1—1 Using Available Potentials of Vision-Based Robots by Converting Passive Watching into Active Seeing

Minh-Chinh Nguyen
Institute of Informatics
Technical University Munich

Volker Graefe
Institute of Measurement Science
Bundeswehr University Munich

Abstract

An approach to make use of available potentials of uncalibrated vision-guided robots is introduced. It allows the robot to manipulate differently shaped objects that may be located anywhere in its workspace, even if they are not visible in the initial fields of view of the cameras. Key points of the approach are the conversion of passive "watching" into active "seeing", so that the fields of camera view scan the robot's whole workspace, and a direct transition from image coordinates to motion control commands, so that the need for a calibration of the robot and of the vision system is eliminated.

1 Introduction

Object manipulation by a robot independent of any inbuilt quantitative models of the robot and of pre-defined numerical values of any parameter would never need any explicit calibration and, therefore, promises great advantages in terms of robustness and cost of ownership [1]. In recent years a variety of methods have been developed for reaching such advantages. For example, [2] avoid using the manipulator's exact kinematic model and the camera parameters by performing a self-calibration at four known points, combined with the use of visual feedback. [3] uses the visual-motor Jacobian matrix approximating the dependencies of visual features on motor changes valid around the current system configuration. [4], [5] avoid the necessity of calibration for robots by a direct transition from visual input information to robot control commands. However, in their demonstrations these methods have used the assumption that the object to be grasped must be visible in both images, i.e., the robot's workspace is limited to the initial fields of view of the cameras. Moreover, even if a sensor with a 180-degrees field of view was available, it would not be possible to cover the robot's whole available operating space. Such systems do not make use of the robot's available potentials and, thus, reduce its efficiency.

Overcoming such shortcomings is the focal point of this article. Key points of the approach presented in the sequel are the introduction of a searching behavior and a direct transition from image coordinates to motion control commands of a robot.

Fig. 1: The objects used in our experiments

2 Problem Statement

Various objects (Fig. 1) are to be recognized and grasped by a 5-DOF manipulator equipped with a motor-operated parallel-jawed gripper and a stereo vision system. The objects may be located in nearly arbitrary orientations and positions anywhere in the robot's work space and need not be visible in the initial fields of view of the cameras. None of the coefficients characterizing the arm and the cameras are known (completely uncalibrated system).

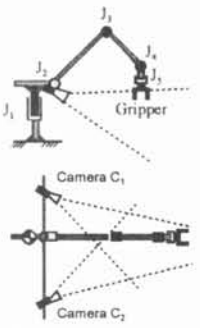

Fig. 2: The robot arm joints, the arrangement of the cameras and their fields of view.

The arm. The arm's 5 DOF correspond to the 5 joints J_1 to J_5. Joint J_5 refers to the gripper's rotation around its symmetry axis. Joint J_1 allows the arm to be rotated around a vertical axis. The remaining joints J_2, J_3 and J_4 allow the gripper to be moved within a certain section of a vertical plane, the work plane. In our experiments joint J_4 was normally controlled in such a way that the gripper was in a vertical orientation (Fig. 2).

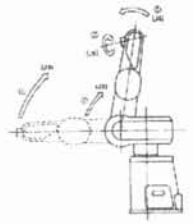

Fig. 3: The nest configuration of the arm [6]

The arm's configuration is internally represented by a set of 5 joint coordinates. The joint coordinates are abstract state variables corresponding to the arm's 5 joints. The relationship between joint coordinates and the actual joint angles and, thus, the robot's corresponding physical configuration are considered unknown.

The arm is controlled by sending commands to the arm's drive unit. The following commands are available:

- *Nest: NT*

It must be the first one after switching the robot on. It causes the arm to assume a certain configuration as defined by the manufacturer (Fig. 3). By definition, all joint coordinates are zero in the nest configuration.

