CLOSED-LOOP VISUAL GRASPING AND MANIPULATION

Billibon H. Yoshimi and Peter Allen
Department of Computer Science, Columbia University *

Abstract

Sensors play an important role in providing feedback to robotic control systems. Vision can be an effective sensing modality due to its speed, low cost, and flexibility. It can also serve as an external sensor that can provide control information for devices that lack internal sensing. Many robot grippers do not have internal sensors. These grippers can be prone to error as they operate under open loop control and usually require a precise model of the environment to be effective. This paper describes 2 visual control primitives that can be used to provide position control for a sensorless robot system. The primitives use a simple visual tracking and correspondence scheme to provide real-time feedback control in the presence of imprecise camera calibrations. Experimental results are shown for the positioning task of locating, picking up, and inserting a bolt into a nut under visual control. Results are also presented for the visual control of a bolt tightening task.

1 Introduction

In most manufacturing tasks, it is necessary to have the ability to move parts together in useful configurations to make an assembly process more efficient. For example, we might require a robot to grasp a part off the assembly line and insert it into some fixture. In order to perform the grasping portion of this task, the robot must be able to locate the part, move its gripper to the vicinity of the part, locate the best grasping points to grasp the part, move the fingers of the gripper to those grasping points, and finally, verify that the grasp is sufficient for holding the part. Usually these kinds of tasks are performed by blind robots which use world coordinates, jigs and other devices to remove the need for the robot to recover where the objects are located. Systems built using this kind of robot control usually require large start-up costs in pre-production measurement, setup and testing. These systems also exhibit inflexible, brittle qualities. If a small design change is made in how the product is manufactured, the cost of replanning the robot assembly line which may include extensive retooling and rejigging and a total revision of the robot control strategy can be prohibitively expensive. The research we propose in this section addresses some of the problems associated with vision-based robot control. We will show that many eye-hand coordination skills can be thought of as sensory-control loops, where specialized reasoning has been embodied as a feedback or control path in the loop's construction. Our framework captures the essence of these eye-hand coordination skills in simple visual control primitives with codified reasoning. We will also describe new ways in which vision can be integrated with a robotic task to give the robot the necessary compliance to perform in cases where the robot has little preconceived knowledge about the environment or the objects to be acted upon. This research will concentrate on understanding how different visual control methods can be applied to grasping and manipulation problems. In particular, we will concentrate on three parts of a bolt grasping and tightening task: how to move the gripper to the vicinity of the bolt, how we can use vision to move the fingers to grasp the bolt, and how we can use vision to monitor the bolt tightening task. We will show that visual feedback gives many systems the compliance necessary to perform the task and the ability to gauge its performance.

Other researchers have built systems that use vision to control robot motion. Hollingshurst and Cipolla [5] have developed a system for positioning a gripper above an object in the environment using an affine stereo transform to estimate the object's position. They correct for the transform's errors by using a second control scheme which converts the relative error in image space into a real world positioning change. Castano and Hutchinson [2] use visual constraint planes to create compliant surfaces for constraint robot movement in the real world. Both Hager et al. [4] and Feddema et al. [3] have used Image Jacobian-based control to perform various positioning tasks. Sharma et al. [8] use perceptual 3D surfaces to represent the workspace of the gripper and object and they plan their positioning tasks along these surfaces. Other systems which have used visual control include [10, 7, 6, 9, 13, 1]. Our system is specifically aimed at merging vision with grasping by providing visual control of position and contact using sensorless robotic fingers and hands. Details on this research can be found in [12].

2 Experimental system

The system pictured in figure 1 shows the major components of the system: a robot, a gripper, and two fixed stereo cameras. The gripper we used is a Toshiba FMA gripper (see figure 3). The gripper is comprised

*This work was supported in part by DARPA contract DACA-76-92-C-0007, an ONR MURI Grant, NSF grants CDA-90-24735 and IRI-93-11877, and Toshiba Corporation

of 4 fingers where each finger is made up of 3 stretchable, flexible chambers [11]. The gripper could be positioned in a variety of different poses by changing the pressure in each of the finger chambers. The original design used binary pressure valves which limited the movement of each finger to only eight possible configurations in a 3 degrees of freedom workspace. In order to perform precise manipulation, we needed greater finger positioning flexibility. We removed the eight position constraint from two fingers by adding 6 continuous pneumatic servo valves. Each valve took an input voltage and converted it into a pressure output. By changing the voltage, we could change the pressure inside each finger chamber. These voltages were generated by a D/A board attached to the central computer.

