Issues and Experimental Results in Vision-Guided Robotic Grasping of Static or Moving Objects

CHRISTOPHER E. SMITH
Department of Computer Science and Engineering, University of Colorado at Denver, Campus Box 109, P.O. Box 173364, Denver, CO 80217-3364, U.S.A.

NIKOLAOS P. PAPANIKOLOPOULOS
Department of Computer Science and Engineering, University of Minnesota, 200 Union St. SE, Minneapolis, MN 55455, U.S.A.

Issues and Experimental Results in Vision-Guided Robotic Grasping of Static or Moving Objects

ABSTRACT: *Many research efforts have turned to sensing, and in particular computer vision, to create more flexible robotic systems. Computer vision is often required to provide data for the grasping of a target. Using a vision system for grasping of static or moving objects presents several issues with respect to sensing, control, and system configuration. This paper presents some of these issues in concert with the options available to the researcher and the trade-offs to be expected when integrating a vision system with a robotic system for the purpose of grasping objects. The paper includes a description of our experimental system and contains experimental results from a particular configuration that characterize the type and frequency of errors encountered while performing various vision-guided grasping tasks. These error classes and their frequency of occurrence lend insight into the problems encountered during visual grasping and into the possible solution of these problems.*

Keywords: Robotic Grasping, Visual Tracking, Feature Selection, Eye-in-Hand Robotic Systems, Sensor Placement

1 Introduction

In the field of robotics, sensors have long been viewed as providing the solution to many difficult problems involving the interaction of the robot with its environment. Rarely has this promise been met. Typically, sensing fails either from overly ambitious goals for the sensing system or from unanticipated problems relating to the sensor, the robot, and the integration of their respective systems. In this paper, we address the issues related to sensing for robotic tasks by analyzing a standard robotic sensor problem: providing vision-based data (in this case a sequence of greyscale images) for the grasping of a target (see Figure 1).

In most grasping applications, the desired effect of incorporating a vision sensor into the task structure is to increase the success rate of the robotic grasp. We assume that the need for a sensor implies that the position of the object with respect to the manipulator is unknown or partially known, otherwise the sensor would be providing redundant information. Certainly there are other reasons for sensing, such as object recognition (one knows “where,” but not “what”), object detection (one knows “what” and “where,” but not “when”), and so forth; however, it is the uncertainty of “where” that typically dominates the decision to use vision-based robotics.

