

A framework for visual servoing

Danica Kragic and Henrik I Christensen

Centre for Autonomous Systems,
Royal Institute of Technology,
S-10044 Stockholm, Sweden,
{danik,hic}@nada.kth.se
<http://www.nada.kth.se/~danik>

Abstract. We consider typical manipulation tasks in terms of a service robot framework. Given a task at hand, such as "Pick up the cup from the dinner table", we present a number of different visual systems required to accomplish the task. A standard robot platform with a PUMA560 on the top is used for experimental evaluation. The classical *approach-align-grasp* idea is used to design a manipulation system. Here, both visual and tactile feedback is used to accomplish the given task. In terms of image processing, we start by a recognition system which provides a 2D estimate of the object position in the image. Thereafter, a 2D tracking system is presented and used to maintain the object in the field of view during an approach stage. For the alignment stage, two systems are available. The first is a model based tracking system that estimates the complete pose/velocity of the object. The second system is based on corner matching and estimates homography between two images. In terms of tactile feedback, we present a grasping system that, at this stage, performs power grasps. The main objective here is to compensate for minor errors in object position/orientation estimate caused by the vision system.

1 Introduction

Robotic visual servoing and manipulation has received significant attention during the past few years. Still, most of the existing systems rely on one visual servoing control strategy or one sensory modality. This commonly limits the system to concentrate on one of the *approach-align-grasp* steps. It has been pointed out that one of the key research areas in the field of visual servoing is the integration of existing techniques, regarding both the estimation and control, [1].

In terms of robotic appliances for service robotics, it is of inevitable importance to observe the complete robotic task. Assuming basic *fetch-and-carry* tasks, there are varying demands for precision and degrees of freedom in control depending on complexity. As proposed in [2], a key to solving robotic hand-eye tasks efficiently and robustly is to identify how precise control is needed at a particular time during task execution. The required level of precision should then be matched with appropriate sensory input. This is also one of the main ideas pursued in our work.

We consider three levels for an object manipulation sequence, see Fig. 1:

- **Transport** considers motion of the robot platform and/or the robot arm to the vicinity of the object. From this position, the arm should be able to reach the object without moving the base.
- **Alignment** of the hand with the object such that a grasp can be performed.
- **Grasping** of the object which can be performed using tactile feedback or in a predefined open-loop manner.

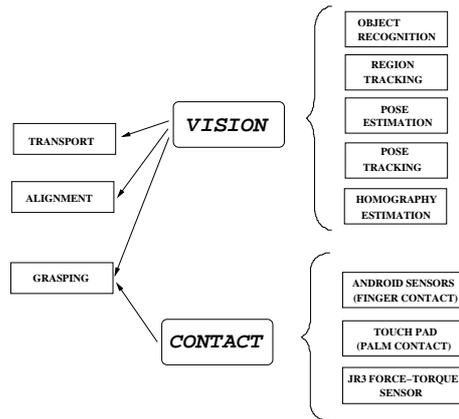


Fig. 1. Robot control versus sensory feedback hierarchy. The required complexity and type of feedback depends on the current step of a manipulation task.

Our main goal is to present a number of different techniques that allow robots to perform manipulation tasks in real world scenarios according to the above. We will show how visual and tactile feedback can be used together with a number of visual servoing strategies to manipulate simple, everyday objects. We do not offer a general solution from a system point of view, rather a first step towards it. Compared to our previous work [3], where the main consideration was the overall control framework and systems integration, here we concentrate on the actual building blocks. In particular, visual and tactile feedback and underlying servoing strategies are studied. We believe that our approach is relatively easy to build upon - each of the individual techniques can easily be extended and combined to perform more complex tasks. Similarly, in human physiology there is a differentiation between identification of the object, ballistic motion to the proximity of the object, preshaping of the hand, alignment, and interaction. Here, the identification and ballistic motion are fused into a single task for convenience.

The paper is organized as follows. In Section 2 basic control strategies for visual servoing are presented together with commonly facilitated camera/robot configurations. In Section 3 image processing algorithms currently used in the system are briefly presented. The strategies for using tactile feedback for grasping are discussed in Section 4. The experimental platform and few experiments are presented in Section 5. And finally,

Section 6 discusses the current limitations of the system and provides topics for future research.

2 Visual Servoing

In terms of the design of visual servoing systems, there are three major issues that have to be considered: i) the choice of control law, ii) camera-robot configuration, and iii) the choice algorithms used to provide the feedback for the control loop. We will touch upon the first two issues briefly in Section 2.1 and Section 2.2. Section 3 outlines the image processing algorithms currently available in our system.