There are three problems with the gripper which make it difficult to use. First, the gripper system is totally devoid of sensors. This is a common problem shared by many other grippers. The variation in a finger's dimensions as it changes positions makes it very difficult to attach sensors to the fingers. One method which can be used to overcome this problem is the use of external sensors to provide feedback for closed loop control. We use vision to control the positioning of the gripper's fingers and to estimate when they have made contact with other objects in the environment. Second, the control space of the fingers is non-linear. Changes in pressure to a chamber do not necessary correspond to a equivalent linear change in finger position. In a later section, we will describe how we used visual control to compensate for this problem. And finally, we need to compensate for minor occlusions, a problem common in most grasping and finger-scale manipulations. To solve this problem, we have added additional structures to each fingers to move the fiducial marks associated with that finger to positions where they were not occluded by the object.

3 Object tracking algorithm

In order to quickly develop a working experimental system, and thus gain a better understanding of the limits of integrating vision and control, we demanded that the vision system fulfill two requirements. First, it must have the capability of tracking multiple objects. For our experiments, it was necessary to track the position of the gripper, its pose, and the positions of other objects in the environment. And second, the position information should arrive at a high rate with as little delay as possible. Real time control requires low latency.

The vision system we have developed satisfies these constraints. We are able to track multiple moving targets at real time rates. In our system, each moving target is a fiducial mark (a black dot on a white background) which can be attached to a robot or other objects in the environment. The upper bound of the number of targets trackable at any one time is 255. Each finger of the robotic hand has 4 fiducial marks, and each object to be manipulated also has 1 or 2 fiducial marks. The tracker uses intensity thresholds to segment the fiducial marks from the background.

Once a feature has been identified in each camera, we used stereo back-projection to determine its 3D position in space. The accuracy of this procedure is highly dependent on the accuracy of the camera calibrations. In our system, we perform a simple least squares calibration on each camera. Correspondence between features in each camera is determined initially by the user and the tracking maintains these correspondences at each sample interval.

4 Move-to-3D-position

The basic problem in visual control is to relate an image space signal to a corresponding control movement. In our case, the robot's position in the real world, $\mathcal{W}$, is known and controlled by us. What is unknown is the relationship between the position error observed by the vision system and the corresponding control movement in $\mathcal{W}$. In our system, we compute the error in position between a known point on the robot, $P1$, and a goal point in the workspace, $P2$, as

$$\vec{e} = \vec{P2}' - \vec{P1}' \qquad (1)$$

$P1'$ and $P2'$ are the back-projected locations of the two points which have been imaged by the stereo cameras. Hence, we are defining a relative error computed from the vision system that can be used to position the robot. Since we are dealing with point features only, we restrict our analysis to translational degrees of freedom; we assume the gripper is in a fixed pose.

Given this error vector, which is expressed in 3D space, we can position the robot to the new position that will zero the error. However, the new position error will only be zero if the calibration is exact, which in our case it may not be. To alleviate this, we continually update this position error from the vision system. As we move the robot in the direction of the error vector, we can update the relative positions of the points $P1$ and $P2$ using the vision system. Small errors in positioning can therefore be compensated for in an on-line fashion. At each step of the process we are recomputing the relative position error and reducing it. As this is a simple position controller, it can suffer from the usual problems of oscillation and/or infinite convergence. In our case, we were able to control the position accurately enough using a variable position gain and a cut off threshold to prevent oscillation. The controller finishes when $\vec{e}$ falls below some critical threshold.

5 Guarded-move

The *guarded-move* primitive assumes that F, a point on the manipulator (e.g. a fiducial mark on a finger) and O, some point in the environment, are known. We know how to make F move to O (using the *move-to-3D-position* primitive described in section 4). If,

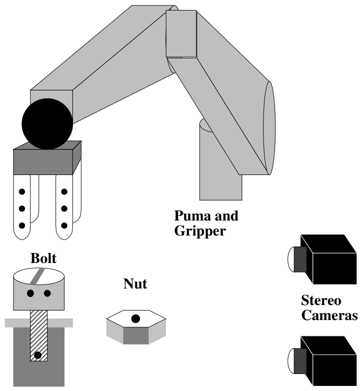

Figure 1: Experimental system used to test visual control of grasping and manipulation.

along its trajectory from F to O, the observed position of F does not change, we can apply Newton's first law of motion: *Every body persists in its state of rest or of uniform motion in a straight line unless it is compelled to change that state by forces impressed on it.* In the case of the *guarded-move* operation, if F is moved along a trajectory toward O and at some point F' along this trajectory its observed position stops changing, there exists two possible reasons for this stoppage. First, the manipulator has reached the limits of its workspace. But, as we have stated earlier, we assume the position O lies in the workspace of the manipulator and that the path from F to O also lies in the workspace of the manipulator. Hence only the second reason is possible, the manipulator must have made contact with some object in the environment which has impeded its movement along the given trajectory. This relationship can be generalized for a set of fingers, F_i, and a set of points associated with an object, O_i. Using this formulation, we can determine when a group of fingers has established contact with a given object.