- *Move joints: MJ (<control word vector>)*

The control word vector consists of 5 real numbers (control words) where each one of the control words corresponds to the motion being performed by one of the joints. Suppose, the joint coordinates were A, B, C, D, E, respectively. Executing the command MJ (a, b, c, d, e) changes the joint coordinates to A+a, B+b, C+c, D+d, E+e.

Joints	Permissible range of joint coordinates
J_1	0 . . . -300
J_2	0 . . . -130
J_3	0 . . . 120
J_4	0 . . . 220
J_5	0 . . . -360

Table 1: Permissible ranges of joint coordinates (relative to the nest configuration of the arm, derived from [6])

- *Read error status: ER*

Normally the error status is zero. When a command that would cause at least one joint coordinate to exceed its permitted range (Tab. 1) is sent to the drive unit, the error status is set to 2, whilst it is set to 1 if a hardware failure is detected. The command that caused the error condition is not executed and all further commands are ignored unless the error status is first reset. No infor-

mation is provided as to which one of the 5 control word parameters has caused the error condition. The erroneous parameter may, however, be found by breaking the command down into a series of five commands with one non-zero parameter each.

- *Reset error status: RS*

The error status is reset to zero, allowing subsequent motion commands to be executed.

- *Open/close gripper: GO, GC*

It should be noted that no command exists for reading joint coordinates or joint angles. (In principle such information is available, but it is hidden by the drive unit).

The sensors. The sensors of the robot are two video cameras of unknown characteristics which are mounted on the arm and rotate around joint J_1 together with the arm, but relative to the robot's work plane they are fixed (Fig. 2). The cameras are mounted in a rather unstable way to make the impossibility of any calibration or precise adjustment obvious, and to allow easy random modifications of the camera arrangement. The cameras' locations and orientations are somewhat arbitrary and not exactly known, but each camera is mounted in such a way that its field of view covers that area of the work plane where the gripper is supposed to work.

The camera images are processed by a vision system, implemented on two processors TMS320C40 (Texas Instruments), one processor for each camera. The vision system performs all necessary object recognition regarding the detection of the gripper and the objects [6], the object classification [7] and the determination of the poses of the gripper and the object, i.e., their reference points and orientations in the images [5].

3 Control Strategy Overview

To allow vision-based manipulation, both the object to be manipulated and the gripper must be visible in both images. At the start of a manipulation task the arm is at the nest configuration and the gripper is not visible in the images. Regarding the object, one out of the following 3 situations exists:

a) The object is visible in the images of both cameras.

b) The object is visible in the image of one camera only.

c) The object is not visible in any of the images.

If one of the situations, b) or c), exists, a search motion of the cameras and the arm with a rotation around joint J_1, i.e., an "object search", is executed until situation a) exists. The object search is described in section 4. To determine if the object is visible in an image, a detection algorithm ("object detector") is executed separately in both images. If the object is found in an image, the object detector delivers the image coordinates of the object's reference point in that image. The object detection is not described in this article.

After the manipulated object has been detected, the gripper is moved without visual feedback from the nest configuration to a "start position" where it is visible in both images (section 5). Then the gripper is moved to the object under vision-based control and the object is grasped (Sections 6, 7).

4 Object Search

We assume that the object is located somewhere in the robot's workspace, i.e., within the reach of the arm and in the potential fields of view of the cameras. We are implicitly assuming that the work space actually exists, in other words, that the cameras are arranged in such a way that their fields of view are partly overlapping, and that part of the common field of view is accessible to the gripper. Since the cameras rotate together with the arm (joint J_1), the workspace has a toroidal shape.

If the result of at least one object detector is that no object has been detected in its image, a search motion is initiated. The principle of the object search is the conversion of passive "watching" into active "seeing" by rotating the cameras so that their fields of view scan the robot's whole workspace. Due to the way the cameras are attached to the arm that we used for our experiments, only one degree of freedom, the rotation of the joint J_1 (Fig. 2), is available for this motion. The action actually to be taken depends on which one of the three possible cases exists:

(1) *The object is visible in both images:*

Obviously, the object search is not required and simply terminated successfully.

