Relative Position Estimation for Manipulation Tasks by Fusing Vision and Inertial Measurements

Andreas Huster and Stephen M. Rock
Aerospace Robotics Lab, Stanford University, CA, 94305
and Monterey Bay Aquarium Research Institute, Moss Landing, CA, 95039
{huster,rock}@arl.stanford.edu

Abstract—**This paper proposes a position estimator that fuses monocular vision with accelerometer and gyro measurements to generate a direct, relative, 6-DOF position estimate between a free-floating underwater vehicle and a stationary object of interest. This type of position estimate is useful in the context of autonomous manipulation tasks, during which the vehicle needs to control its position relative to objects in its environment. Most autonomous manipulation tasks are vision-based and assume known camera motion. However, on free-floating vehicles, camera motion is generally unknown and must be estimated together with relative position. Various vision-only systems have been used to estimate relative position and camera motion, but these are often difficult to implement in real underwater environments.**

The system we propose relies on vision to generate relative position information, but also fuses inertial rate sensors to reduce the amount of information that needs to be extracted from the vision system. The result is a system that is simpler and more robust than a vision-only solution. However, the use of inertial rate sensors introduces several issues. The rate measurements are subject to biases, which need to be estimated to prevent the accumulation of unbounded drift when the measurements are integrated. This problem is non-linear, which presents several challenges in the estimator design.

This paper presents some results from initial experiments with a fixed-base manipulator. So far, we have implemented a simplified estimator for relative position when camera motion is known. The estimator is part of the closed-loop control of the manipulator. With this system, we have demonstrated a simple autonomous manipulation task.

I. Introduction

This paper describes a position sensing system for free-floating underwater vehicles capable of performing autonomous manipulation tasks, such as placing sensors and retrieving samples. These are complex and difficult tasks composed of modeling, planning and execution phases during which the robot requires accurate control (or at least knowledge) of its position and orientation relative to the object of interest. Free-floating vehicles, which cannot lock their position (e.g., by thrusting into the sea floor) during an autonomous manipulation task, require a real-time estimate of the relative 6-DOF (degrees of freedom) position between the vehicle and the object of interest. This paper proposes a sensing strategy that enables underwater vehicle control relative to objects in the environment.

Underwater manipulation tasks are usually performed by remotely operated vehicles (ROVs) or manned submersibles. These tasks require the control of many degrees of freedom of the vehicle and the manipulator and are performed by very experienced pilots. Typically, the vehicle is thrust into the sea floor or locked to a structure with an auxiliary arm to simplify the task. Very simple manipulation tasks (e.g., sampling of volcanic glass with the Autonomous Benthic Explorer [1]) have also been demonstrated on autonomous underwater vehicles (AUVs).

Our focus is on autonomous manipulation tasks that can relieve ROV pilots and that enable new manipulation capabilities on AUVs (which cannot be controlled directly by a human pilot). While humans are superior at analyzing a scene and specifying tasks, computers can often execute manipulation tasks more accurately and efficiently. Computers can derive an advantage from incorporating a larger suite of sensors and coordinating more degrees of freedom. However, determining the position of objects in the environment is still a major challenge in autonomous manipulation.

When the underwater vehicle and the base of the manipulator are fixed in inertial space, the scenario of underwater autonomous manipulation tasks is similar to its land-based equivalent. For land-based robots, various researchers have demonstrated autonomous manipulation of known and prepared objects (e.g., [2]) and others have made progress towards autonomous manipulation of unknown and *a priori* unmodeled objects [3], [4], [5]. Most of these techniques use vision to determine the relative 6-DOF position between the robot and the object. Because the robot is fixed in inertial space, this work assumes that camera motion is known.

However, fixing the underwater vehicle is not always possible (e.g., when there is nothing to thrust into) or desirable (e.g., when thrusting would generate a cloud of dust). We therefore consider manipulation from a free-floating base. In this case, the sensing system has to provide the relative 6-DOF position between camera and object even though the camera motion is unknown.

Several vision-only techniques (e.g., Structure from Motion, Photogrammetry) can estimate unknown camera motion together with relative position. However, most of these techniques are difficult to implement in real underwater environments because they require a large number of trackable image features or rely on feature correspondences in multiple cameras. Underwater manipulation tasks may occur in environments with only a small number of good visual features, given the constraints on lighting and visibility imposed by real underwater environments. In this case, robust tracking of multiple features and maintaining reliable correspondences, which are the basis of vision-only techniques, can be very difficult to achieve.