Our implementation detects when contact has been established in the following manner. Each finger is commanded to close in upon one point in the center of the object. The controller monitors the movement of each finger feature, F_i'. As each finger makes contact with the object, it's corresponding feature point stops moving. Since the force caused by the closing action is being counteracted by some other force (otherwise the finger would move), we attribute the lack of movement to the object exerting a force against the finger. Using the continuous pneumatic control system we have developed, we were able to precisely control the movement of the gripper's fingers under visual control.

6 Positioning results

In figure 2, the robot was instructed to move to a position where the two fingers of the gripper surrounded the head of the bolt. In figure 3, the robot was in-

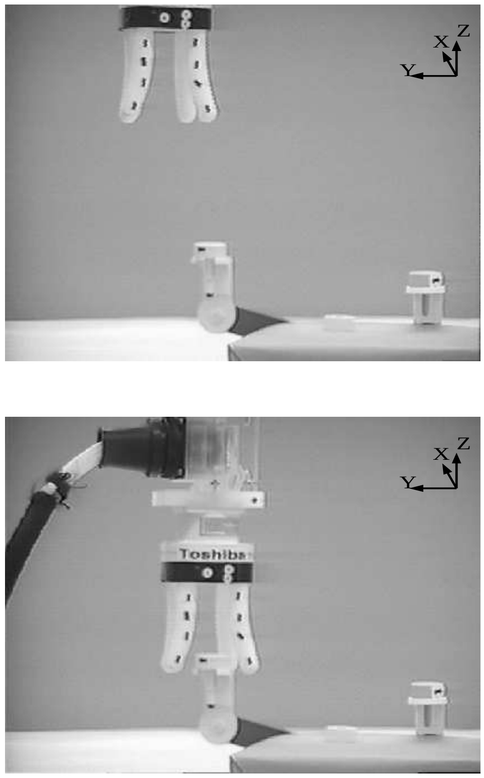

Figure 2: TOP IMAGE: Initial position of gripper and robot. BOTTOM IMAGE: Gripper and robot position after finishing the approach task. In both images, the x axis points into the image.

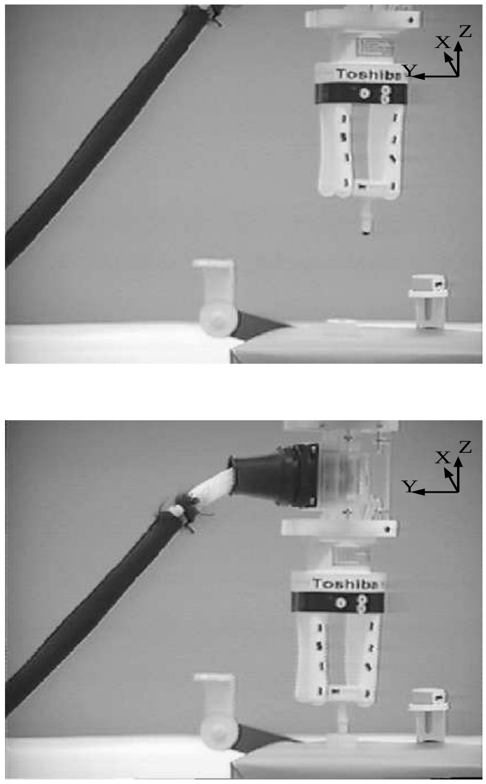

Figure 3: TOP IMAGE: Initial position of gripper and robot. BOTTOM IMAGE: Gripper and robot position after finishing the insert task. In both images, the x axis points into the image.

