TOWARDS A COMPREHENSIVE GRASPING SYSTEM FOR ARMAR-III

Nicolas Gorges, B. Heinz Wörn

Institute for Process Control and Robotics
University of Karlsruhe (TH)
Karlsruhe, Germany

Alexander Bierbaum, Rüdiger Dillmann

Institute of Computer Science and Engineering
University of Karlsruhe (TH)
Karlsruhe, Germany

ABSTRACT

In this paper, we present our approach of merging two existing grasping systems into one system that comprises vision and touch. The first system is a low-level control system which deals with coordination issues and is closely linked to the hand, arm and the tactile sensor system. This system is responsible for the grasp execution. The second one starts from the perception level and deals with object detection and grasp planning. The missing link is a path planner which transfers a chosen grasp plan to the executive control system. The proposed grasping system will be evaluated in a test scenario and be integrated into the new demonstrator robot of our collaborative research center, ARMAR III.

1. INTRODUCTION

The special research area "Humanoid Robots" (formal name SFB 588) aims at the development of a robot system which interacts and cooperates with human beings. The ability to grasp and manipulate common objects is essential for supporting the human with everyday tasks. The improved dexterity of the new SFB 588 demonstrator ARMAR- III allows the realization of new evaluation scenarios but also implies the need for the integration of a new comprehensive grasping system. The humanoid robot ARMAR-III is described in detail in [1].

The grasping procedure requires different levels of planning, controlling and coordination. In order to grasp and manipulate objects, the robot needs e.g. to drive its actuators to the intended position. To get there it needs an adequate sensor system and requires processing of the sensor data. The sensors on the other side need to be embedded into the controlling system.

Grasping is a major issue in robotics. In this context, visual servoing and visually guided grasping is an essential research topic. In the last year also the research of grasping with tactile feedback has become popular again. Still, most of the grasping systems focus on one sensor modality and do not combine them into one comprehensive system.

We will present an approach to merge two concepts of already existing grasping systems to fulfill the given requirements and to establish a basis for ARMAR-III. On the one hand, we have a bottom-up grasping system developed at the IPR (Institute for Process Control and Robotics) which focuses on the controlling level and which is closely connected to arm, hand and the tactile sensor system. It performs the grasp execution which includes coordination and synchronization processes like hand-arm coordination.

On the other hand, we have a top-down grasping system developed at the CSE/IAIM (Computer Science and Engineering/Industrial Applications of Computer Science and Micro Systems) which starts from the visual perception level. The perception incorporates an object detection component which feeds a grasp planning system based on 3D object models. This system is described in detail in [2]. The missing link is a path planning system which connects both systems - it builds an extension of the high-level grasping system and an interface to the low-level coordination framework.

Talking of grasping systems it is important to mention the approach of Kragic et al [3] which also comprises vision system, grasp planner and a robot hand as executive unit. This system needs intervention by a human operator for planning the grasp for an identified and localized object. Further, their object recognition by vision is limited to planar-faced objects. The components of the system presented here comprise an integrated grasp planning module which is directly linked to the object detector and a vision system that can identify and localize arbitrarily complex objects.

2. SYSTEM

At first, we show the mechanical design of arm and hand which is essential for the dexterity - it determines the fundamental skills of the robot. The design of the hand determines what kind of objects can be grasped and what kind of manipulations can be performed. Consecutively, we describe the proposed grasping system which will be tested on the evaluation platform and then be ported to the central demonstrator robot of our collaborative research center shown in Fig. 3.

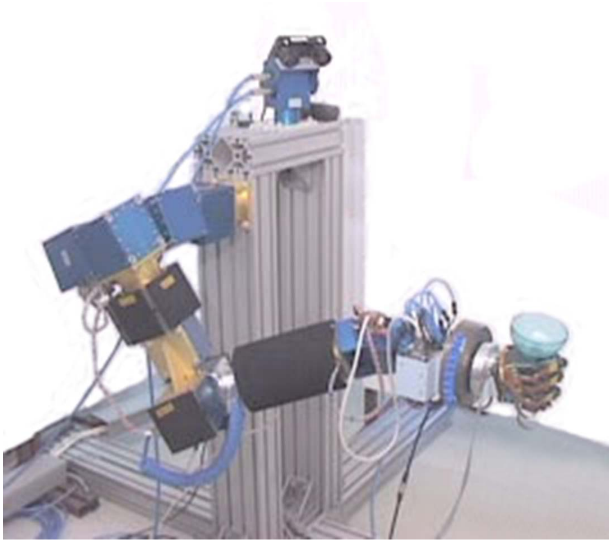

Fig. 1. The current robotic evaluation platform.