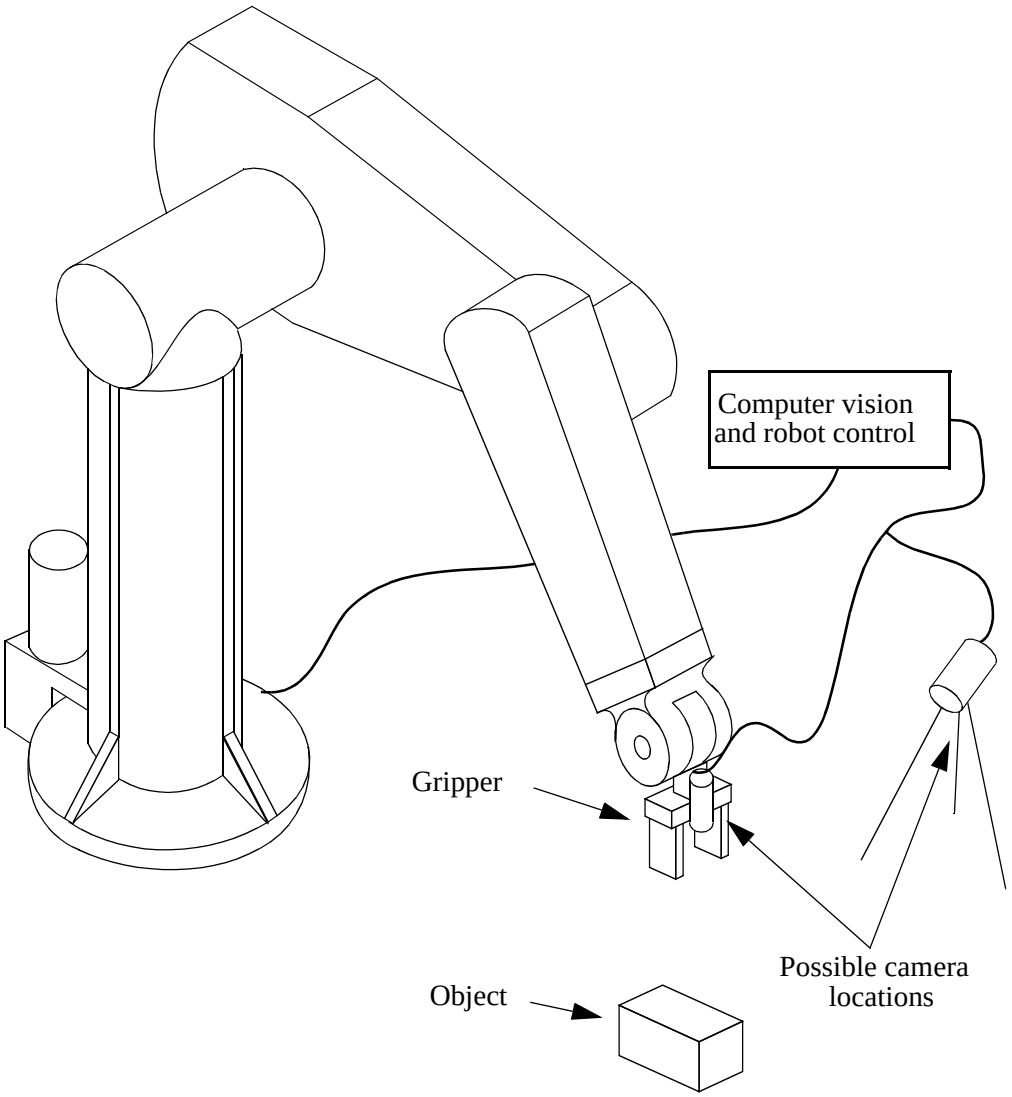

Figure 1: Components of the Vision-Guided Grasping Problem

In this paper, we present an overview of the work related to vision-guided grasping and the design decisions researchers made when addressing sensor system and robot system issues. We then re-examine these issues from a design viewpoint and present some of the alternatives and trade-offs that are key to these issues. Next, we highlight the design choices we have made while developing a vision-guided grasping system, driven in part by the goal of high success rate grasping. We also present the results of extensive runs of grasping experiments, including a discussion of the problems we encountered and our solutions to those problems. Finally, we summarize our results and present the unresolved issues related to vision-based grasping in general and our system in particular. We also use these issues as the motivation for future work and present what we

consider to be the next direction for our research. It is important to mention that systems like the one proposed in this paper can be used in assembly and microassembly operations, in the automobile industry for painting and welding, in nuclear waste cleanup efforts, and in agricultural applications (e.g., citrus picking).

2 Prior Research

A number of research efforts have focused on the problem of using vision information in the execution of various robot control tasks, several of which addressed problems that are related to vision-based robotic grasping. Several researchers have investigated catching and juggling (Kimura, Mukai, and Slotine, 1992) (Rizzi and Koditschek, 1993) (Sakaguchi, Fujita, Watanabe, and Miyazaki, 1993). They chose to use open-loop control, thus raising issues of camera-to-manipulator coordinate transform calibration and of the accuracy and precision of the robotic system. This choice avoided issues regarding using a robot-mounted camera or using vision to sense the position of the end-effector. Two of these systems used vision sensors only to predict the parabolic trajectory of a dropping or tossed object (Kimura, Mukai, and Slotine, 1992) (Sakaguchi, Fujita, Watanabe, and Miyazaki, 1993) and then blindly moved the end-effector to a position along the parabola prior to the arrival of the object. This avoided real-time issues that arose when attempting to sense while the manipulator is in motion; however, the accuracy of the effected motion and problems of graspable versus ungraspable configurations of the objects became relevant concerns.

Prior work in the use of visual information specifically for grasping has resulted in only a limited number of efforts. Many of these proposed systems used a static camera and a calibrated coordinate transformation from the camera frame to the manipulator frame. Additionally, these systems typically used open-loop control. In particular, several efforts used vision only to gather environmental information before performing a blind grasp.