The system we propose relies on monocular vision to generate relative position information, but also fuses inertial rate sensors to reduce the amount of information that needs to be extracted from the vision system. The result is a system that is simpler and more robust than a vision-only solution. Our system requires only a bearing measurement to a fixed point, which can be obtained by tracking a single visual feature. Usually, this feature is associated with the object to be manipulated. Compared to vision-only algorithms, tracking only the best feature in a monocular vision system can be much faster and more reliable.

The measurements from monocular vision and inertial rate sensors complement each other well. The motion of the camera between successive images generates a baseline for range computations by triangulation. Inertial rate sensors, whose acceleration and angular rate measurements can be integrated to obtain vehicle velocity, position and orientation, can account for the 6-DOF motion of the camera along this baseline. When these measurements are fused, the relative position between the camera and the object can be computed. A key benefit of this system is that, after initialization, the inertial rate sensors continue to maintain a useful estimate of relative position during vision drop-outs (e.g., occlusions, lack of correspondence). Furthermore, both inertial rate sensors (for navigation) and monocular vision systems (for science purposes) are already common sensors on underwater vehicles.

However, the use of inertial rate sensors introduces a major issue into the design of the system. Like other dead-reckoning sensors, inertial rate sensors suffer from bias and random noise errors, which lead to unbounded drift in the integrated quantities. While more expensive sensors are associated with less drift, we envision the use of low-cost inertial sensors, which are subject to significant drift errors. Therefore, an estimator to resolve the relative camera position, sensor biases and drift errors is required.

Observability of the states to be estimated is a critical issue because the problem is non-linear and observability depends on the motion of the camera. During camera translation directly towards or away from the feature, the estimator has no new information with which to improve its range estimate. Only camera motions transverse to the feature direction provide new information for the range estimate. As a result, the estimator requires *sufficient* transverse camera motion in order to produce useful position estimates. In some cases, extra camera maneuvers are required to improve observability. This presents an interesting conflict between trajectories which are designed to complete the manipulation task and special camera maneuvers required to ensure estimator observability.

The estimation problem contains two significant nonlinearities. The first is related to the rotational degrees of freedom of the camera and the second is caused by the camera's projection of the three-dimensional world onto the 2D image plane. As a result, the dynamics and measurement equations are non-linear and depend on the actual state of the system. In fact, the estimator exploits the non-linearity of the problem to observe the range to the feature. As motion of the underwater vehicle modifies the system state, the measurement equations change, and new measurements (i.e., bearings to the object from new viewpoints) make the range to the object observable.

Section II summarizes the estimation problem and presents models for the vision and inertial rate sensor measurements. Section III reviews our previous work in merging vision and inertial rate sensors.

We are using two experimental platforms to support this research. We will perform an underwater vehicle demonstration on OTTER (see Fig. 1 and [6]), a small AUV operated in a test tank at MBARI, the Monterey Bay Aquarium Research Institute. We are also conducting experiments in the laboratory with a fixed-base 7-DOF manipulator arm, described in Section IV. This platform provides a very accurate truth measurement and can be used to investigate competing approaches, to simulate different disturbance environments, and to quantify performance.

Fig. 1. OTTER: a small underwater vehicle operated in MBARI's test tank

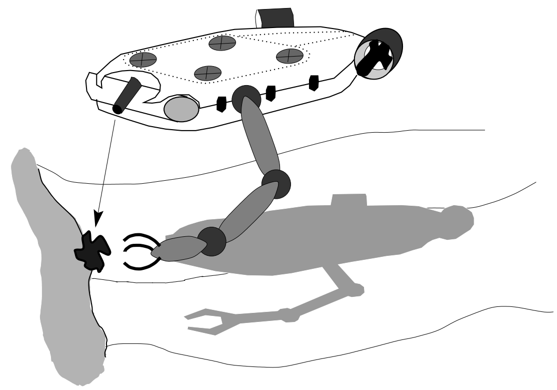

Fig. 2. Relative position estimator scenario

This research is still in progress and we are reporting on initial experiments. The estimator we discuss in Sections V and VI assumes known camera motion and does not need to fuse inertial measurements. It combines known camera motion with the bearing to a single, stationary visual feature to estimate relative position of the feature. To demonstrate the effectiveness of this technique, the manipulator uses the position estimate to locate and press a button.

