Proc. Thirty-Third Annual Allerton Conf. on Communication, Control, and Computing, University of Illinois at Urbana-Champaign, Oct. 4-6, 1995

Visually Servoed Manipulation Using An Active Camera

Bradley J. Nelson

Dept. of Mechanical Engineering (M/C 251)
University of Illinois at Chicago
842 West Taylor Street
Chicago, Illinois 60607-7022

Pradeep K. Khosla

Dept. of Electrical and Computer Engineering
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, Pennsylvania 15213-3890

Abstract

Visual servoing is a robust technique for aligning both static and moving parts using imprecisely calibrated camera-lens-manipulator systems. An important limitation of these systems is the workspace within which the alignment task can be successfully performed due to the position and orientation of the camera. An active camera can extend this region, however this changes the visual representation of the task itself. Therefore, the reference input that drives the visually servoed manipulator must change appropriately. In this paper, a framework that allows for camera-lens motion during visually servoed manipulation is described. The main components of the framework include object schemas and port-based agents. Object schemas represent the task internally in terms of geometric models with attached sensor mappings. Object schemas are dynamically updated by sensor feedback, and thus provide an ability to perform three dimensional spatial reasoning during task execution, a capability traditional image-based visual servoing lacks. Object schemas are also able to dynamically create desired visual representations of the task from which reference inputs for vision-based control strategies are derived. The sensor mappings of object schemas are also used to guide camera motion based on task characteristics. Port-based agents are the executors of the visual reference inputs and the camera motion commands. They interact with the real world through visual servoing control laws. Experimental results that demonstrate system capabilities and performance are presented.

1. Introduction

Visual servoing is a robust technique for guiding a manipulator through an uncertain environment using poorly calibrated camera-lens-manipulator systems. The technique employs a differential view of the world in which only small scene changes between image frames are assumed and compensated for by any of a number of proposed control techniques. Most proposed visual servoing control strategies use image-based techniques [13], because these techniques avoid the need to perform an explicit inverse perspective mapping. In its basic form, image-based visual servoing is a purely reactive manipulator motion strategy. The objective of the control system is simply to make some set of measured feature states defined in the camera's sensor space match some set of desired states defined in the same space by moving the manipulator.

An important limitation to which all visual servoing strategies are subject, is the limited spatial resolution, depth-of-field, and field-of-view inherent to camera-lens systems. Visual servoing systems fail when objects being visually tracked exceed the depth-of-field and field-of-view limitations imposed by vision sensors. These systems also fail when the visual feedback provided by a camera-lens system provides insufficient spatial resolution for visually guiding the manipulation **task.** A dynamically reconfigurable camera-lens system is able to extend valid tracking regions of objects, however camera-lens motion changes the visual representation of the task. Therefore, image-based reference inputs that drive the visually servoed manipulator performing the task must change as sensor parameters change.

In this paper, we describe and experimentally verify a framework for visually servoed manipulation that integrates visually servoed manipulation with an active camera-lens system via a **3D** task representation. This framework provides appropriate image-based reference inputs to a manipulator performing a visually servoed manipulation task during camera-lens motion. Another benefit of the framework is its ability to provide a reasoning capability for geometric motion planners. Within the framework, the task is represented by the actions and interactions of dynamic **3D** geometric models of objects in the environment. Each model is augmented by sensor mappings representing the visual sensors used to provide feedback for task execution. These augmented environment models are referred to **as *object schemas,*** because they simultaneously represent the task and provide an understanding of how the environment is perceived [8]. The mappings are used in two ways. First, they are used to predict how the sensors will view the environment model while the given task is being executed. This provides reference inputs for visual servoing controllers which then guide task execution. Object schemas also use their sensor mappings to determine appropriate motion for active camera-lens systems in order to avoid many of the limitations inherent to static camera-lens system. Object schemas drive both visually servoed manipulators and visually servoed active camera-lens systems. These two visual servoing systems are treated **as** independent, autonomous visual servoing systems and are referred to **as *port-based agents.***

We begin by discussing our task representation in terms **of** object schemas. Next, visually servoed port-based agents for object manipulation and dynamic sensor reconfiguration are presented. A description of the overall system framework that combines object schemas, port-based agents, and the real world then follows. Finally, we present experimental results which demonstrate the framework operating within a dynamically varying, imprecisely Calibrated environment.

2. A Task Representation Using Object Schemas

A key component of our system framework is the structure and content of the task representation. In order to provide the system with the capability to reason about manipulation strategies, the internal representation must contain three dimensional geometric knowledge of the objects being manipulated. The internal representation must also be capable of being correlated with visual information in order to direct the visually servoed manipulator to perform the task. This implies that a visual representation of the task must also exist internally.