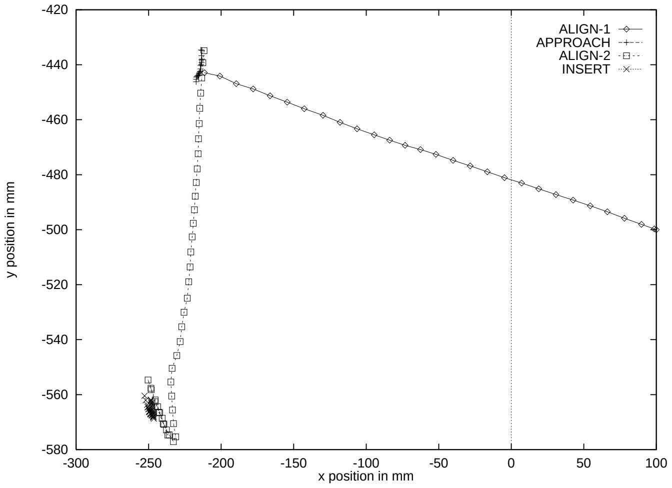

Figure 4: $(x,\ y)$ trajectory taken by the robot in $\mathcal{W}$ for the entire motion.

structed to insert the tip of the bolt into the nut. Using the positioning primitive described above, the robot was able to insert the tip of the bolt into the nut. The system was capable of inserting a peg with a 5mm diameter tip into a 9mm diameter hole.

7 Affect of calibration errors

Similar experiments were performed to test the effects of perturbing the camera calibration. In this experiment, we moved the right camera approximately 400mm from its calibrated position. The robot was then instructed to perform the same task described in the previous section. Figure 5 shows trajectory taken by the robot while performing the align-approach-grasp-extract-align-insert task. While the absolute positioning errors are large, the relative object positioning was sufficient for both grasping and for insertion. The accuracy of our experimental system depends on many factors. We rely on the camera's ability to pin-point the location of objects in the environment and the ability of the camera calibration to determine the 3D position of the objects in the environment. Noise in the image formation process can cause an object which appears in one location to change position in adjacent sampling periods. When passed through the 2D-3D image reconstruction process, this error is magnified. We have tried to limit the effect of these problems by reducing the size of the fiducial marks (decreasing the possible chances for error in spatial localization) and decreasing the speed of the robot (decreasing the error caused by temporal aliasing).

The effects of calibration are noticeable. The trajectory taken by the robot in performing the alignment task is not straight. The curved path demonstrates how relative positioning can be used to correct for calibration error. These are the same errors which appear

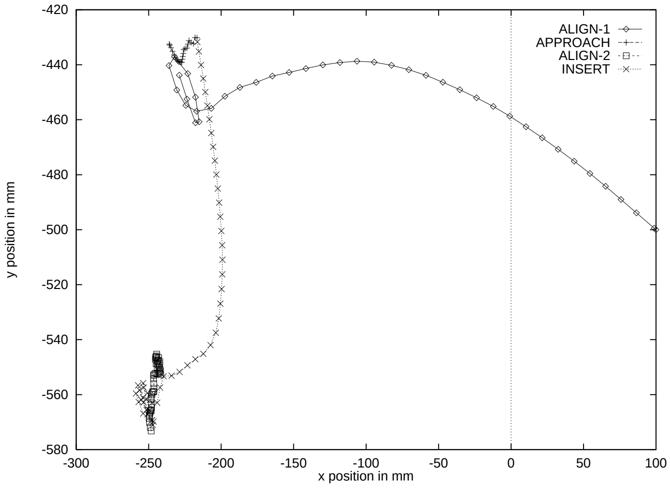

Figure 5: (x, y) trajectory followed by robot in the real world as it performs the align-approach-grasp-extract-align-insert task. In this experiment, the right camera has been moved approximately 400mm from its calibrated position.

to have affected the robot as it neared goal positions. In both approach tasks, the robot position oscillates noticeably near the end of the movement. One way of solving these problems is by adding integral and/or differential terms to the control. In spite of these calibration errors, the system was still able to function correctly albeit sub-optimally.

8 Two-fingered grasping

In this experiment, we examine how vision-based robot control primitives can be coordinated to perform a more complicated task, a task involving a two fingered gripper. The experiments were conducted using the experimental setup pictured in figure 6. The goal of the visual control system is to close the fingers of the gripper around the bolt: grasping it. We assume that the nut is securely attached to the environment. We assume that the bolt has, to some degree, been started in the nut (we are not solving the initial bolt-insertion and threading task). At the start of the experiment, the fingers do not support the bolt. We assume that at the onset of the experiment, the gripper system has been placed in a position over the head of the bolt and that the head of the bolt lies within the workspace of the gripper. This assumption is reasonable since, in an earlier experiment, we have shown that coarse positioning is possible using visual control. The grasping algorithm is:

1. The left finger is ordered to make a *guarded-move* to the center of the bolt.

2. The right finger is ordered to make a *guarded-move* to the center of the bolt.

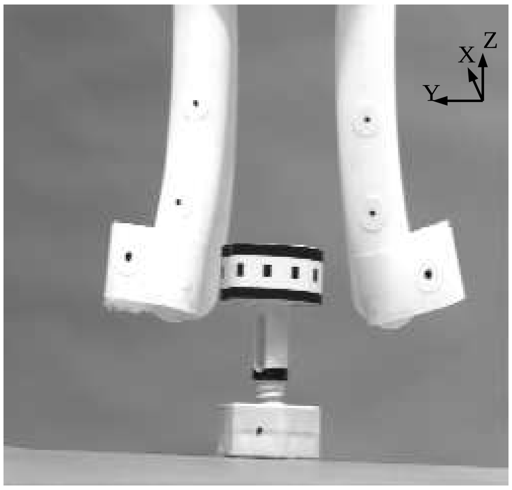

Figure 6: Experimental setup showing the position of the fingers at the start of the two-fingered grasp experiment. Both fingers are not in contact with the bolt.

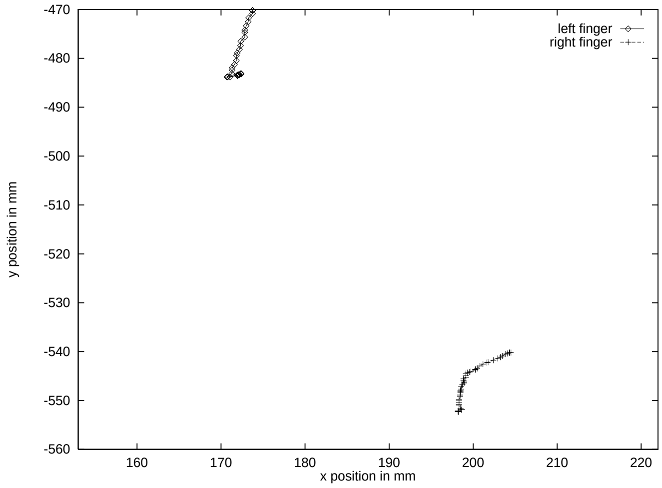

Figure 7: x-y of trajectories taken by both fingers in $\mathcal{W}$ for the grasping experiment. (z is ignored)

8.1 Experimental results

Figure 6 shows the initial position of the fingers before the experiment was executed. The initial position of the fingers is open with the left finger positioned slightly more toward the cameras and the right finger positioned slightly more away from the cameras. This configuration was chosen to satisfy one of the constraints for the tighten-bolt task described in the following section. The fiducial marks used for feedback in this experiment are the marks closest to the tips of each finger (mounted on an extended structure attached to each finger tip).

Figure 7 is a trace of the x-y positions in $\mathcal{W}$ moved to by the fingers during the experiment (z is ignored). In both cases, the fingers move along a fairly smooth trajectory toward the bolt center before terminating with the guard condition. The smooth trajectory of both fingers is more obvious in figure 8. In this figure, we display the y positions of each finger as a function of time. From time 1-24, the left finger closes in upon the head of the bolt. From time 25-69, the right finger closes in upon the head of the bolt. Only the left finger moves during the left finger closing stage. And once, the left finger has stopped moving (the guard condition being satisfied), the left finger ends it guarded move and the right finger starts moving. These results reinforce the idea that vision can be used to monitor the progress of a task and that vision can be used to determine when to switch between tasks.

After extensive testing of the grasping algorithm, we noticed that the halting criterion for the *guarded-move* also allows the gripper to grasp the target object with some force before the primitive is terminated. Since the terminating condition requires at least 5 samples where no movement is observed, the additional displacement not observed manifests itself as a force against the object. In the above scenario, this

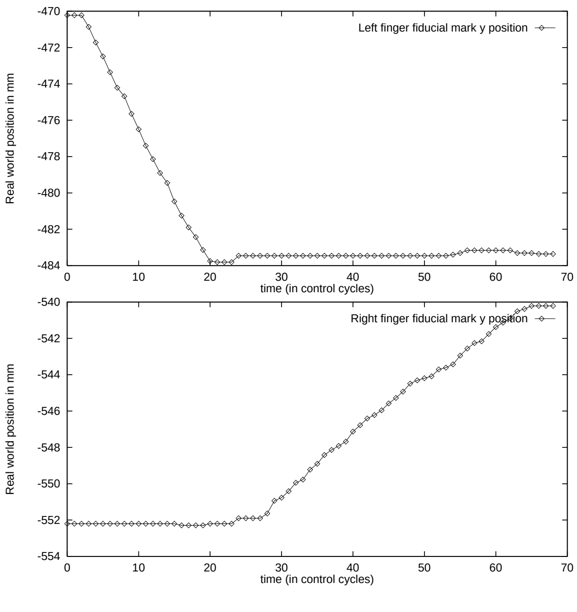

Figure 8: Plot of y position in $\mathcal{W}$ for both left and right fingers during the grasping process. The horizontal axis is time in .2 second increments (5 Hz update rate). The vertical axis is position in mm. Time 1-24 shows the left finger closing. Time 25-69 shows the right finger closing. Each move is cancelled when the *guarded-move* detects no further progress being made by the finger in question.