2.1 Control design

There are two basic approaches to visual servo control [4]: i) image-based visual servoing (IBVS) and position-based visual servoing (PBVS). In IBVS, an error signal is measured in the image and then mapped directly to robot motion commands. In PBVS, features are extracted from the image and then used to compute a partial or a complete pose/velocity of the object. An error is then computed in the task space and thereafter used by the control system. To overcome the problems of IBVS and PBVS systems, several hybrid systems have been proposed [10], [11]. In general, these systems decouple the translational and rotational part of the control signal achieving the desired stability of the system even for cases where the difference between the start and desired pose of the robot is significant. In our system, all three strategies are used.

2.2 Camera-Robot Configurations

Fig. 2 shows some of the most common camera-robot configurations typically used in visual servoing systems. In our system, according to the figure, we are using a combination of VM1 and VM4 which is a special case of VM5. The available configuration usually determines the design of the feedback system. For example, an eye-in-hand camera configuration commonly requires fast image processing (since the image changes with each motion of the arm) as well as the flexibility in terms of scale. Since there is a significant difference between the start and the destination pose, 2 1/2 D approach is commonly adopted control strategy [10], [12]. A stereo stand-alone system requires less features per image and, for the case of static targets, the appearance of the features may remain almost constant throughout the visual servo sequence.

There are numerous examples where one or the other control approach or configuration will perform better. To that end, we have decided to use a number of different systems and use them depending on the task at hand and at the level of detail/complexity needed to perform the given task.

3 Transportation and Alignment

The following sections give a short overview of the image processing methods currently exploited in the system.



Fig. 2. Most common camera–robot configurations: monocular eye–in–hand, monocular stand–alone, binocular eye–in–hand, binocular stand–alone and redundant camera system. In our system, we are using a combination of VM1 and VM4 which is a special case of VM5.

3.1 Recognition

The object to be manipulated is first recognized using the view-based SVM (support vector machine) system presented in [7]. The recognition step delivers the image position and approximate size of the image region occupied by the object. This information is then used i) either by the tracking system to track the part of the image, the *window of attention*, occupied by the object while the robot approaches it, or ii) by the stereo system to provide a rough estimate of the object’s 3D position. Recent research on human vision has clearly demonstrated that *re*-cognition of prior known objects can be efficiently modeled as a view based process [16], [17], which motivates our use of an SVM based approach to recognition.

3.2 Region tracking

Our tracking system is based on integration of multiple visual cues using *voting*, [5]. The visual cues used are motion, color, correlation and intensity variation. Cues are fused using weighted super-position and the most appropriate action is selected according to a winner-take-all strategy. The advantage of the voting approach for integration is the fact that information of different cues can be easily combined without the need for explicit models as it is for example in Bayesian approaches. Lots of perceptual experiments support the idea that when it comes to aspects of visual scenes, people most likely mention color, form and motion as being quite distinct. There is a belief that information about form, color, motion and depth is processed separately in the visual system. However, it has also been shown that the segregation is not complete and there is a cross-talk among different cues [18].

3.3 Pose estimation and tracking

Our model-based tracking system integrates the use of both appearance based and geometrical models to estimate the position and orientation of the object relative to the camera/robot coordinate system, [6]. There are basically three steps in the system:

- *Initialization* - here, Principle Component Analysis (PCA) is used to provide an approximation to the current object pose.

- *Pose Estimation* - To estimate the true pose, the initialization step is followed by a local fitting method that uses a geometric model of the object. This is made possible by the fact that we deal with an object that has already been recognized and thus its model is known. The method used here was proposed in [8].
- *Pose Tracking* - If the object or the camera start to move, the system will provide a real-time estimate of the object pose. Again, the method proposed in [8] is used.

It has been shown in [20] that visuo-motor actions such as grasping, use the actual size of the object and that the position and orientation are computed in egocentric frames of reference. Thus, human reaching movements are planned in spatial coordinates, not in joint space. If an accurate pose of the target is available together with a good arm model (which is true in our case), one can use the ideas proposed in [19] to generate human-like arm trajectories. In our case, both IBVS and PBVS are used as demonstrated in [5].

3.4 Homography based matching

Using a stored image taken from the reference position, the manipulator can be moved in such a way that the current camera view is gradually changed to match the stored reference view (*teach-by-showing* approach). Accomplishing this for general scenes is difficult, but a robust system can be made under the assumption that the objects are piecewise planar. In our system, a wide baseline matching algorithm is employed to establish point correspondences between the current and the reference image [12]. The point correspondences enable the computation of a homography H relating the two views which is then used for 2 1/2D visual servoing.