Houshangi (Houshangi, 1990) developed a system to grasp targets exhibiting planar motion. The system utilized a static camera and pre-computed manipulator poses for the start position and the grasp position. This simplified the issues of calibrated camera-to-manipulator transforma-

tions, but limited the flexibility of the system. It was assumed that the target's geometry and pose were known, that the motion was only translational in the X and Y components, and that the velocity was constant over time (i.e., no speed and no direction changes). By limiting these factors, the control of the manipulator was simplified, and the control system could accurately predict object position for a given future point in time. There was no discussion regarding the use of visual feedback during the grasping motion, raising questions of sensitivity to sensor noise and to manipulator positioning accuracy when using this configuration.

Kimura *et al.* (Kimura, Mukai, and Slotine, 1992) proposed a system capable of catching a free flying ball using a four-degree of freedom manipulator. The target's geometry (a sphere) rendered the object graspable from any approach angle and thus eliminated any issues regarding approach angles and alignment constraints. Additionally, since the object was tossed, the target's trajectory was assumed to be planar, translational motion of known acceleration (specifically, acceleration due to gravity). This allowed the prediction of the trajectory by the vision system, reducing the system's dependency upon a real-time vision system. The camera position was static and it was assumed that any motion energy in the view of the camera corresponded to the object to be caught.

Allen *et al.* (Allen, Timcenko, Yoshimi, and Michelman, 1993) presented a system that tracks a target moving in an oval path and grasps the target when tracking becomes stable. The grasping motion used no sensor feedback and was performed quickly enough to negate the effects of any changes in the motion of the target, solving several problems inherent to using vision during the manipulator's reach. This did, however, raise concerns regarding accuracy of the grasp and ability of the system to compensate for errors in calibration and for noise in the sensor.

3 Issues for Vision-Based Grasping

In this section we explore the issues and trade-offs that must be considered when designing a system to performing vision-based grasping of objects. Several of the relevant issues have already been introduced in the discussion of the previous work in the area. We will attempt to clarify the issues that need to be addressed, offer some of the trade-offs related to these issues, and

detail our own design decisions that were made during the development of our vision-guided grasping system (Smith and Papanikolopoulos, 1995a)(Smith and Papanikolopoulos, 1995b) based upon the MRVT (Brandt, Smith, and Papanikolopoulos, 1994).

3.1 Open-Loop Versus Closed-Loop Control

One of the most basic design issues in any vision-based robotic system is the choice of open- or closed-loop control. Many grasping systems use open-loop control to eliminate the need to either sense the end-effector via the vision system, or to use an eye-in-hand system where manipulator motion introduces camera motion. Often, open-loop control is required since the processing of the vision system is too slow for real-time control or the vision system is used only to determine the object's position and orientation prior to a blind grasp. Closed-loop control requires that the visual data is used as a feedback signal in the manipulator control and requires vision processing with acceptable speed and delay factors for real-time applications. Closed-loop control also allows visual data to compensate for manipulator positioning inaccuracies and (to a limited extent) sensor noise. Typically, open-loop control requires more accurate calibration of the camera-manipulator system while closed-loop control requires more and faster vision hardware.

If closed-loop control is selected, then the selection of a control algorithm and the incorporation of the visual data into the control scheme becomes an issue. Trade-offs between adaptive versus non-adaptive control arise as do alternatives for the objective of the controller and for the measurement of error in the system.

For our system, we chose an adaptive, closed-loop controller to reduce calibration requirements and to provide what we considered a better potential for the subsequent extension of the system to moving objects that exhibit non-constant motion (Smith and Papanikolopoulos, 1995a)(Smith and Papanikolopoulos, 1995b). The controller is a based upon a repositioning controller that attempts to drive object feature points to selected positions on the image plane (see (Smith and Papanikolopoulos, 1995b) for a discussion of the control law). One may argue that the use of iteration and learning may help us design a good grasping system. We disagree with this, since the tolerances required are very tight. Furthermore, the adaptive controller has a perfor-

mance that can be predicted and analyzed mathematically (we can predict when the grasping will be unsuccessful).

3.2 Blind or Visually-Guided Grasps

Many vision-based robotic systems for grasping attempt to determine the 3-dimensional workspace of the manipulator via vision and then perform a blind grasp of the target object. This decision relaxes real-time constraints since the manipulator is not guided to the object by vision data, but it places a premium on the calibration and accuracy of the vision system and the certainty with which the camera's position is known relative to the manipulator. Small errors in calculated object positions can cause the system to fail when attempting the blind grasp.