The experiment described in this paper highlights how the estimator can be combined with a trajectory generator and a controller to build a system that can accomplish a simple manipulation task. Once we complete the development of an algorithm that can estimate relative feature position as well as camera motion (by fusing inertial rate measurements), we can integrate this new capability into the existing control structure. Similarly, an algorithm that can provide more efficient trajectories can be substituted for the current trajectory generator.

Finally, Section VII states our conclusions and outlines future work.

II. Problem Definition

A. Estimation Scenario

Fig. 2 shows an underwater vehicle which has extended its manipulator arm to grasp a stationary object in the environment. To succeed, the grasping task requires a relative 6-DOF position measurement of the object in body coordinates. A camera on the vehicle is tracking the object and inertial rate sensors are reporting the vehicle's acceleration and angular velocity. These measurements can be fused to estimate relative position.

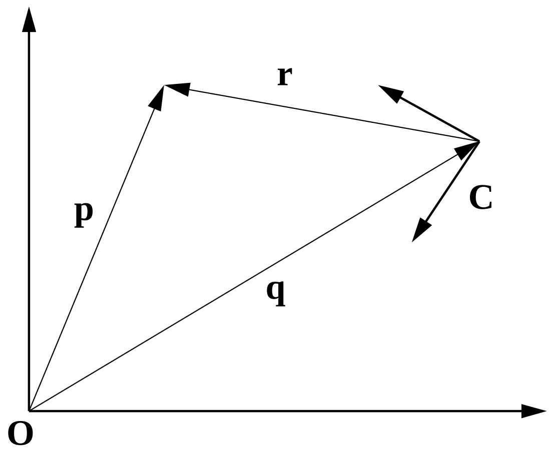

Fig. 3. Geometry of the estimation problem

The geometry of the estimation problem is represented in Fig. 3. Frame $\mathbf{O}$ indicates the inertial reference frame and $\mathbf{p}$ indicates the position of the stationary feature. Let $\mathbf{q}$ be the position of the camera, which is attached to the body frame $\mathbf{C}$. In principle, the camera and the inertial sensors can be mounted anywhere on the vehicle, as long as their relative position and orientation are known. To simplify the discussion, we assume that all the sensors are collocated at $\mathbf{q}$. $R_{oc} = R_{co}^T$ is the rotation matrix from the camera to the inertial frame and $\boldsymbol{\omega}$ is the associated rotational velocity.

In general, $\mathbf{q}$ and R_{oc}, which represent the camera motion, are not known and need to be estimated. However, for the experiments presented in this paper, we still assume that $\mathbf{q}$ and R_{oc} are known.

The position of the feature as seen by the camera is $\mathbf{r} = \mathbf{p} - \mathbf{q}$. We assume that the feature is stationary in the inertial frame, so $\dot{\mathbf{p}} = \ddot{\mathbf{p}} = \mathbf{0}$. Therefore, $\dot{\mathbf{r}} = -\dot{\mathbf{q}}$ and $\ddot{\mathbf{r}} = -\ddot{\mathbf{q}}$. Because of this assumption, a measurement of the acceleration $\ddot{\mathbf{q}}$ in inertial space is useful for estimating the relative feature position $\mathbf{r}$.

We use the superscript C (e.g., ${}^C\mathbf{r} = R_{co}\mathbf{r}$) to indicate that the vector is resolved in Frame $\mathbf{C}$ instead of inertial coordinates.

B. Models for the Sensor Measurements

The vision measurement $\mathbf{z}_s$ is the projection of ${}^C\mathbf{r}$ onto the image plane, and is modeled as follows:

$$\begin{aligned} {}^C\mathbf{r} &= R_{co}(\mathbf{p} - \mathbf{q}) && (1) \\ \rho &= {}^C r_z && (2) \\ \mathbf{z}_s &= \begin{bmatrix} s_x \\ s_y \end{bmatrix} = \frac{1}{\rho} \begin{bmatrix} {}^C r_x \\ {}^C r_y \end{bmatrix} + \mathbf{v}_s && (3) \end{aligned}$$

where $\mathbf{v}_s$ is zero-mean Gaussian noise $\mathcal{N}(0, R_s)$. For simplicity, we assume that the camera measurements are normalized so the effective focal length is 1. With a slight abuse of terminology, we will refer to ρ as the range to the feature and to s_x and s_y as the bearing. The optical axis of the camera is aligned with the z-axis of Frame $\mathbf{C}$.