4 Grasping

The following sections give a short overview of the current grasping strategies where tactile and force-torque feedback are considered. The grasping system is still in its initial phase and does not perform any intelligent grasp planning. The main objective here was to design a system which will be able to perform a grasp even if the pose of the object is not perfectly known. Therefore, the current implementation considers only power grasps.

4.1 Grasp modeling

After the arm (hand) is aligned with the object, grasping can be performed. Using the available pose estimate and tactile feedback, the grasping system compensates for minor errors in the pose estimate. The grasping strategy is formulated using finite state machines (FSM) [14]. Using the general idea proposed by [15], the basic states, q_i of a FSM are shown in Fig.3. These states basically mimic the human grasping procedure. In addition, Fig.3 shows the actions, a_i , needed to execute the grasp. Also, basic conditions, e_i under which the actions, a_i are running are outlined.

For the control of grasping, our three-fingered Barrett hand has been equipped with two types of tactile sensors. The palm is equipped with a touch pad for detection of palm

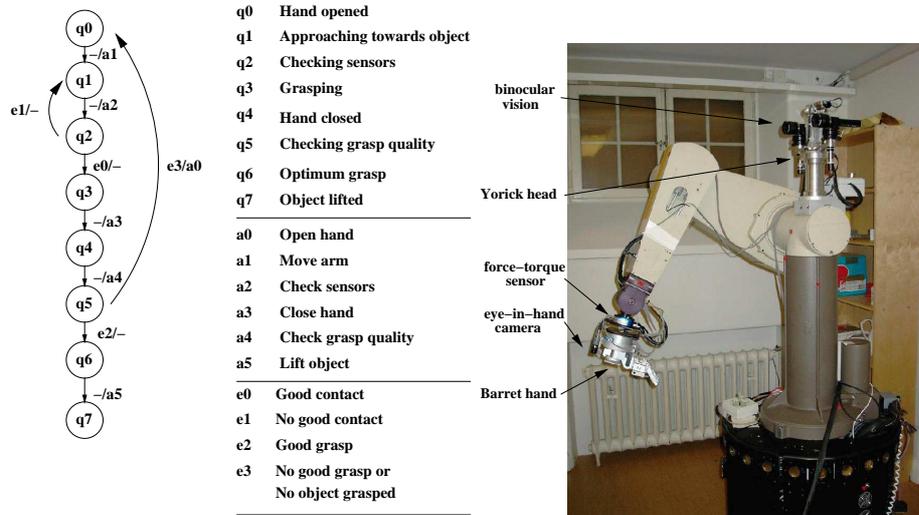


Fig. 3. Left) Minimized abstract representation of the grasping process, and Right) XR4000 equipped with an eye-in-hand camera, stereo head, JR3 force-torques sensor, and Barret hand with tactile sensors.

contacts. In addition, each link of the three fingers have basic sensors for detection of contact. In addition, the arm has a force torque sensor for overall sensing of hand forces. The details of the setup are described in Section 5.1. To achieve the desired flexibility of the system, ideas from behavior-based planning were used, see [13] for details.

5 Experimental evaluation

In this section, a few examples are presented to demonstrate the system. Since the task is given (i.e. “Robot, pick up the raisins.”), the object to be dealt with is known in advance as well as the transport-align-grasp strategy to be used. This is explained in more detail with each of the examples.

5.1 Experimental platform

The experimental platform is a Nomadic Technologies XR4000 equipped with a Puma 560 arm for manipulation (see Fig. 3). The robot has two rings of sonars, a SICK laser scanner, a wrist mounted force/torque sensor (JR3), and a color CCD camera mounted on the gripper (Barrett Hand). On the robot shoulder, there is a Yorick robot head providing a stereo visual input. The palm of the Barrett hand is covered by a VersaPad sensor. The VersaPad was designed to be used as a touch pad on a laptop. It reports the following: i) a Boolean value if the pad is active (contact occurred), ii) the coordinates of the contact point, and iii) pressure at the contact point. On each finger link, an Android sensor is placed. It reports the pressure applied on the link. The wrist mounted

JR3 force–torque sensor is here primarily used as a “safety–break” for the system: if the contact occurs on the VersaPad’s “blind” spot, it can still be felt by the JR3 sensor.

5.2 Example 1.

This example shows the basic idea for “Robot, pick up the raisins” task. The object is first located in the scene using the recognition system, see Fig. 4. The object is polyhedral and in this case a homography based approach is used during the align step. The current image is compared with the stored image of the object as presented in Section 3.4. 2 1/2 D visual servoing is used to control the motion of the robot. After the hand is aligned with the object, an open–loop grasp strategy is performed to pick-up the object. A few example images during servoing onto and grasping a package of raisins are shown in Fig. 5.