Visual guidance of the grasp offers a system that can more easily recover from the effects of sensor noise and errors in calibration and can more easily be adapted to the grasping of objects exhibiting unknown, non-constant motion. However, it requires that the vision processing period and delays are appropriate for real-time control.

We emphasize visual guidance throughout the grasp since we believe that this offers the best possibility for successful grasps under uncertainty with respect to manipulator positioning, visual data, and calibration (Smith and Papanikolopoulos, 1995a)(Smith and Papanikolopoulos, 1995b).

3.3 Monocular, Stereo, or Structured Light Vision

Stereo and structured light systems share the issue of baseline calibration. While stereo systems can be designed that tolerate small errors in calibration, the canonical stereo system requires baseline calibration and solution of the correspondence problem. Likewise, structured light systems typically use a laser striper and a camera with a calibrated baseline to determine the three-dimensional position of an object. To achieve equivalent performance, the stereo system requires either twice as much vision processing hardware, or hardware that is twice as fast as that required by a structured light system (since there are two images to process).

A monocular system shares the vision hardware advantage that structured light systems hold over stereo. In addition, there is no baseline that requires calibration. Unfortunately, monocular systems cannot provide the three-dimensional location of an object without employing some

method to recover depth (see (Smith and Papanikolopoulos, 1994)) or it must assume a calibrated ground plane that is not parallel to the image plane of the camera. Typically, monocular systems attempt to define tasks by relating the motion of features on the image plane to the motion of objects in the workspace. We have selected a monocular approach (Smith and Papanikolopoulos, 1995a) to grasping that utilizes a repositioning controller to effect changes in the pose of the manipulator with respect to an object to be grasped. This choice was made to reduce hardware requirements, to eliminate the correspondence problem, and to eliminate baseline calibration.

3.4 Camera Placement

The choice of a mounting position for the camera can cause drastic changes in the basic system design. Camera positioning usually involves a choice between a statically mounted camera and a robot mounted camera (eye-in-hand configuration). The static mount removes the problems associated with a moving camera, but introduces issues related to camera-to-manipulator coordinate transformation calibration, occlusion of the object by the manipulator, and what assumptions (e.g., ground plane) or configurations (e.g., stereo or structured light) must be used to derive the three-dimensional position of the object.

The robot-mounted camera can eliminate the need for accurate calibration, stereo or structured light systems (in most cases), and assumptions about the workspace (e.g., ground plane assumptions). The robot mount usually has problems with a moving camera (if the actual grasp is to be vision-guided), a wide range of operating depths dependent upon the manipulator configuration, and a potential for the manipulator's position to cause lighting changes in the workspace.

We have chosen to use an eye-in-hand configuration for several reasons (Smith and Papanikolopoulos, 1995a). Primarily, we want a closed-loop system, but we do not wish to incur the cost of using the vision system to sense the position of the end-effector, as discussed previously. We also wish to avoid expensive and possibly repetitive calibration processes that could affect the reliability of the system. Finally, we hope to avoid occlusion of the object view by part of the manipulator by selecting this configuration.

3.5 Object Geometry

We use rectangular prisms with one graspable dimension (Smith and Papanikolopoulos, 1995a)(Smith and Papanikolopoulos, 1995b), thus requiring the consideration of gripper alignment without needing a grasp planner or object recognition system (at this point). This choice balances object complexity and system complexity in a way that emphasizes vision-guided grasping, rather than recognition or planning.

4 Experimental Results

As presented in previous work, we have implemented and tested our system for visually-guided grasping on both static object and moving object grasps. We used the MRVT system for testing (see Figure 2). The MRVT is a multi-architectural system with two main parts: the Robot/Control Subsystem (RCS) and the Vision Processing Subsystem (VPS). The RCS consists of a PUMA 560 manipulator (see Figure 3), its associated Unimation Computer/Controller, and a VERSA Module Eurocard (VME) Single Board Computer (SBC). The manipulator's trajectory is controlled under VAL II via the Unimation Controller's Alter line and requires path control updates once every 28 msec. Those updates are provided by an Ironics 68030 VME SBC running Carnegie Mellon University's CHIMERA real-time environment. A Sun SPARC 330 serves as the CHIMERA host and shares its VME bus with the Ironics SBC via BiT-3 VME-to-VME bus extenders.