The task representation is defined in terms of for objects. The object representation we call an ***object schema.*** Our object schema definition includes a geometric environment model of an object augmented by sensor mappings that describe how the object will be perceived by the actual system sensors. Forward projective sensor mappings and their associated Jacobian matrix for each visual sensor that exists in the system provides this representation. For a single camera these forward mappings are simply perspective projection mappings and are of the

form

$$x_S = \frac{fX_C}{s_x Z_C} \qquad y_S = \frac{fY_C}{s_y Z_C} \tag{1}$$

where x_S and y_S are the projected image coordinates of a point on the object in the internal representation located with respect to the camera frame at (X_C, Y_C, Z_C), f is the focal length of the camera lens, and s_x and s_y are the pixel dimensions of the CCD array. The Jacobian of this mapping is of the form

$$\delta \mathbf{x}_S = \mathbf{J}(\phi)\delta \mathbf{X}_T \tag{2}$$

where $\delta \mathbf{x}_S$ is an infinitesimal displacement vector in sensor space and $\delta \mathbf{X}_T$ is an infinitesimal displacement vector in task space. $\mathbf{J}(\phi)$ is the Jacobian matrix and is a function of the extrinsic and intrinsic parameters of the visual sensor as well as the number of features used for tracking the object and their locations on the image plane.

For the experimental results to be presented, an orthogonal stereo pair is used. Figure 1 shows the coordinate frame definitions for this type of camera-lens configuration. If the axes are aligned as shown in the figure, the Jacobian mapping from task space to sensor space for a single feature can be written as

$$\mathbf{J} = \begin{bmatrix} \frac{f}{s_x Z_{Cl}} & 0 & -\frac{x_{Sl}}{Z_{Cl}} & -\frac{x_{Sl}Y_T}{Z_{Cl}} & \left[\frac{fZ_T}{s_x Z_{Cl}} + \frac{x_{Sl}X_T}{Z_{Cl}}\right] & -\frac{fY_T}{s_x Z_{Cl}} \\ 0 & \frac{f}{s_y Z_{Cl}} & -\frac{y_{Sl}}{Z_{Cl}} & -\left[\frac{fZ_T}{s_y Z_{Cl}} + \frac{y_{Sl}Y_T}{Z_{Cl}}\right] & \frac{y_{Sl}X_T}{Z_{Cl}} & \frac{fX_T}{s_y Z_{Cl}} \\ \frac{x_{Sr}}{Z_{Cr}} & 0 & \frac{f}{s_x Z_{Cr}} & \frac{fY_T}{s_x Z_{Cr}} & \frac{x_{Sr}Z_T}{Z_{Cr}} - \frac{fX_T}{s_x Z_{Cr}} & \frac{x_{Sr}Y_T}{Z_{Cr}} \\ \frac{y_{Sr}}{Z_{Cr}} & \frac{f}{s_y Z_{Cr}} & 0 & -\frac{fZ_T}{s_y Z_{Cr}} & \frac{y_{Sr}Z_T}{Z_{Cr}} & \frac{y_{Sr}Y_T}{Z_{Cr}} - \frac{fX_T}{s_y Z_{Cl}} \end{bmatrix} \tag{3}$$

In (3), we assume the camera-lens parameters are identical for both cameras. The other terms in (3) correspond to Figure 1. The tracking of features will be described in Section 3.1.

Our previous work in defining the sensor placement measure ***resolvability*** [7] uses the Jacobian mapping to compare various visual sensor systems from a control standpoint in terms of the accuracy of control. This measure quantifies the ability of various monocular and stereo camera configurations to resolve the position and orientation of visually servoed

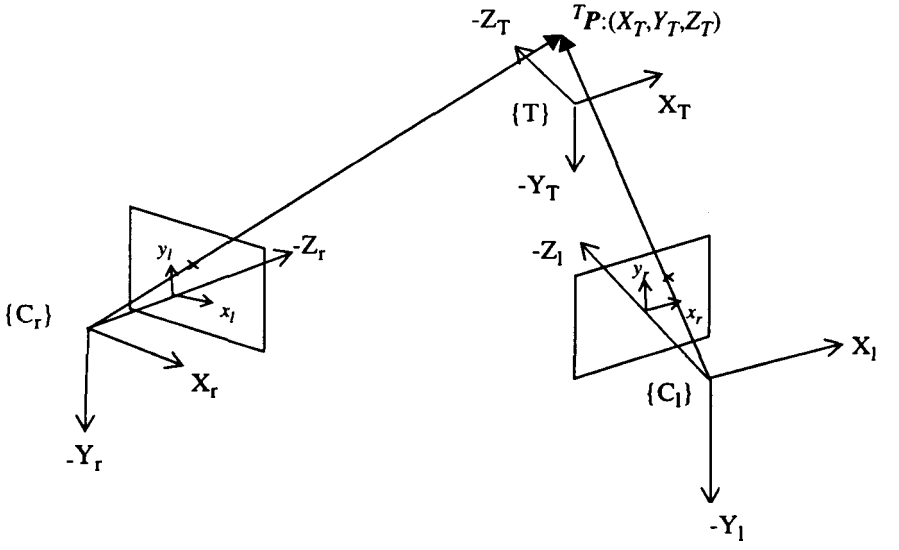

Figure 1. Task frame-camera frame definitions.