Fig. 4. a) and b) Images observed by the left and right head camera, respectively. The windows show the position of the raisins package estimated by the recognition system. c) image observed by the eye–in–hand camera, and d) destination image used for 2 1/2 D visual servoing.

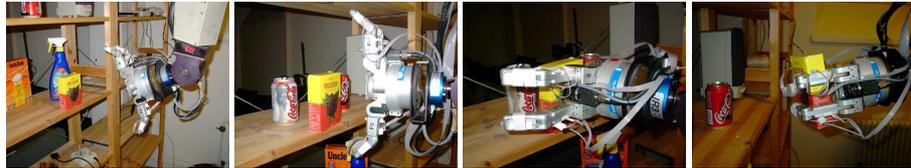


Fig. 5. A few example images during servoing onto and grasping a package of raisins.

5.3 Example 2.

Fig.6 shows an example where model-based pose estimation/tracking system is used to estimate the complete pose of the object and then align the gripper with it. After that, the object can be grasped. Since the model of the object is available, it is enough to use one camera during the whole servoing sequence.



Fig. 6. The basic idea of our approach: after the object is recognized, 2D tracking is used to approach the object. After that, the appearance based approach followed by a local fitting stage is used to estimate the current pose of the object. After that, simple grasping can be performed.

5.4 Example 3.

Fig. 7 shows the image position of two example objects (a soda bottle and a cleaner item) estimated by the visual recognition system. Using the knowledge of the head camera intrinsic and extrinsic parameters, an approximate 3D position of the object is estimated. It is assumed that the object is in vertical position. The arm is then aligned with the object so that the palm of the hand is facing the object, see Fig.8. Finally, the hand is moved towards the object in the direction orthogonal to the palm plane.



Fig. 7. a) and b) The position of a soda bottle, and c) and d) the position of a cleaner bottle in left and right head camera images estimated by the recognition system.

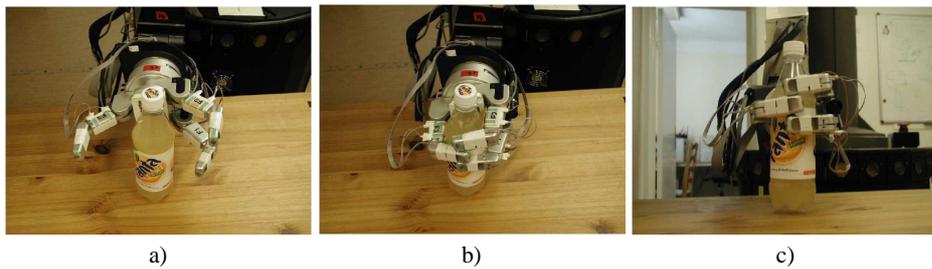


Fig. 8. a) Approaching the bottle, b) Grasping, and c) Lifting.

5.5 Example 4.

Fig.9 shows an example grasping sequence where the Android sensors on the hand's fingers are used during grasping. Here, a contact with one of the fingers occurs before the contact with the palm. The hand is driven in the horizontal plane to center the object inside the hand. The hand is moved until contact with each of the fingers is reported. If no contact is reported from one of the sensors, the principle described in the previous experiment is used.

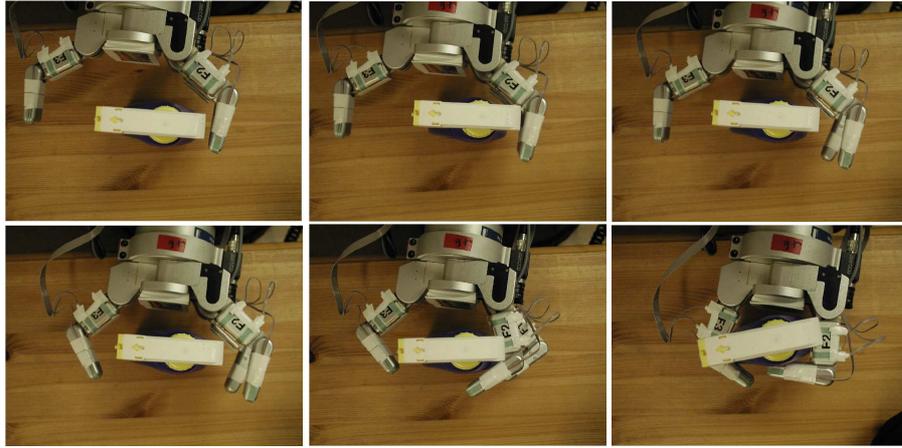


Fig. 9. Grasping a cleaner bottle (see text for detailed description).