The VPS receives input from a Panasonic GP-KS102 miniature camera that is mounted parallel to the end-effector of the PUMA and provides a video signal to a Datacube system for processing (the camera has a focal length of 7.5mm). The Datacube is the main component of the VPS and consists of a Motorola MVME-147 SBC running OS-9, a Datacube MaxVideo20 video processor, a Datacube Max860 vector processor, and a BiT-3 VME-to-VME bus extender. The bus extender allows the VPS and the RCS to communicate via shared memory, eliminating the need for expensive serial communication. The VPS performs the optical flow, calculates the desired control input, and supplies the input vector via shared memory to the Ironics processor for inclusion as an input into the robot control software. The video processing and calculations

required to produce the desired control input are performed under a pipeline programming model using Datacube's Imageflow libraries. We detect and track at any time four feature points using the SSD algorithms described in (Brandt, Smith, and Papanikolopoulos, 1994). The processing of these features lasts 32 ms. Our experiments were performed under regular laboratory lighting conditions.

Additionally, a serial port on the MVME-147 is dedicated to gripper control. When the VPS identifies that the conditions required for grasping have been met, it signals the Unimation Controller to close the gripper via an inexpensive, custom hardware interface. The hardware interface is a Motorola 68HC11 microcontroller that reads a byte of data from a serial interface and outputs the byte as high and low signals on eight digital output lines. Only one output line is used as the gripper control line.

Our earlier work comprised preliminary results of the system, but it did not categorize failure types and analyze the results of extended numbers of random experiments. We performed this type of analysis on static object grasps in order to evaluate the system and suggest modifications to improve system performance. When system performance met our criteria for success, we tested the system on a preliminary series of moving object grasps to determine the success rate.

Three sets of experiments were conducted. The first set exposes a problem in the feature selection and reselection process. The second set demonstrates that the improved version meets our target success rate for static grasping, while still suggesting improvements that might be made to the system. The third set shows that the system performance for moving grasps achieves promising results during initial experiments.

4.1 Experimental Design

The object's position with respect to the end-effector was varied by a vector (? *x*? ? *y*? ? *z*) and the rotation about the Z-axis (optical axis) was varied by an amount ? ? . Each delta was derived using a uniform random number generator and was bounded by a maximum and a minimum chosen for each delta. The deltas were applied to an initial position that corresponded to the