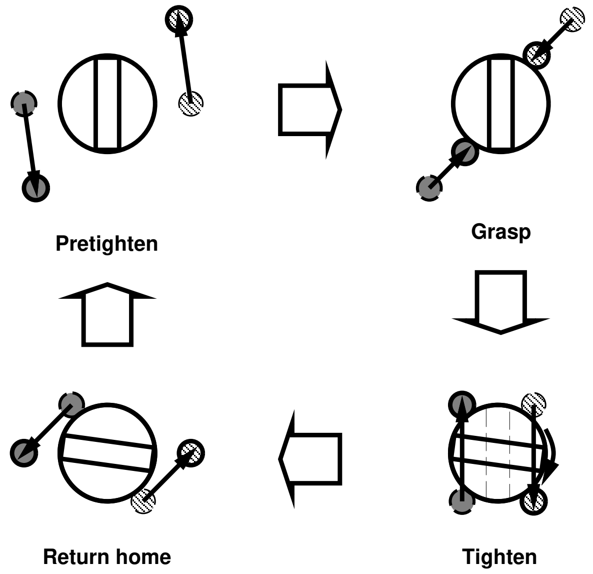

Figure 9: Finite state representation of the bolt tightening tasks. Two smaller circles represent fingers. The large circle with the rectangle through the center represents the bolt head. Each node is an overhead view of the movement and position of the bolt and fingers during one part of the bolt tightening process.

provided us with the necessary normal forces necessary for manipulating light objects ($\leq$ 50grams.)

9 Two-finger bolt tightening

In addition to developing the two-fingered grasp task, we have also developed one additional multi-finger task: bolt tightening. The goal of the bolt tightening task is to take a grasped object (a bolt) and rotate the object while maintaining contact with the object during the rotation. Like the grasping task, tasks similar to the bolt tightening task can be found in almost all manufacturing environments. Where the grasping task required that each finger perform only one movement and the movements were conducted in a serial fashion, the bolt tightening task requires that the fingers move synchronously when performing the task. Also, the bolt tightening task requires that the system compute the new intermediate configurations between the start and goal states of the rotated object. The system we present will demonstrate that visual control is capable of handling grippers which require the coordination and monitoring of multiple fingers.

9.1 Experimental method

After analyzing the bolt tightening task, we have decomposed the task into four, individual subtasks. As shown in figure 9, the four subtasks (corresponding to four states in a finite state machine) are move to pretighten position, grasp bolt, tighten bolt, and return to home state. We have devised a visual control strategy which uses this finite state diagram to control the bolt tightening task. A key part of the development of the bolt tightening task was in recognizing how the previously developed tasks and primitive operations could be used to perform these operations. In order to make the bolt tightening experiment tractable, we constrained the problem in the following ways: 1) pretighten and post-tighten positions are known in advance (these positions can be easily computed though), 2) bolt head is in the workspace of the gripper and 3) all fiducial marks are trackable throughout the the entire experiment.

9.2 Experimental results

Figure 6 shows the position of the gripper system at the start of the experiment. The fingers have been moved to the pretighten position. The fiducial marks used for feedback in this experiment are the marks closest to the tips of each finger (mounted on an extended structure attached to each finger tip).

After calling the grasp procedure, figure 10 shows the end position of the fingers after both guarded moves finished. The results of this grasping operation are very similar to the results presented in the previous section on grasping.

Figure 11 shows the position of the fingers after the tightening operation. While not obvious from the static picture, the fingers actually traveled in an arc around the rotational axis of the bolt. This footage is available in video form. This is also clearly shown in figure 13. Figure 13 shows the position where the fingers moved when the tighten task terminated. It is an overhead shot showing the x and y positions of the points in $\mathcal{W}$ ignoring the z-coordinate values. This graph clearly shows the arc-like 3D trajectory taken by each fiducial mark as the gripper is ordered to tighten. This trajectory is resulted from the finger complying around the head of the bolt as it tried to satisfy the guarded move condition. Since the torques exerted by both fingers was sufficient to overcome the bolt's stiction and jamming forces, it was possible for the gripper to turn the bolt. If, on the other hand, the movement vector generated by the guarded move did not provide sufficient torque, the bolt would not have moved. It should also be noted that if the fingers did not comply around the head of the bolt, the tightening operation would not have been possible.