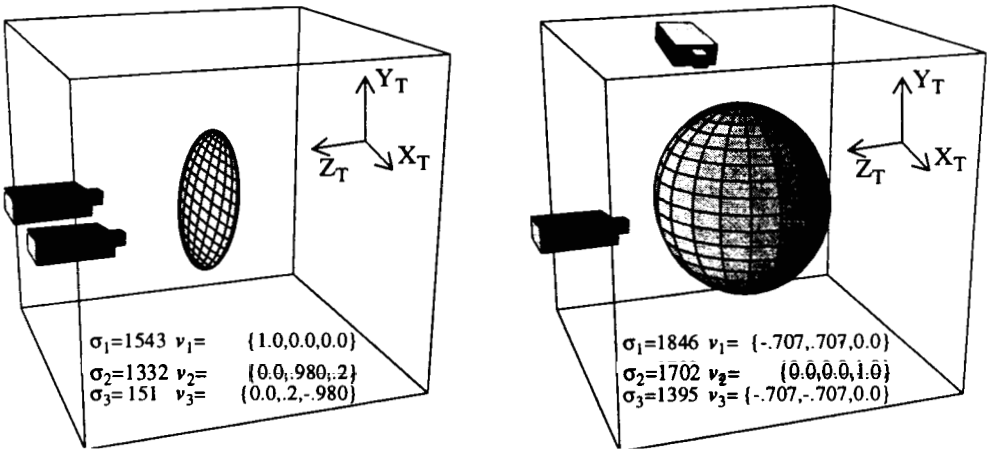

Figure 2. Resolvability ellipsoids for a stereo pair with parallel optical axes and a stereo pair with perpendicular optical axes. The singularvalues of J and the eigenvectors of $J^T J$ are given for each configuration.

objects. As discussed in [7], the measure is easily extendable to other visual sensors including multi-baseline stereo and laser rangefinders. A key component of this measure, the Jacobian mapping from task space to sensor space, is also a critical component of our visual servoing control strategy.

Figure 2 shows the resolvability of two different stereo camera configurations. A sensitivity analysis of resolvability with respect to the variable extrinsic and intrinsic camera-lens parameters shows how camera-lens systems can be reconfigured in order to improve the precision with which objects can be observed and manipulated [6].

Figure 3 shows a diagram of an object schema, including the current internal pose of the geometric model; its desired pose (for controllable schemas) as determined by a meta-schema (an assembly planner or supervisor); and sensor mappings used to direct variable sensor parameters as well as update the current internal pose of the schema based on actual sensor feedback. Note that the pseudoinverse of the Jacobian mapping used for determining sensor resolvability is also used to "servo" the geometric model in order to reduce errors between the current internal visual representation of the object and the actual visual representation.

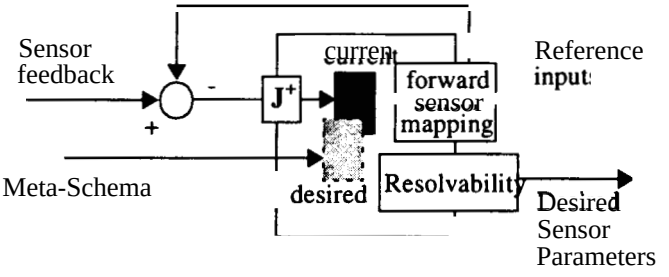

Figure 3. An object schema diagram.

3. Visually Servoed Port-Based Agents

3.1. A Manipulation Agent

The desired visual representation of a task is derived from object schemas. This desired visual representation provides reference inputs to a visual servoing system. The visual servoing system we consider to be an agent that continuously accepts reference commands from object schemas and attempts to achieve the desired visual representation of the task based on these inputs using visual servoing techniques.