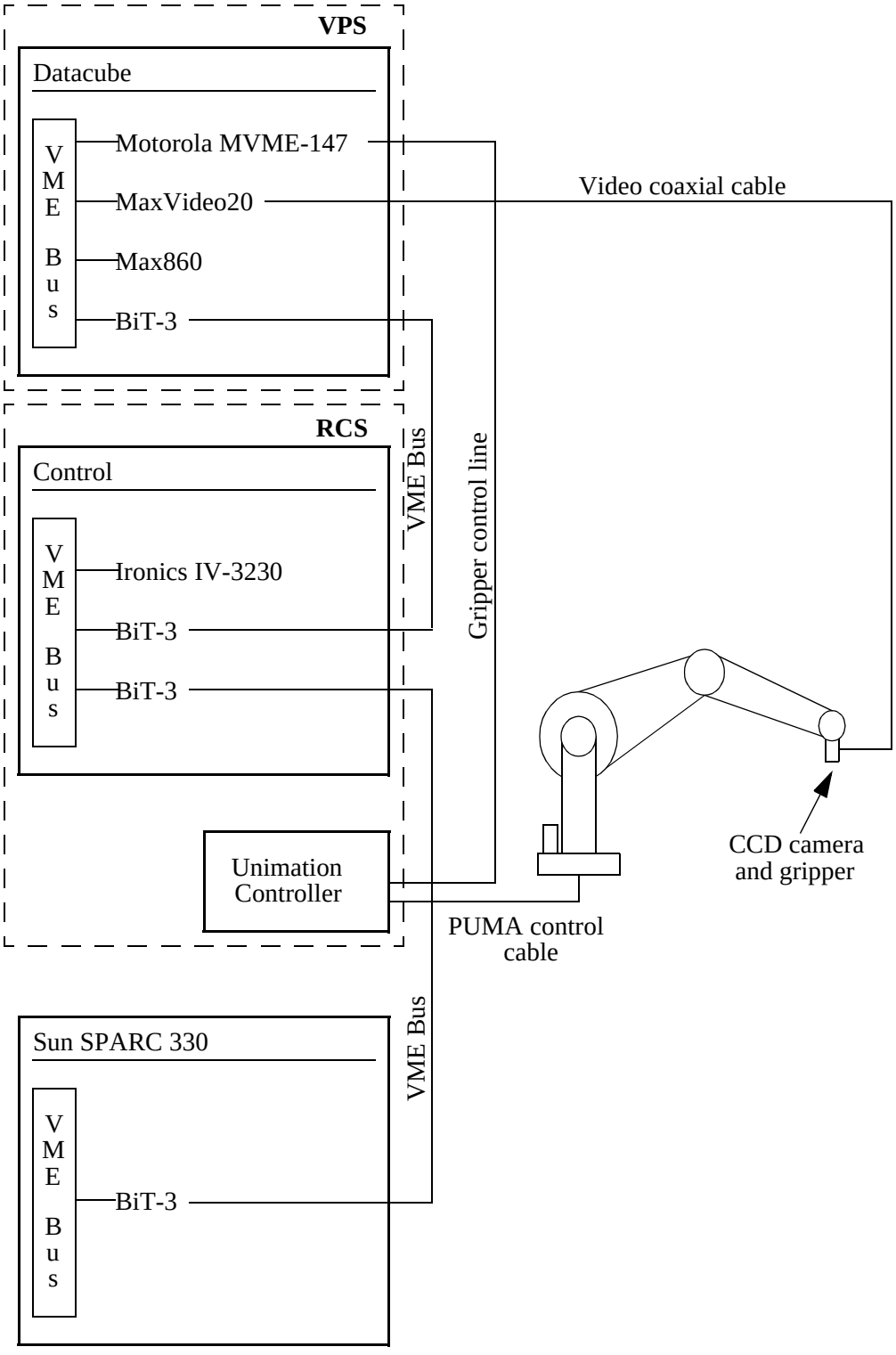

Figure 2: MRVT System Architecture

object centroid aligned with the optical axis ($?x = 0$ and $?y = 0$), the graspable dimension aligned with the gripper ($?? = 0$), and with an object depth of 581.7 mm.

We identified three basic failure categories that correspond to prior observed system behavior. The categories are 1) bad feature point selection/reselection; 2) loss of tracking for one or more feature points; and 3) feature point passing out of the view of the camera. By using the term features, we imply corner points or areas with significant intensity variations. Bad selection/reselection of a feature means that the selected feature may belong to an almost uniform intensity area or that the feature may belong to an edge (in other words, the selection mechanism failed to capture the characteristics of the feature).

4.2 Initial Experimental Run

Our initial experiment run consisted of 64 random object placement experiments (the object had the following dimensions: 14cm(L) x 6cm(W) x 8cm(H)). Each of the 64 experiments had a random variation in all four dimensional variables of interest. The results of the 64 runs are shown in Table 1. The results of these experiments uncovered a systemic error in the feature selection/ reselection algorithm. When the depth of the object was less than a specific amount during the selection of fine features or when the features were severely distorted during reselection, the algorithm would incorrectly chose a degenerate (untrackable) feature. It should be pointed out that some of the “lost track” failures could also be due to a poor feature selection or reselection. Since overall performance was inadequate and a known problem existed, another run of random object placement experiments to determine overall system performance was required after correcting the problem in the selection algorithm.

Table 1: Outcomes, Experimental Run 1

Result	Number of instances out of 64
Success	43
Bad select	16
Lost track	5
Out of view	0

4.3 System Performance Experimental Run

A second experimental run of 100 random object placement experiments was conducted. The range of variation of some of the dimensional variables was expanded for the second run since the failure rate of experiments where the confirmed reason was not the systemic error was exceptionally small. Table 2 shows the failure categories and the frequency of occurrence over the 100 random runs.