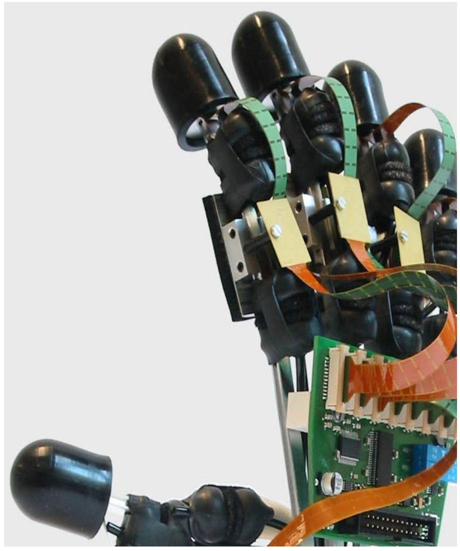

Fig. 2. Hand protype with tactile sensors

2.1. Evaluation platform

The current evaluation platform consists of a 7 DOF arm of amtec PowerCubes, a pneumatic hand, and a pan-tilt unit carrying a stereo camera. The hand has been developed by our project partners at the IAI in the Karlsruhe Research Center FZK (reference?). The arm and the hand are equipped with tactile sensors on the upper arm, elbow, forearm, hand shafts and finger tips. A 6 DOF force-torque sensor at the wrist is used to measure 3 DOF forces and 3 DOF torques. The robot is pictured in Fig. 1, and a close-up of the hand equipped with tactile sensors in Fig. 2.

2.2. Grasp strategies

The underlying grasping approach is based on the idea that certain objects are linked with a corresponding grasp. The grasps to be performed in most of the scenarios can be restricted to a base of relatively simple geometric shapes. An offline grasp planer can compute an adequate grasp for the basis shapes and store them in a grasp library. For performing a grasp online, the object to be grasped is sampled into a set of basis shapes. Then, an adequate grasp skill must now be selected from the database and transferred to the current shape.

The execution of a grasp is decomposed into four phases. Each of these phases is represented by different hand configurations, arm trajectories, velocity of the arm movement, sensibility of the hands' tactile sensors.

1. **Coarse approach:** The vision system detects the object and gives a position estimate. The arm is moving with comparatively high velocity and the hand moves to pre-grasp configuration.
2. **Fine approach:** The arm is moving with slower fine movement and the hand is waiting for contact.
3. **Grasp object:** The arm position is adjusted with small movements while the finger move towards objects. Contact forces are are increased to desired values while grasp configuration is checked.
4. **Depart:** Slower fine movements are performed with fixed Cartesian orientation of the hand.

These four phases follow from the concept of synchronization and coordination of arm-hand movements (c.c. 2.3.4) which requires a discrete presentation. Each of these phases is represented by different hand configurations, arm trajectories, velocity of the arm movement and each of them reacts differently on the input of the sensor system. As soon as the hand or the arm reaches a certain state, the next step of the synchronization process is triggered. This behavior can be adapted according to the current grasp task.

2.3. High-level grasp control

The control architecture describes the basic building blocks that make up the robot grasping system and the relations they have with each other. It is depicted in Fig. 4. The high-level robot control symbolizes the interface to a superior control structure. This can be a task planner of a robot or an interface to the human. However, it determines the grasping task according to the current situation which involves the command of grasping a specific object and selecting a grasp type. The object detector identifies objects and provides the high-level robot control with information about the environment. The involved modules of this grasping system are described in the following sections.

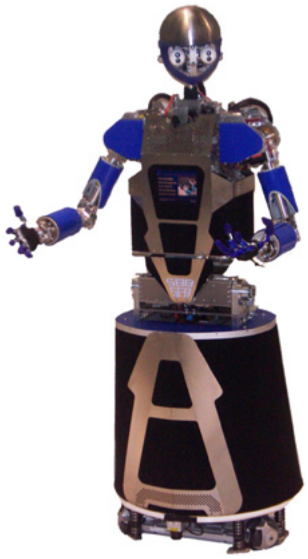

Fig. 3. The demonstrator robot of our collaborative research center.