(2) The object is not visible in any of the images:

In this case, the joint coordinate C_1 associated with joint J_1 should be modified. The problem is, however, how it should be modified. If it is modified by a small amount, the search requires many steps and is slow; if the modification is large, the search is fast, but it may happen that the fields of view of the cameras before and after the motion do not overlap, in which case an object might be overlooked. Therefore, C_1 should be modified by an *optimal increment (optinc)* so that the search is as fast as possible, while the fields of view of the cameras before and after each motion still overlap.

For determining this *optinc* the partial derivatives, or gain coefficients, $\partial x/\partial C_1$ and $\partial y/\partial C_1$, relating the image coordinates, x and y, of an observed feature to the joint coordinate, C_1, must be known. The robot learns these gain coefficients by modifying C_1 by a small amount and evaluating the resulting image motions, as described in the sequel.

We suppose that the robot's surroundings are inhomogeneous and that some features can be seen in the images. The gain coefficients can then be determined by observing the image displacement of features after the execution of a motion command.

An interesting question is, by what amount C_1 should be modified in the beginning of the learning phase, when nothing is known yet about the gain coefficients. If ΔC_1, the change of C_1, is too small, the image displacement is too small to be recognized. If it is too large, the image displacement is too large for tracking the extracted features.

After switching the robot on, the arm is brought to the nest position (Fig. 3). At first, the proper sign of ΔC_1, and of *optinc*, is learned by a trial-and-error method: The robot modifies C_1 by the smallest positive value allowed for the corresponding parameter of the command MJ, e.g., 10^{-1} [6]. If no error occurs after this modification was carried out, it can be concluded that the chosen sign is right. Otherwise, ΔC_1, and *optinc*, must be negative.

To track the extracted feature easily, C_1 should be modified by an amount, ΔC_1, that is large enough for the resulting image displacement to be recognized, but not too large. Such a ΔC_1 can be determined on the assumptions that the illumination condition in the robot's surroundings is approximately homogeneous and the robot's surroundings are not ideally homogeneous, i.e., the images perceived before and after the modification of C_1 will be

different. The image motion can be estimated by comparing pixel by pixel the gray levels of the two images before and after the motion. If at some point in the images the gray level difference is greater than a *certain threshold*, which corresponds to the noise amplitude, it is assumed that the difference was caused by the motion. A suitable value for the threshold can be easily determined by comparing the gray level differences of three or four different images perceived at different times without moving the cameras [2]. The maximum gray level difference between those images is the result looked for.

To determine the value of the gain coefficients the robot first modifies C_1 by the smallest possible amount with the determined sign and observes the resulting image motion by comparing the images as described above. If no image motion has resulted, the amount of the change, ΔC_1, is doubled and this step is repeated. C_1 is then modified by the determined ΔC_1 as long as the minimum displacement of the features extracted from the images can be recognized, which is, for instance, in our experiments about 10 pixels. This learning phase requires, of course, a certain amount of time since a number of motion steps have to be performed and each time the whole image has to be evaluated.

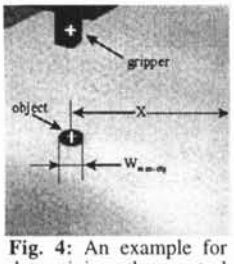

Fig. 4: An example for determining the control word increment.

Having learned the gain coefficients the robot may next determine a value for *optinc* that lets the fields of view of the cameras overlap after each modification of C_1 by at least the width of the object in the image, but – in the interest of speed – not by much more. The object's motion trajectory in the image is a complex line, but for reasons of simplification, we assume that it is approximately a line coinciding with an image row (Fig. 4; X is the distance between the object center and the vertical image margin to which the object will move).