Table 2: Outcomes, Experimental Run 2

Result	Number of instances out of 100
Success	96
Bad select	1
Lost track	3
Out of view	0

The data from a typical random object placement experiment is shown in Figure 4 and Figure 5 (each cycle corresponds to 28ms). The plot for the Z-axis translation (Figure 5) has been annotated to show the moment when the gripper is closed and the manipulator withdraws along the Z-axis. We do not include X and Y rotation plots since they are almost zero during the grasping experiments.

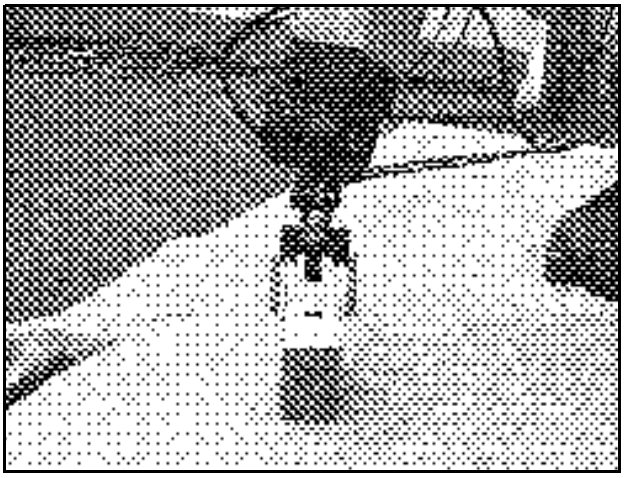

Figure 3: Oblique View at Grasping

One may argue that the system is too slow. We think that the system is slow but indicates the feasibility of vision-based grasping in uncalibrated environments and under various environmental conditions. The system spends a significant amount of time implementing the adaptive control algorithm in order to compensate for the uncertainty of environmental parameters like depth. The system also has to evaluate and re-select features continuously. We are currently in the process of increasing the computational power of our system in order to improve the performance.

4.4 Moving Object Experimental Run

Considering the success of the static object grasping after identifying problems and incorpo-

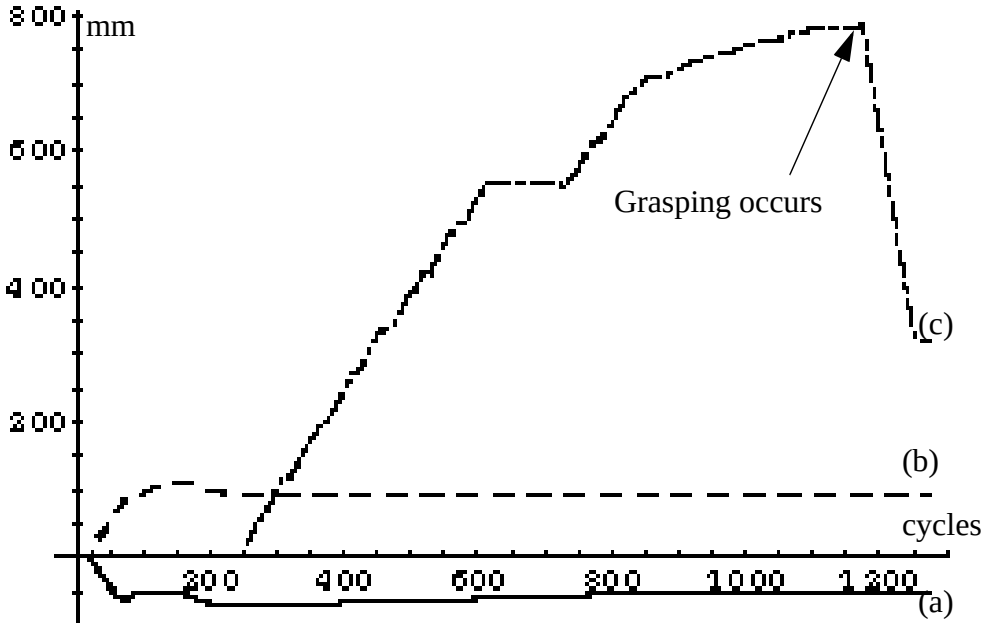

Figure 4: X (a), Y (b), and Z (c) Translation

rating changes, we applied the revised grasping system to the problem of moving target grasping. A limited run of 10 experiments was conducted to examine the suitability of the changes to the moving grasping problem. These experiments used the same method as the previous static runs, with the initial object placement was randomized over the four dimensional variables. Table 3 shows the failure categories and their frequency of occurrence over the 10 trials. These prelimi-

nary results are encouraging, considering that the same control law, without a disturbances term, was used.