Controlled active vision [9] is used to derive a control strategy for visually servoed port-

based agents. A state equation for the visual servoing system can be derived by discretizing **(2)** and writing this discretized equation as

$$\mathbf{x}(k+1) = \mathbf{x}(k) + T\mathbf{J}(k)\mathbf{u}(k) \tag{4}$$

where $\mathbf{x}(k) \in R^{2M}$ and is a vector of feature states, T is the sampling period of the vision system, $\mathbf{u}(k) = \begin{bmatrix} \dot{X}_T & \dot{Y}_T & \dot{Z}_T & \omega_{X_T} & \omega_{Y_T} & \omega_{Z_T} \end{bmatrix}$ is the commanded manipulator end-effector velocity, and M is the number of features being tracked. Using optimal control techniques, an objective function that places a cost on feature error and control energy of the form

$$F(k+1) = [\mathbf{x}(k+1) - \mathbf{x}_D(k+1)]^T \mathbf{Q} [\mathbf{x}(k+1) - \mathbf{x}_D(k+1)] + \mathbf{u}^T(k)\mathbf{L}\mathbf{u}(k) \tag{5}$$

can be minimized at each time instant to obtain the control law

$$\mathbf{u}(k) = -(T^2\mathbf{J}^T(k)\mathbf{Q}\mathbf{J}(k) + \mathbf{L})^{-1} T\mathbf{J}^T(k)\mathbf{Q}[\mathbf{x}(k) - \mathbf{x}_D(k+1)] \tag{6}$$

The vector $\mathbf{x}_D(k+1)$ represents the desired feature state, i.e. the desired visual representation of the scene, at the next time instant. $\mathbf{Q}$ and $\mathbf{L}$ are weighting matrices and allow the user to place a varying emphasis on the feature error and the control input. Extensions to this control strategy and guidelines for choosing the matrices $\mathbf{Q}$ and $\mathbf{L}$ can be found in [10]. New sensor configurations can be quickly and easily added to the system by substituting the correct $\mathbf{J}$ and by adjusting $\mathbf{Q}$ and $\mathbf{L}$ accordingly.

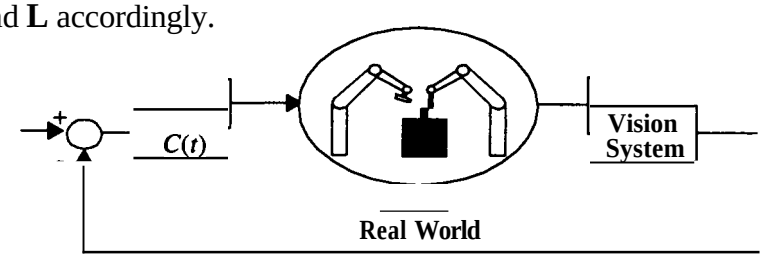

Figure 4. Visual servoing feedback loop.

The measurement of the motion of the features on the image plane must be done continuously and quickly. Our method for measuring this motion is based on optical flow techniques and is a modification of the method proposed in [1]. This technique is known as Sum-of-Squared-Differences (SSD) optical flow, and is based on the assumption that the intensities around a feature point remain constant as that point moves across the image plane. A more complete description of the algorithm and its original implementation can be found in [10].

3.2. A Dynamic Sensor Placement Agent

We have previously defined a dynamic sensor placement agent [6] by integrating sensor placement measures with a visual servoing control strategy. An objective function of the form

$$F(k+1) = [\mathbf{x}(k+1) - \mathbf{x}_D(k+1)]^T \mathbf{Q} [\mathbf{x}(k+1) - \mathbf{x}_D(k+1)] + \\ + \mathbf{u}^T(k)\mathbf{L}\mathbf{u}(k) + \frac{S}{w'(\mathbf{q}(k))} + \frac{U}{(foc)} + \frac{V}{\sigma_{min}} + W(fov) \tag{7}$$

is minimized at each time instant in order to derive a completely closed loop control law for the dynamic sensor placement agent of the form

$$\mathbf{u}(k) = -(\mathbf{J}^T(k)\mathbf{Q}\mathbf{J}(k) + \mathbf{L})^{-1}\mathbf{J}^T(k)\mathbf{Q}\Big[(\mathbf{x}(k) - \mathbf{x}_D(k+1)) - \frac{S}{2w'(q)^2}\nabla^T_{\mathbf{u}(k)} w'(\mathbf{q}) \\ - \frac{U}{2(foc)^2}\nabla^T_{\mathbf{u}(k)}(foc) - \frac{V}{2\sigma_{min}^2}\nabla^T_{\mathbf{u}(k)}\sigma_{min} + \frac{W}{2}\nabla^T_{\mathbf{u}(k)}(fov) \Big] \tag{8}$$

The term *foc* represents a focus measure; ***fov*** represents a field of view measure; $w(\mathbf{q})$ represents a manipulability measure for the manipulator holding the active camera; and σ_{min} represents the minimum singular value of the image Jacobian. These sensor placement measures are discussed in detail in [6].