After determining *optinc* it should be checked if it is right as expected. This can be easily achieved by modifying C_1 by *optinc*. If, after the modification, the whole object is still in the image, *optinc* is correct.

After the determination of *optinc* the last task of object search is to find the object to be manipulated in the robot's workspace. The robot achieves this task by modifying C_1 by the determined *optinc* step by step, and thus scanning the workspace with the cameras. If after a step an object is found in both images, the search finishes successfully. Otherwise, it terminates without success, after the robot's whole workspace has been searched.

It often happens that the object is found first in only one image. In this case, the program is switched to the case (3) that is described in sequel.

(3) The object is visible in the image of one camera only

In this case, the robot first learns to determine the sign of *optinc* and the values of the gain coefficients in a similar way as described for case (2) above. However, differently from the above case, the gain coefficients can be learned directly by observing and evaluating the image motion of the detected object instead of the features in the whole image, i.e., only a small image area surrounding the detected object is used. With this, the time required for learning can be reduced noticeably. As in case (2), the value of *optinc* is determined in such a way that it would cause an as large as possible image displacement of the object with the whole object still in the image as before.

If the object that was originally visible in the image of one camera disappears before it appears in the image of the second camera, the object search will finish unsuccessfully. This may happen due to an unfavorable arrangement of the cameras.

If the search terminates unsuccessfully, no object that can be grasped exists in the robot's workspace. Therefore, the arm is then brought to the nest position. Otherwise, the gripper is brought to a *start position* by activating the initial gripper positioning behavior that is described in the following section.

In principle, the workspace could be expanded without changing the kinematics of the arm by providing an additional degree of freedom (tilt) for the camera motion.

5 Initial Gripper Positioning

The remaining activities for accomplishing a grasping operation require the robot's gripper to be visible for both cameras. In principle, this may be accomplished in two ways: either the cameras or the gripper may be moved until the gripper is within the fields of view of both cameras. Our robot has its cameras fixed relative to the first movable link between J_1 and J_2 of the robot (Fig. 2). Therefore, specific control commands are sent to the motors of J_2, J_3, and J_4 causing the gripper to move to a predefined start position. The start position may be freely chosen, provided that the arm can move to it from the nest position without collision, and it allows the gripper to be seen by both cameras.

Strictly speaking, this implies a deviation from our goal of the calibration-free robot, since we must know a priori those numerical values of the parameters of the control commands which will make the gripper move to a suitable start position. However, only a weak, approximate calibration is required, since it suffices to move the gripper to any position where it may be seen by both cameras. Such a weak calibration is so easy to perform that, for practical purposes, a robot requiring only a weak calibration may be considered equivalent to a calibration-free robot.

6 Object Approach

Since the object approach was already described by [5], it is only briefly sketched here.

Let us assume that the gripper has been brought to the start point, and both gripper and object are visible in both camera images. They are modeled by their (suitably chosen) reference points [5]. For grasping the object the gripper must be moved to where the object is; in other words, the reference points of the gripper and the object must coincide. The two points coincide in the real world if, and only if, they coincide in both camera images, regardless of any particular characteristics of the camera or the robot. The task of the object approach procedure is then considered equivalent to making the two reference points coincide in the *images* of both cameras. The key idea here is that we are not at all concerned with the distances, coordinates, or any other relations in the real world, but only with the image coordinates of visible features.

The robot accomplishes the rendezvous between the gripper and the object by modifying the contents of C_1, C_2 and C_3 associated with J_1, J_2, and J_3 respectively, by a small amount and observing the effects in the image. It then estimates the gain coefficients relating image motions to the

commanded control words and computes by linear extrapolation those control words that would bring the gripper to the object in both images if the system were linear. Since linearity is not guaranteed and collisions should be avoided, it executes only a fraction of the computed motion, say about 80 %, which brings the gripper closer to the target than it was before. After some iterations the rendezvous is accomplished.