Figure 12 shows the final position of the fingers after executing the return to home part of the task. The fingers were ordered to move to the post-tighten position and followed by a movement back to their initial starting position, the pretighten position. We present this figure for completeness to demonstrate the complete cycle of positions required for the bolt tightening operation.

Figure 14 shows how monitoring the feature associated with the bolt can be used to determine the status of the tightening operation. In addition to the movement of the finger, this figure also displays the

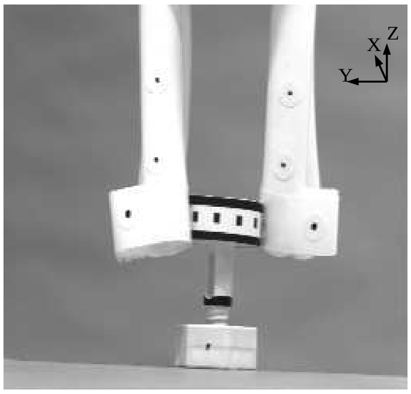

Figure 10: Position of fingers after grasping portion of bolt tightening experiment.

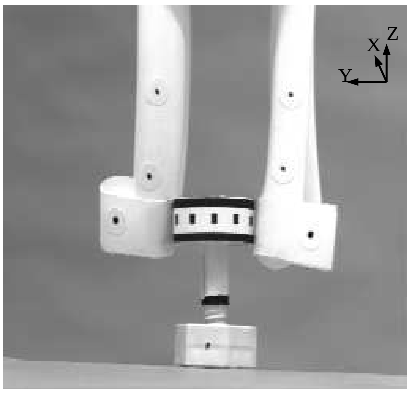

Figure 11: Position of fingers after finishing the tightening operation.

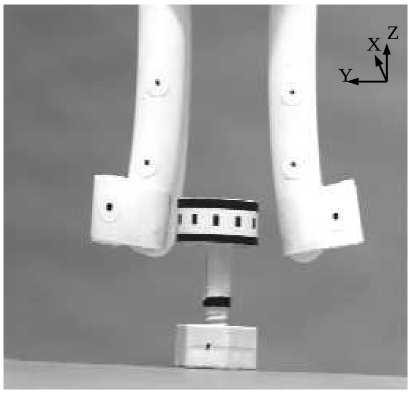

Figure 12: The end position of the fingers after returning to home state.

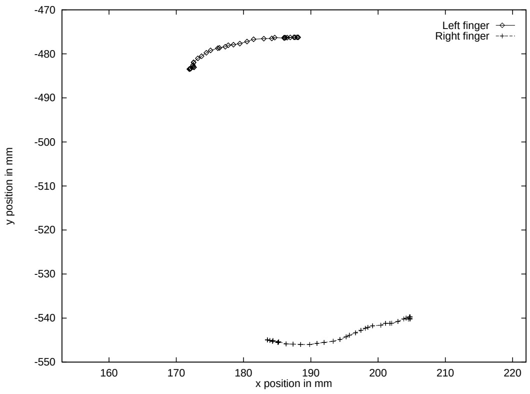

Figure 13: x-y plot of 3D finger positions during the tightening phase of the experiment. (z is ignored)

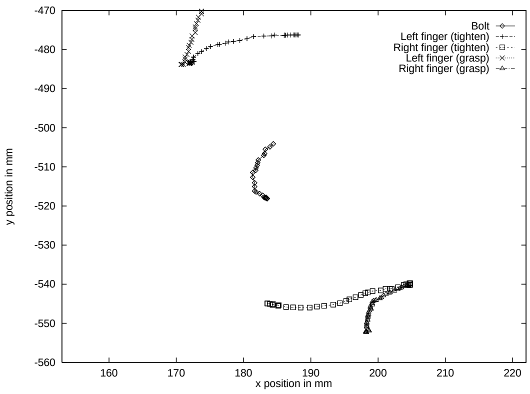

Figure 14: Combined x-y plot for grasping and tightening experiment including the position of a fiducial mark on the bolt head (overhead view).

x-y movement of a fiducial mark on the head of the bolt. This result indicates that vision is also capable of monitoring the status of the tightening task. By observing the movement of bolt features, we can verify that the bolt head actually moved during a bolt tighten operation. Without this information it would be difficult to determine if progress had been made in an assembly task. In many bolt tightening scenarios, it is possible for the bolt to jam during the tightening phase. In these cases, it is very important to detect if and when jamming occurs. By detecting jamming early in the tightening cycle, it is possible prevent potentially costly repairs by taking remedial measures to cure the jamming condition. The potential jam condition could then be relayed back to a higher level process to prompt that task to take an alternative strategy to the current operation. We have not implemented such feedback but note that such a strategy is possible.