4. A Framework of Schemas and Agents

Our system framework of schemas and port-based agents for robotic manipulation draws many of its concepts from past work in the development of expectation-based and verification approaches for guiding mobile robots. An expectation-based approach to scene understanding was first explicitly proposed by Dickmanns [2]. His work was originally concerned with guiding autonomous mobile systems in rapidly changing environments, particularly autonomous vehicles and aircraft. Roth and Jain propose a "verification-based" approach to navigation in the world [11]. A key point of both the expectation and verification-based approaches is that strong internal models of the recent world state are maintained. This significantly reduces the number of hypotheses that must be considered when determining the current state of the world based on sensory data. Neisser's view of the human "perceptual cycle" [8], as Jain points out [5], is similar in many ways to a verification or expectation-based approach. Figure 5 shows a modified representation of Neisser's "perceptual cycle." This figure illustrates our view of the relationship between object schemas of the world, the real world, and where visually servoed agents exist within this scheme. The counter-clockwise flow of information represents the cyclical nature of the system; sensory data updates schemas, which in turn provide inputs to visually servoed agents, which provide sensory data obtained from the real world to the environment model and the object schemas existing within the environment model. This cycle illustrates the interaction between perception of the world, actions taken within this world, and plans made about the world.

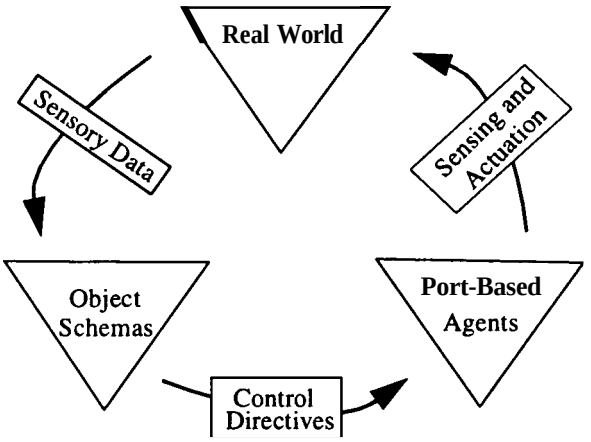

Figure 5. A modified "perceptual cycle" for visually servoed manipulators.

Our proposed integration of visually servoed agents into systems capable of performing manipulation tasks in the real world uses this cycle to clearly delineate the flow of information from the schemas to the visually servoed port-based agents. By clearly separating the system in this way, different types of sensor-based manipulation strategies can be more easily integrated into the system. This is because the internal representation of the task becomes clearly separate from the hardware and low-level behavior-based control systems which perform actions and sensing in the real world.

It is within the environment modeler that the object schemas exist. "Goal achieving functions'' provided by a supervisor or some type of motion planner operate on the object schemas in order to describe the task. For example, a supervisor could use a graphical repre-

sentation of the internal schema representation rather than live video imagery to guide his or her actions. This is similar to the remote teleoperative techniques described in [3] and [4].

5. Hardware Implementation

The schema-agent framework has been implemented on a robotic assembly system consisting of three Puma 560's called the Troikabot. The Pumas are controlled using the Chimera 3.0 reconfigurable real-time operating system[12]. An Adept robot is also used for providing accurate target motion for experimental purposes.

A Datacube Maxtower Vision System calculates the optical flow of the features using the SSD algorithm discussed in Section 3.1. An image can be grabbed and displacements for up to five 16x16 features in the scene can be determined at 30Hz. Stereo system implementations result in half the sampling frequency because only a single digitizer exists on the Datacube. For the stereo system with which experimental results were obtained, visual servoing is performed at 15Hz while tracking ten features. The vision system VME communicates with the robot controller VME using BIT3 VME-to-VME adapters.

The dynamic environment model runs on a Silicon Graphics Crimson. The modeler communicates with the VME-based robot controllers via a Bit3 VME-VME hardware adapter. This allows direct shared memory communication between the motion planner and the manipulators running under the Chimera real-time operating system.

6. Experimental Results

This section demonstrates the use of a moving camera for extending the workspace in which objects can be grasped. Two object schemas are used to represent the gripper and the object to be grasped. A visually servoed port-based agent attempts to achieve the desired visual representation of the scene, while a dynamic sensor placement port-based agent determines appropriate camera motion based on task characteristics. The motion of the visually servoed port-based agent is induced by the object schema representing the gripper. Motion of the dynamic sensor placement port-based agent is induced by the object to be grasped. As dictated by the system framework, the communication between these agents occurs through the task as represented by object schemas. No direct communication between the agents themselves occurs.

Two sets of experimental results are presented. In the first set of results, the dynamic sensor placement agent pans the camera in order to keep the object being tracked within the field-of-view of the camera while the gripper maintains its alignment with the moving object. This illustrates the advantages of using simple sensor behaviors to extend the tracking region of the system. This also demonstrates visually servoed manipulation with a visually servoed camera. The next set of results extends the behavior of the dynamic sensor placement port-based agent to simultaneously maintain the field-of-view behavior by panning and translating along the optical axis of the camera. Vision resolvability is also incorporated as a sensor behavior in order to improve the resolution with which the object is observed and to guarantee convergence of the sensor placement port-based agent.