7 Objects of General Shape and Orientation

While in the implementation of [8] only objects with rotational symmetry could be handled, that first implementation of the calibration-free approach was augmented by [5] and [7] to allow a greater variety of objects to be manipulated (Fig. 1). To grasp such objects the gripper must not only be in the right position, but also in the right orientation. If the objects are either lying flat on a table or standing upright, the gripper's axis may be vertical, and a rotation of the gripper around J_5 is sufficient for obtaining a suitable orientation of the gripper.

The correct orientation of the gripper may be reached by first bringing the gripper to an intermediate point [5] where it is near the object in the camera images. Then the joint coordinate C_5, associated with J_5, is modified until the gripper edge that characterizes the gripper orientation is parallel to the object orientation in the image [5]. Since the gripper is in a vertical orientation, and the object edge characterizing the object orientation is horizontal, this will occur simultaneously in both images. Even if the gripper orientation is exactly parallel to the object axis in the images, due to perspective distortion this does not mean that they are exactly parallel in the world, too. However, if the gripper is near the object, the angle difference is small enough to allow the object to be grasped.

8 Experimental Results

The approach outlined above was tested in a series of searching and grasping experiments. In the experiments the object was detected, located and grasped reliably, regardless of its initial location in the robot's 3-D work space.

We have also performed separate experiments to check the object search behavior by locating an object at different positions. When it was seen near the left margin of the image, then modifying C_1 by *optinc*, as determined by the method described above, caused the object to move to the right image margin, as it should.

9 Conclusions and Outlook

We have introduced a search behavior for locating objects to be manipulated by a vision-guided robot. The objects may initially be anywhere in the robot's 3-D workspace and need not be visible in the initial fields of view of the cameras. We have realized it on an uncalibrated vision-guided robot.

The robot can learn the gain coefficients relating image motions to motor control words by observing either the detected object or, if no object is visible, features of the natural inhomogeneity of the robot's surroundings without a need of special landmarks. In contrast to our earlier systems [5], [6], [8] the robot now can make full use of its available potential, and its efficiency is increased.

The concepts introduced here will be further developed and will be implemented on a 6-DOF manipulator that is part of a humanoid service robot.

References

[1] V. Graefe, "Calibration-Free Robots." *Proc. The 9th Intel. System Symp. Japan Society of Mechanical Engineers,* pp. 27-35, 1999.

[2] R. Cipolla, N. J. Hollinghurst, "Visually-guided grasping in unstructured environments." *Robotics and Automation*, Elsevier Pub., pp. 337-346, 1997.

[3] M. Jägersand, R. Nelson, "Visual Space Task Specification, Planning and Control," *IEEE Int. Symp. on computer Vision,* pp. 521-526, 1995.

[4] M.-C. Nguyen, V. Graefe, "Visual Recognition of Objects for Manipulating by Calibration-Free Robots." *In Kenneth, et al. (eds.): Machine Vision Applications in Industrial Inspection VIII.* Proc. of IST/SPIE, Vol. 3966, pp. 290-298, San Jose, 1999.

[5] M.-C. Nguyen, V. Graefe, "Stereo Vision-Guided Object Grasping." *Int. Sym. on Automotive Technology and Automation ISATA'99;* Proc. of Advanced Manufacturing in the Automotive Industry. Vienna, pp. 77-85, 1999.

[6] Mitsubishi, "Industrieroboter: Bedienungsanleitung RV-M2," *Mitsubishi Electric Europe GmbH,* 1993.

[7] M.-C. Nguyen, V. Graefe, "Stereo Vision-Based Object Classification." *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS '00.* Takamatsu, pp. 76-81, 2000.

[8] **V. Graefe,** "Object- and Behavior-oriented Stereo Vision for Robust and Adaptive Robot Control." *International Symposium on Microsystems, Intelligent Materials, and Robots.* Sendai, pp. 560-563.