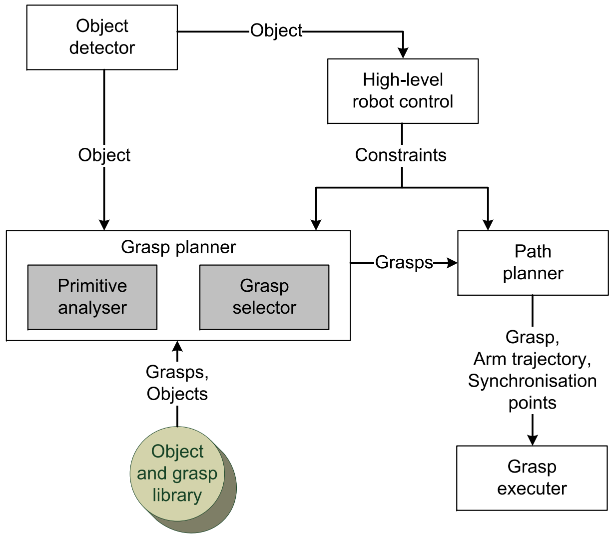

Fig. 4. High-level grasp control

Fig. 5. Object segmentation by color

2.3.1. Object detector

The vision system has to perform object recognition and localization as the grasp planner must be provided with information about identity and location of the object that is to be grasped. To achieve this, the system has to compensate for several problems. As it is intended for use on a mobile robot it must be operating properly on a moving platform, thus object segmentation can not be performed by simple background subtraction, the recognition must be invariant against different perspectives. As the objects to be recognized might be scattered arbitrarily in the scene, recognition must also be invariant against 3D rotation and translation. The objects must become fully localized in 6D space,i.e. including their orientation. All of these tasks should be accomplished in real-time, which means it should take an interval in the dimension of the frame rate period for calculating the information described above.

We have combined visual recognition and localization systems for two classes of objects: objects that can be segmented by a single color and objects that can be recognized by their texture features. The focus for the vision system design was laid on appropriate performance for grasp planning and grasp execution by a humanoid mobile robot. For the unicolor global appearance-based object recognition we deploy the algorithm from [4] which is suitable for uniformly colored objects. This approach uses a non-adaptive color model, which is sufficient for constant lighting conditions as in our test environment. During a learning phase the dataset for different views of an object is generated automatically using a 3D-model. For that reason this method can be regarded as a combination of an appearance-based with a model-based visual recognition system. These datasets are provided with orientation information from the generated model views. For recognition, candidate regions are segmented from the camera image and matched with the formerly acquired datasets. The matching process is realtime capable, with a database of five objects it takes approximately 5 ms for one potential region to analyze. An example of this is shown in figure 5. Localization of an object via stereo triangulation of the matched regions centroids in the left and right image is supported. The orientation of a matched object in the dataset can be determined with an accuracy of +/-5mm within the robot demonstrators workspace. The example database used for unicolored objects includes different types of dishes like cups and plates.

For recognition of objects with textured surface appearance a different approach is used [5] which is capable of identifying and discriminating textured objects like printed Tetra Pak boxes. The method is appearance-based as it only deploys trained images but no models of the objects. For feature extraction, patches from the intensity image with extreme gradient appearance or extreme gradient curvature are considered, as shown in figure 6. In the learning phase these patches are stored as original and as warped transformations, which creates translation and orientation invariant datasets. To compensate for illumination conditions the patches are normal-

Fig. 6. Object segmentation by texture

ized. The database representations are clustered to increase computational efficiency and provide more accurate results by the tree-search in the recognition phase. The algorithm is tolerant to occlusion if enough features can be extracted from the image. Though, the algorithm can naturally not compensate for reflections on the surface of objects. The recognition performance is about 350ms for a 20 object database on a Pentium 4 PC with 3GHz. In the near future the system will be extended to provide orientation and localization information, which is implicitly already available as parameter of the correspondence search.

2.3.2. Grasp planner

The grasp planner module can access a global database with CAD-models of objects. The current database includes cups, plates, bottles or primitive bodies like cubes. The grasp planning process is currently executed offline using the program *GraspIt!* [6]. *GraspIt!* is a robotic grasping simulator, which uses geometric models of the robot hand and objects in a virtual workspace to determine feasibility and quality of a grasp. It provides collision detection for rigid bodies, considers material friction in the contact model and calculates a quality measure for contacts between hand and object. New robot hands and object models can be added to the simulator. As described in [7] the program was extended to support the SFB588 humanoid robot hand for planning. This robot hand has eight joints which are partially coupled leading to four degrees of freedom for actuation. Despite this restriction the planner takes into account the full eight degrees of freedom as a hand with full control over all joints will soon be available and simulation of the coupled joints is complex. In the simulator we can evaluate with this model for three power grasp types (hook,cylindrical,spherical) and two precision grasp types (pinch,tripod) which represent a subset of the grasp taxonomy of Cutkosky [8].