6.1. Maintaining the Field-of-View Measure During Grasp Alignment

Visually guided manipulation obviously fails when the projection of objects being tracked does not remain on the image plane. Simply by panning a camera about its vertical axis, the visible workspace can be significantly extended. Figure 6 compares the tracking path for visually servoed manipulation that uses two statically mounted cameras with a larger path that is made possible by panning one of the cameras during the alignment task. The controllable camera views the objects's path from a location in the -Z+X quadrant of the graphs shown in Figure 6. The most obvious benefit of allowing the camera to pan is that objects are suc-

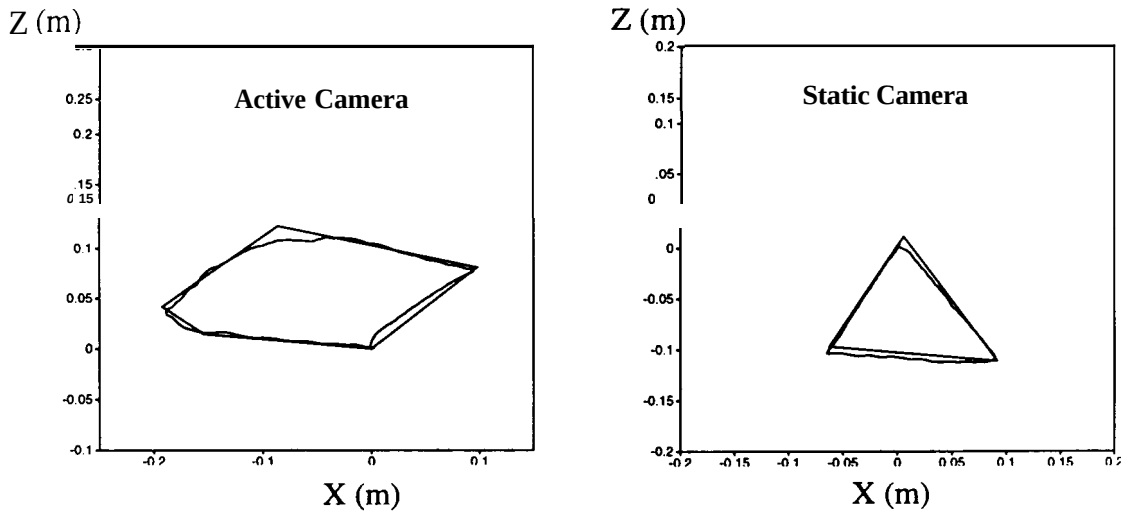

Figure 6. Tracking paths

cessfully tracked for twice the distance along the X axis as they are when the camera remains stationary. From Figure 6, one can observe that the gripper moves approximately 30cm along the X axis. For the static camera case, the maximum possible translational motion along X is approximately 15cm.

The motion of the camera is controlled by the dynamic sensor placement agent and is induced as features on the object and the gripper approach the boundary of the image plane. As the gripper and the object translate along X, the field-of-view measure rises as measured features approach the boundaries of the image plane. This is shown in the left plot of Figure 7.

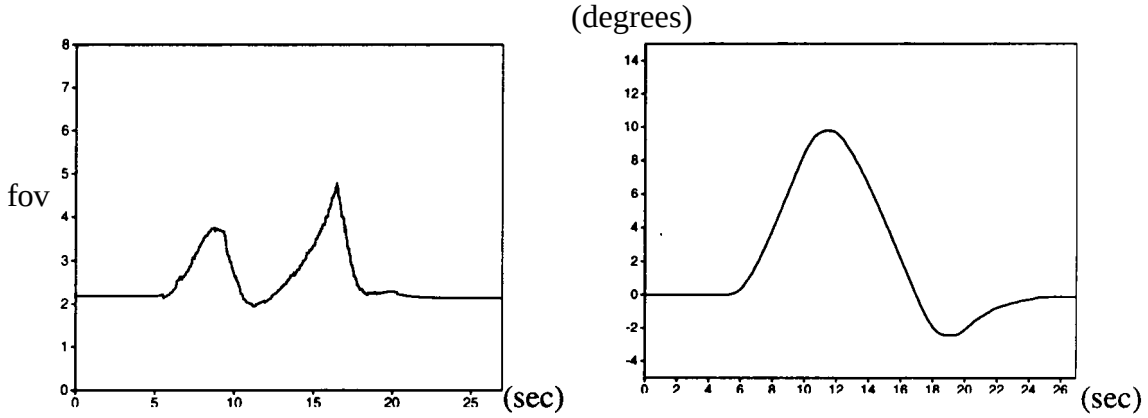

Figure 7. Field-of-view measure and panning angle

This rise in the value of *fov* induces camera motion about Y, and the camera pans with the object and gripper. Figure 8 shows that sensor resolvability is decreased as the depth of the object increases; however, kinematic manipulability remains constant because motion about the panning axis has no effect on singularities. Although this particular configuration for the dynamic sensor placement port-based agent allows no degree of freedom for increasing the resolvability of the vision sensor, the simple field-of view behavior is successful in extending the usable workspace of the system.