Table 3: Outcomes, Moving Experimental Run

Result	Number of instances out of 10
Success	8
Bad select	0
Lost track	2
Out of view	0

5 Conclusion

In this paper, we have presented important issues related to the system design using a vision-based approach to robotic grasping. We have framed the issues with the trade-offs that are related to each one of these issues. In addition, we have presented prior research efforts in this area and detailed how the various issues and trade-offs were treated by each of these previous efforts.

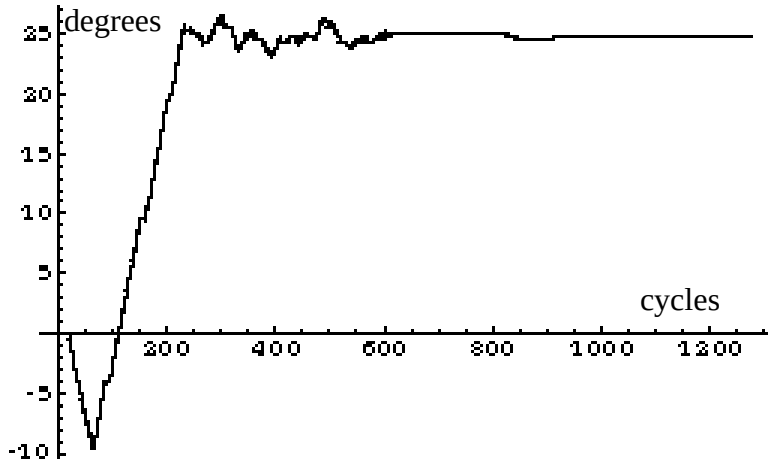

Figure 5: Z-axis Rotation

Our own approach to these issues has been presented, including the rationale behind the decisions that were made when we were designing our grasping system. We also present the results from three sets of random object placement experiments that emphasize failure analysis in order to improve the system's performance. After the first set of random experiments, we identified and fixed an intermittent, systemic error in our feature selection/reselection scheme. The second run demonstrates the efficacy of our approach and meets our initial requirement of an acceptable success rate.

The same system achieved promising results when applied to the grasping of a moving object, with a preliminary run of random object placement experiments showing an 80% success rate.

6 Acknowledgments

This work has been supported by the Department of Energy (Sandia National Laboratories) through Contracts #AC-3752D and #AL-3021, and the National Science Foundation through Contracts #IRI-9410003 and #IRI-9502245.

7 References

[1] Allen, P., Timcenko, A., Yoshimi, B., and Michelman, P. (1993) "Automated tracking and grasping of a moving object with a robotic hand-eye system," *IEEE Transactions on Robotics and Automation*, Vol. 9, No.2, pp. 152-165.

[2] Brandt, S., Smith, C., and Papanikolopoulos, N.P. (1994) "The Minnesota Robotic Visual Tracker: A Flexible Testbed for Vision-Guided Robotic Research," *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1363-1368.

[3] Houshangi, N. (1990) "Control of a robotic manipulator to grasp a moving target using vision," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 604-609.

[4] Kimura, H., Mukai, N., and Slotine, J. (1992) "Adaptive visual tracking and Gaussian network algorithms for robotic catching," *Winter Annual Meeting of the American Society of Mechanical Engineers*, pp. 43:67-74.

[5] Rizzi, A. and Koditschek, D. (1993) "Further progress in robot juggling: the spatial two-juggle," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 919-924.

[6] Sakaguchi, T., Fujita, M., Watanabe, H., and Miyazaki, F. 1993 "Motion planning and control for a robot performer," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 925-931, 1993.

[7] Smith, C. and Papanikolopoulos, N.P. (1994) "Computation of shape through controlled

active exploration," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2516-2521.

[8] Smith, C. and Papanikolopoulos, N.P. (1995a) "Grasping of static and moving objects using a vision-based control approach," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 329-334.

[9] Smith, C. and Papanikolopoulos, N.P. (1995b) "Theory and Experiments in Vision-Based Grasping," *Proceedings of the 34th IEEE Conference on Decision and Control*, pp. 4053-4058.