10 Conclusions

The key to our solution lies in the development of primitive visual control operations. The primitive visual control operations allowed us to examine many of the real time problems associated with visual control in concise, manageable units. By decomposing a complex manipulation into a series of these operations, we removed much of the complexity associated with creating a visual control system. Also, by developing the system in a modular fashion, we were able to reuse the primitive operations to solve a more complex task. We have also shown that such methods are readily adapted to different robot systems. For our experiments, we applied visual control techniques to both a standard Puma robot and to a non-standard pneumatic actuator, the Toshiba gripper. The Toshiba gripper provided a unique challenge for our controller. The gripper was capable of performing many compliant manipulations in spite of the fact that it was devoid of any feedback sensors. While the rigid metal surfaces of the Puma robot could have easily been outfitted with a "skin" of pressure sensors, the fingers of the Toshiba gripper changed drastically during any operation thus increasing the difficulty of mounting and using many conventional pressure sensors. By implementing a vision-based robot control solution to this problem, we have avoided this problem. Not only was it not necessary to add sensor to the fingers, but the results have shown that visual control is capable of performing many of the operations necessary for elementary manipulation. Future extensions to this work include the examination of the occlusion problem, how to detect slippage and how to estimate contact forces visually.

References

[1] P. Allen, A. Timcenko, B. Yoshimi, and P. Michelman. Automated tracking and grasping of a moving object with a robotic hand-eye system. *IEEE Trans. on Robotics and Automation*, 9(2):152–165, 1993.

[2] A. Castano and S. Hutchinson. Visual compliance: Task-directed visual servo control. *IEEE Trans. on Robotics and Automation*, 10(3):334–342, June 1994.

[3] J. Feddema and C. S. G. Lee. Adaptive image feature prediction and control for visual tracking with a hand-eye coordinated camera. *IEEE Transactions on Systems, Man and Cybernetics*, 20:1172–1183, Sept./Oct. 1990.

[4] G. Hager, W. Chang, and A. Morse. Robot feedback control based on stereo vision: Towards calibration-free hand-eye coordination. In *Proc. IEEE Conf. on Robotics and Automation*, volume 4, pages 2850–2856, 1994.

[5] N. Hollinghurst and R. Cipolla. Uncalibrated stereo hand-eye coordination. Technical Report CUED/F-INFENG/TR126, Department of Engineering, University of Cambridge, 1993.

[6] A. Koivo and N. Houshangi. Real-time vision feedback for servoing robotic manipulator with self-tuning controller. *IEEE Transactions on System, Man, and Cybernetics*, 21, No. 1:134–142, Feb. 1991.

[7] N. Papanikolopoulos, B. Nelson, and P. Khosla. Six degree-of-freedom hand/eye visual tracking with uncertain parameters. In *Proc. of IEEE International Conference on Robotics and Automation*, pages 174–179, May 1994.

[8] R. Sharma, J. Herve, and P. Cucka. Analysis of dynamic hand positioning tasks using visual feedback. Technical Report CAR-TR-574, Center for Auto. Res., University of Maryland, 1991.

[9] S. Skaar, W. Brockman, and R. Hanson. Camera-space manipulation. *International Journal of Robotics Research*, 6(4):20–32, Winter 1987.

[10] T. M. Sobh and R. Bajcsy. Autonomous observation under uncertainty. In *Proc. of IEEE International Conference on Robotics and Automation*, pages 1792–1798, May 1992.

[11] K. Suzumori, S. Iikura, and H. Tanaka. Development of a flexible microactuator and its application to robotic mechanisms. In *IEEE International Conference of Robotics and Automation*, pages 1622–1627, April 1991.

[12] B. H. Yoshimi. *Visual Control of Robotics Tasks*. PhD thesis, Department of Computer Science, Columbia University, May 1995.

[13] B. H. Yoshimi and P. Allen. Alignment using an uncalibrated camera system. In *Proceedings: Image Understanding Workshop*, pages 411–420, 1993.