The output of the planner module is a set of parametrized grasps comprising the following information:

- Grasp starting point (GSP)
- Grasp approaching vector
- Hand orientation rotated around the axis of approaching vector. This is specific to robot hand and grasp type.
- Preshape posture and joint closing velocities. This is specific to the grasp type.

The description created by the planner only covers starting conditions, not final conditions, which makes it suitable for robot hands without tactile sensors and force control. Therefore, obviously the planner does not cover reactive or dynamic grasping scenarios. The dataset is generated automatically with primitive models of objects following [9]: for a given object the planner generates suitable GSPs and approaching vectors and starts testing the available grasp types for the robot hand. For power grasps it simulates the hand approaching the object with full aperture until contact is detected and then closes it. If the quality of the contact is below a certain threshold the planner repeats the test with the hand closing at an increased distance until the quality measure indicates maximum stability for the grasp. For precision grasps the test sequence is different: from the GSP the hand is continuously tested for contact while closing from the preshape posture at regular distance intervals on the approaching vector. All grasps for which a quality measure value above a certain threshold is calculated, are stored in the grasp dataset of the particular object.

2.3.3. Path planner

The adapted path planner [10] is based on a static scene graph of the environment and takes the shapes and kinematics of the robot hand and arm into account. The framework has to be enhanced to allow a dynamic scene where the human and the robot interact with the environment. After a firm grasp with an object is established, the path planner must also consider that the object is now part of the kinematic chain.

For synchronization of hand and arm movements, the execution of a grasp is divided into four phases (c.c.). For each of these phases, a 3D-point (6DOF) the path planner specifies the position of the hand root where the next step of the synchronization process is triggered. The first synchronization point is given by the starting position. The second one must be chosen within a safety distance to the object and to the environment as the arm is moving with higher velocity. The third one is given by the pre-grasp position computed by the grasp planner and the fourth point is given by the depart position.

As the grasp planner does not completely consider the kinematic of the arm, a desired grasp position might not be reachable. Therefore, the path planner get a set of potential grasp and process these grasps until a valid configuration is planned. A process loop is shown in the following:

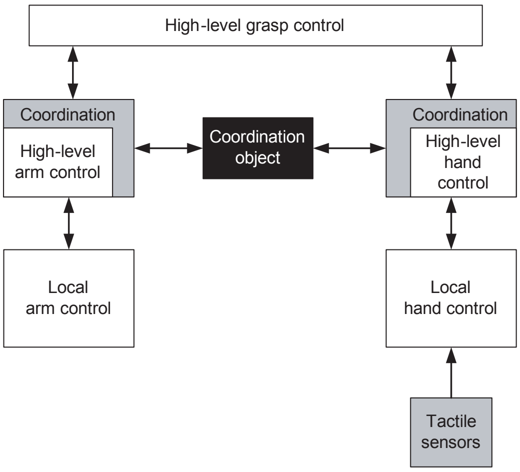

Fig. 7. Overview, low-level grasp control and interfacing to upper level

1. Take a grasp candidate. The grasp candidate contains the pre-grasp finger positions and the grasp configuration.

2. Compute a collision-free path for the arm from the starting position to the pre-grasp position. Compute a collision-free path for fingers from the pre-grasp position to the grasp position.

3. Choose the first three synchronization points for the hand-arm coordination process.

4. Compute a collision-free path from the grasp position to the desired depart position.

2.3.4. *Grasp execution*

The grasp execution is responsible for setting the parameters of the low-level components and for observing of their correct execution. Figure 7 illustrates the control framework as proposed in [11] [12] and shows the different components which are involved in the low-level control of such a grasping process.

At the top is the high level robot control which sets the parameters of the low level components and supervises the whole grasping procedure. On the next lower level are high level controllers. These are able to communicate with their neighboring components via the coordination object in order to synchronize and coordinate their activities. This communication is independent from the high level robot control and increases therefore the efficiency of the control system. Grasping needs a coordination and synchronization of hand and arm movements. The arm places the hand in the right grasping positions before. After performing a grasp, the hand has to depart. The coordination object determines thereby the behavior of the hand-arm coordination.