6.2. Maximizing Vision Resolvability During Grasp Alignment

In this experiment, camera motion along Z, the optical axis, is allowed in addition to

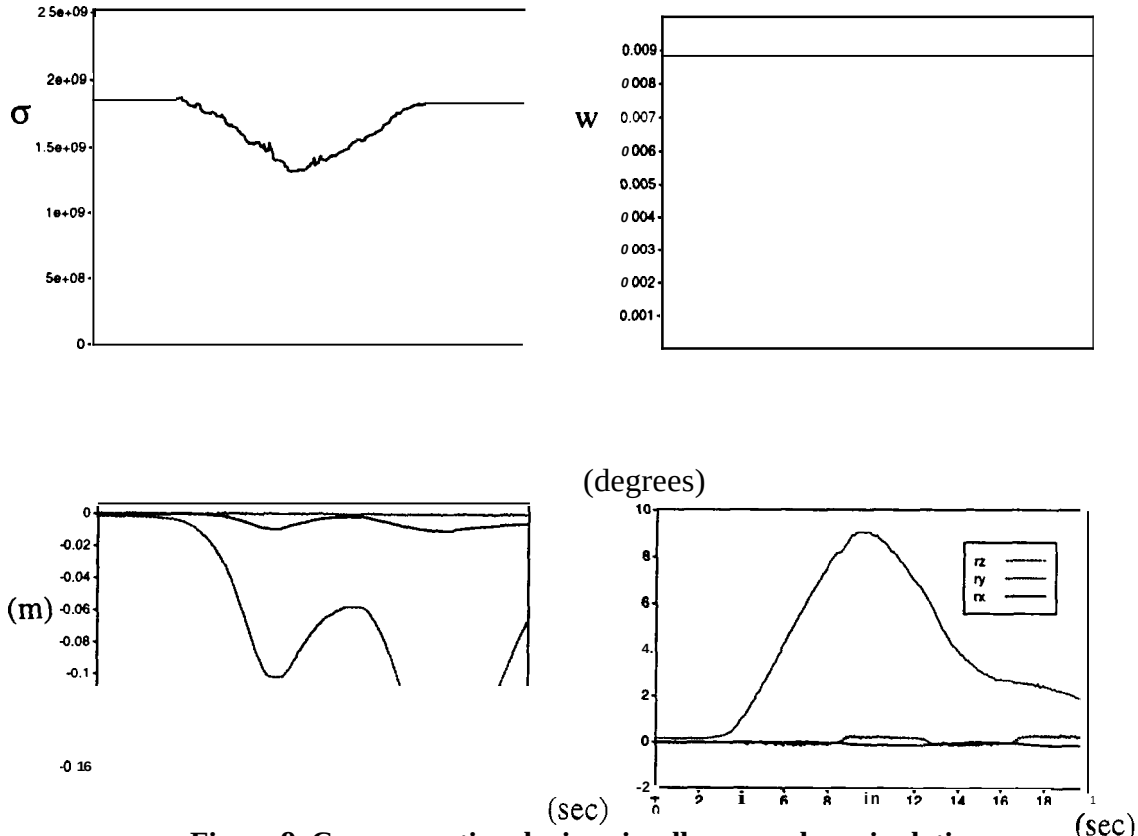

Figure 9. Camera motion during visually servoed manipulation

motion about **Y** in order to minimize the field-of-view measure. If field-of-view is the only sensor placement criteria governing motion along **Z,** the manipulator would attempt to move as far from the object as possible and would have no impetus to approach the task. Therefore, in order to ensure a convergent algorithm, resolvability is also considered along Z in order to place a cost on straying too far from the objects of interest. Manipulability is also considered in order to ensure that a kinematic singularity or a joint limit is not reached by the manipulator providing camera motion.

Figure 9 shows translational and rotational camera motion while gripper alignment is maintained with the object being tracked. Figure 10 shows the motion of the gripper during object motion. From the graphs one can observe the camera panning to bring the object back towards the center of the field-of-view and approaching the object in order to increase vision resolvability as the object moves along **X. As** the object reverses its motion along **X,** the increase in the field-of-view measure cannot be compensated for by panning alone and the camera begins to move away from the object. Eventually, the field- of-view measure decreases sufficiently to allow vision resolvability to drive the camera back toward the object being observed. The objective of the dynamic sensor placement port-based agent is to minimize the field-of-view measure, the inverse of vision resolvability, and the inverse of manipulability.

