar, G.K., Hochgeschwender, N., Nowak, W., Hegger, F., Schneider, S., Dwiputra, J., Berghofer, J., Bischof, R.: Robocup@work: competing for the factory of the future. In: RoboCup Symposium (2014)
11. TAPAS Project. Robotics-enabled Logistics and Assistive Services for the Transformable Factory of the Future (TAPAS) (2013). http://tapas-project.eu/
12. Xu, L., Oja, E., Kultanen, P.: A new curve detection method: Randomized Hough transform (RHT). Pattern Recogn. Lett. **11**(5), 331–338 (1990)