For parametrization, the high-level arm control component get the trajectory of the arm movement and the synchronization points for hand-arm coordination. The high level hand control needs the pre-grasp and grasp finger configuration as well as the finger trajectories.

3. EVALUATION

Finally, the grasping system has to cope with certain benchmark scenarios. The benchmarks are currently restricted by the design and the integration level of the mechanical components of the robot. Therefore, we consider only the use of one hand at the moment and focus on the scenarios listed in the following:

1. Grasping an known object from a known position

2. Give and take objects to/from an human

3. Grasping known objects from unknown positions

Results will be included in the final paper.

4. CONCLUSION

To enable the robot to perform dextrous fine manipulation in general, an adequate mechanical and control system is needed. We have shown that basis skills can already be performed with the first version of our hand and with the proposed grasping framework. The robot hand is capable to grasp a certain range of object using vision and touch. The new hand to be built will have a significantly increased functionality and allow tasks like fine manipulation and tactile exploration.

Currently, the grasp planner requires geometric models of the objects to be investigated for grasp planning. As these object models need to be created a-priori, the planning process is executed offline. There is no module which creates a geometric model from the perception of the vision system. The vision system recognizes an object and makes information about location, orientation and identity available to the grasp selector. For the future, we plan to implement a more versatile but possibly less exact geometric component recognition and estimation which could be directly fed to the grasp planner in an online grasp planning scenario. The capability of estimating unknown geometries in 3D space by vision is a key feature for our planned approach in online grasping of a-priori unknown objects. Finally, future work will also include visual servoing in an extended coordination framework to increase the accuracy of the hand positioning.

5. REFERENCES

[1] T. Asfour, K. Regenstein, P. Azad, J. Schroeder, A. Bierbaum, N. Vahrenkamp, and R.Dillmann, “Armar-iii: An

integrated humanoid platform for sensory-motor control," in *Submitted to: IEEE-RAS International Conference on Humanoid Robots, Genova, Italy*, 2006.

[2] A. Morales, T. Asfour, P. Azad, S. Knoop, and R. Dillmann, "Integrated grasp planning and visual object localization for a humanoid robot with five-fingered hands," in *To appear in: International Conference on Intelligent Robots and Systems (IROS), Beijing, China*, 2006.

[3] D. Kragic, A. T. Miller, and P. K. Allen, "Real-time tracking meets online grasp planning," in *IEEE International Conference on Robotics and Automation, Seoul, Republic of Korea*, 2001.

[4] P. Azad, T. Asfour, and R. Dillmann, "Combining appearance-based and model-based methods for real-time object recognition and 6d localization," in *To appear in: Proc. International Conference on Intelligent Robots and Systems (IROS), Beijing, China*, 2006.

[5] K. Welke, P. Azad, and R. Dillmann, "Fast and robust feature-based recognition of multiple objects," in *Submitted to: IEEE-RAS International Conference on Humanoid Robots, Genova, Italy*, 2006.

[6] A. T. Miller and P. K. Allen, "A versatile simulator for robotic grasping," in *IEEE Robotics & Automation Magazine*, 2004, vol. 11.

[7] A. Morales, P. Azad, and A. Kargov, "An anthropomorphic grasping approach for an assistant humanoid robot," in *Procceddings of the 37th International Symposium on Robotics, Munich, Germany*, 2006.

[8] M. Cutkosky, "On grasp choice, grasp models and the design of hands for manufacturing tasks," in *IEEE Transactions on Robotics and Automation*, 1989, vol. 5.

[9] A. T. Miller and S. Knoop, "Automatic grasp planning using shape primitives," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2003.

[10] D. Mages, B. Hein, and H. Wörn, "Introduction of additional information to heuristic path planning algorithms," in *IEEE Conference on Mechtronics and Robotics*, Aachen, Germany, 2004, pp. 1023–1028, Paul Drews.

[11] Dirk Osswald, Sadi Yigit, Oliver Kerpa, Catherina Burghart, and Heinz Woern, "Arm-Hand-Koordination eines anthropomorphen Roboterarms zur Mensch-Roboter Kooperation," in *18. Fachgespräch "Autonome Mobile Systeme" (AMS)*, Karlsruhe, Germany, 2003.

[12] Dirk Osswald, Sadi Yigit, Catherina Burghart, and Heinz Woern, "Programming of Humanoid-Robots with Movement-Objects for Interactive Tasks," in *Proceedings of the Int. Conf. on Computer, Communication and Control Technologies (CCCT)*, Orlando, FL, 2003.