7. Conclusion

A framework and task representation for integrating visually servoed manipulation strategies with an active camera system **has** been described and experimental results presented. **A** key component of the framework is the use **of** an expectation-based approach in which augmented geometric models called object schemas are used to describe and reason about the environment. Object schemas are continuously updated based on sensor feedback to insure that the current state of the environment model agrees with current sensory input. Visually servoed port-based agents are used to ensure that desired actions described within the environ-

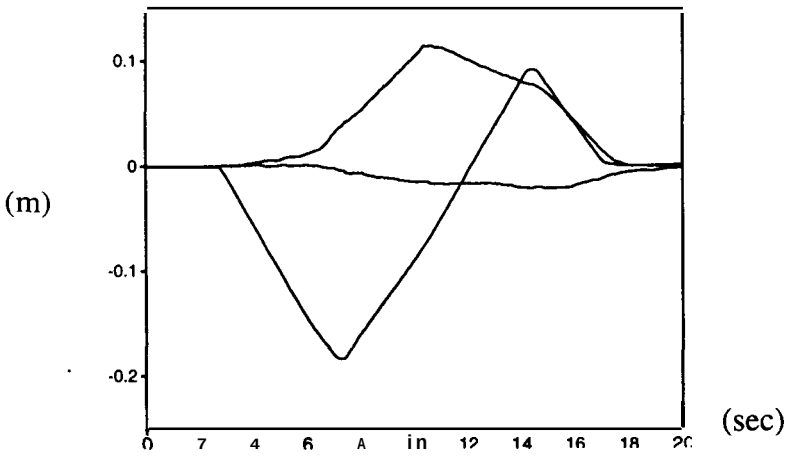

Figure 10. End-effector motion during grasp alignment with a moving camera

ment model are carried out in the real world. These actions include manipulation and dynamic sensor reconfiguration. Our goal is to develop a scalable **and** extendable framework that will allow the use of plug-and-play types of sensor-based manipulation components.

Acknowledgments

This research was supported in part by the **U.S.** Army Research Office through Grant Number DAAL03-91-G-0272 and by Sandia National Laboratories through Contract Number AC-3752D.

References

[1] P. Anandan, Measuring visual motion from image sequences. Tech. Rept. COINS-TR-87-21, Amherst, Mass., University of Massachusetts COINS Department, 1987.

[2] E.D. Dickmanns, "Expectation-based Dynamic Scene Understanding," in ***Active Vision,*** eds. A. Blake and A. Yuille, 303-335, The MIT Press, Cambridge, 1992.

[3] C. Fagerer, D. Dickmanns, and E.D. Dickmanns, "Visual Grasping with Long Delay Time of a Free Floating Object in Orbit," ***Autonomous Robots,*** 1(1):53-68, 1994.

[4] G. Hirzinger, "ROTEX-the first space robot technology experiment," ***Experimental Robotics Ill: The Third Int. Symp., Kyoto, Japan, Oct. 28-30, 1993,*** eds. T. Yoshikawa and F. Miyazaki, Springer-Verlag, pp.579-598, 1994.

[5] R. Jain, "Environment Models and Information Assimilation," Technical Report RJ 6866(65692), IBM-Yorktown Heights, 1989.

[6] B. Nelson and P.K. Khosla, "Integrating Sensor Placement and Visual Tracking Strategies," in ***Experimental Robotics III: The Third International Symposium, Kyoto, October 28-30, 1993,*** eds. T. Yoshikawa and F. Miyazaki, Springer-Verlag, London, pp. 169-181, 1994.

[7] B.J. Nelson and P.K. Khosla, "The Resolvability Ellipsoid for Visual Servoing," in ***Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR94),*** pp. 829-832, 1994.

[8] U. Neisser, ***Cognition and Reality,*** W.H. Freeman and Co., New York, 1976.

[9] N.P. Papanikolopoulos, Khosla, P.K. and Kanade, T., "Adaptive robotic visual tracking," ***Proc. of the American Control Conference.*** Evanston, IL.:American Automation Control Council, pp. 962-967, 1991.

[10]N.P. Papanikolopoulos, Nelson, B. and Khosla, P.K., "Full 3-d tracking using the controlled active vision paradigm," ***Proc. I992 IEEE Int. Symp. on Intelligent Control (ISIC-92),*** pp. 267-274, 1992.

[11]Y. Roth and R. Jain, "Integrated Architectures for Autonomous Systems," ***Proc. of the SPIE - The Int. Society for Optical Engineering,*** vol. 1571, pp. 628-639, 1991.

[12]D.B. Stewart, Schmitz, D.E. and Khosla, P.K., "The Chimera 11 real-time operating system for advanced sensor-based control systems," ***IEEE Trans. Sys., Man Cyber.*** 22(6):1282-1295., 1992.

[13]L.E. Weiss, "Dynamic visual servo control of robots: an adaptive image-based approach," Ph.D Thesis CMU-RI-TR-84-16, Pittsburgh, PA:The Robotics Institute Carnegie Mellon University, 1984.