6 Conclusions and Future work

We have discussed major building blocks of a typical manipulation tasks in terms of a service robot. Assuming a task such as 'Pick up the cup from the dinner table', we have presented a number of different visual systems required to accomplish the task. The classical transport-align-grasp strategy was used to choose between available feedback systems. Both visual and tactile feedback were used to accomplish the given task.

In terms of image processing used during the transportation step, we have presented a recognition system which provides a position estimate of the object, and a 2D tracking system used to keep the object in the field of view. For the alignment step, two systems are available. The first is a model based tracking system that estimates the complete pose/velocity of the object. The second system is based on corner matching and estimates homography between two images. In terms of tactile feedback, we have presented a grasping system that currently performs power grasps. The main objective here was the design of a system capable of compensating for minor errors in object position/orientation estimate caused by the vision system.

References

1. Workshop on visual servoing, *IEEE International Conference on Intelligent Robots and Systems, IROS2002*, Lausanne, Switzerland, 2002
2. Dodds, Z., Jägersand, M., Hager G., and Toyama K.: A hierarchical vision architecture for robotic manipulation tasks, *International Conference on Computer Vision Systems, ICVS'99*, pp. 312-331, 1999
3. Petersson, L., Jensfelt, P., Tell, D., Strandberg, M., Kragic, D. and Christensen, H.: Systems Integration for Real-World Manipulation Tasks, *IEEE International Conference on Robotics and Automation, ICRA 2002*, pp. 2500-2505, vol 3, 2002
4. Hutchinson, S., Hager, G., and Corke P.: A tutorial on visual servo control, *IEEE Transactions on Robotics and Automation*, **12**(5), pp. 651–670, 1996.
5. Kragic, D.: Visual servoing for Manipulation: Robustness and Integration Issues, PhD thesis, Computational Vision and Active Perception Laboratory (CVAP), Royal Institute of Technology, June, 2001.
6. Kragic, D. and Christensen, H.: Model Based Techniques for Robotic Servoing and Grasping, *IEEE International Conference on Intelligent Robots and Systems, IROS2002*, pp. 2002
7. Roobaert, D.: Pedagogical Support Vector Learning: A Pure Learning Approach to Object Recognition, PhD thesis, Computational Vision and Active Perception Laboratory (CVAP), Royal Institute of Technology, May, 2001.
8. T.W. Drummond and R. Cipolla. Real-time tracking of multiple articulated structures in multiple views. *ECCV'00*, 2:20–36.
9. L. Petersson, D. Austin, and H.I. Christensen. "DCA: A Distributed Control Architecture for Robotics", *IEEE International Conference on Intelligent Robots and Systems, IROS2001*, pp. 2361–2368, vol 3, 2001
10. Malis, E., Chaumette, F., and Boudet, S.: Positioning a coarse-calibrated camera with respect to an unknown object by 2-1/2-d visual servoing. *IEEE International Conference on Robotics and Automation, ICRA98*, pp. 1352–1359, 1998.
11. Deguchi, K.: Optimal motion control for image-based visual servoing by decoupling translation and rotation, *IEEE International Conference on Intelligent Robots and Systems, IROS1998*, pp. 705-711, 1998.
12. Tell, D.: Wide baseline matching with applications to visual servoing, PhD thesis, Computational Vision and Active Perception Laboratory (CVAP), Royal Institute of Technology, June, Stockholm, Sweden, 2002.
13. Crinier, S.: Behavior-based Control of a Robot Hand Using Tactile Sensors, Master thesis, Centre for Autonomous Systems, Royal Institute of Technology, December, 2002.
14. Katz, R.H.: Contemporary logic design, Benjamin Cummings/Addison Wesley Publishing Company, 1993
15. Horswill, I.D.: Behavior-Based Robotics, Behavior Design, Technical report CS 395, Northwestern University, 2000
16. Tarr, M.J., and Bulthoff, H.: Object recognition on man, monkey and machine, *International Journal of Cognitive Science*, Vol. 69, No. 1-2, July 1998.
17. Edelman, S.: Representation and Recognition in Vision, *MIT Press*, Cambridge, MA. 1999.
18. Palmer, S.E.: Vision Science: Photons to Phenomenology, *MIT Press*, Cambridge, MA. 1999
19. Goodman, S.R. and Gottlieb, G.G.: Analysis of kinematic invariances of multijoint reaching movements, *Biological Cybernetics*, vol 73, pp.311-322, 1995
20. Hu, Y. and Goodale, M.: Constraints in Human visuomotor systems, *IEEE International Conference on Intelligent Robots and Systems, IROS 2000*, pp. 1633-1638, vol 2, 2000