The accelerometer measurement $\mathbf{z}_a$ includes the acceleration $\ddot{\mathbf{q}}$ of the camera, the acceleration due to gravity, a sensor bias $\mathbf{b}_a$, and sensor noise.

$$\mathbf{z}_a = R_{co}(\ddot{\mathbf{q}} + \mathbf{g}) + \mathbf{b}_a + \mathbf{v}_a \tag{4}$$

We assume that $\mathbf{v}_a$ is $\mathcal{N}(0, R_a)$ noise. $\mathbf{g} = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}^T$ is the acceleration due to gravity in inertial coordinates.

The gyro measurement includes the rotational velocity of the camera, sensor bias $\mathbf{b}_\omega$, and sensor noise $\mathbf{v}_\omega$ modeled by $\mathcal{N}(0, R_\omega)$.

$$\mathbf{z}_\omega = R_{co}\boldsymbol{\omega} + \mathbf{b}_\omega + \mathbf{v}_\omega \tag{5}$$

III. Previous Work

The measurement equations in (1) to (5) include several non-linearities. Because of the rotational degrees of freedom of the vehicle, the dynamics are also non-linear.

Various methods exist to handle non-linear estimation problems. We consider estimators based on extensions to the popular Kalman Filter. The most widely used is the Extended Kalman Filter (EKF), which linearizes the dynamics and measurement equations of non-linear systems in order to take advantage of the Kalman Filter equations. Although the EKF works very well for a wide range of applications, it comes with no guarantees and can lead to very poor performance. An alternative to the EKF is the two-step estimator [7], which reformulates the problem so that all the measurement equations become linear and all of the non-linearities appear in the dynamics, where they can be handled more accurately with computational methods.

In [8], we derive an EKF and a two-step estimator for a 2D version of the estimation problem, and we present a simulation that compares their ability to fuse vision and inertial rate measurements. We compare the two estimators based on the accuracy of the state estimate, the accuracy of the covariance estimate, and the tendency of the estimators to diverge. This simulation indicates that in the context of this fusion problem, the two-step estimator performs significantly better.

In this paper, however, we derive only an EKF because the estimation problem is simpler (assumes known camera motion) and the EKF, which is easier to derive and implement, performs sufficiently well in this case.

IV. Laboratory Testbed

The goal of this research is to develop a relative position sensing strategy which is useful for autonomous manipulation on underwater vehicles. However, in our initial work, we have used a fixed-base manipulator. Fig. 4 shows the K-1607 manipulator built by Robotics Research Corporation[1] and located at NASA Ames Research Center. It is a 7-DOF, kinematically redundant manipulator whose endpoint can be moved to any position and orientation in its workspace. We have collocated the camera and the inertial rate sensors on the endpoint of the manipulator.

We can use this general purpose manipulator to simulate various conditions. For example, it simulates an eye-in-hand manipulator mounted on an underwater vehicle which has been fixed in inertial coordinates, possibly by thrusting into the sea floor or by attaching itself to a structure. We can also treat

[1] http://www.robotics-research.com

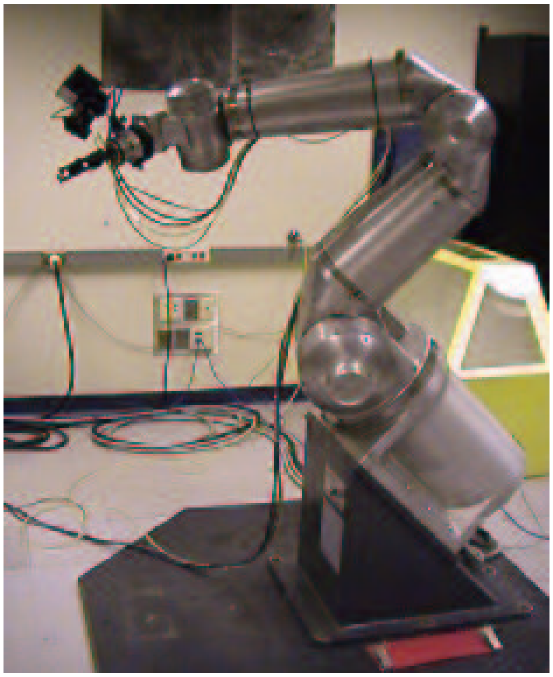

Fig. 4. RRC K-1607 7-DOF Manipulator

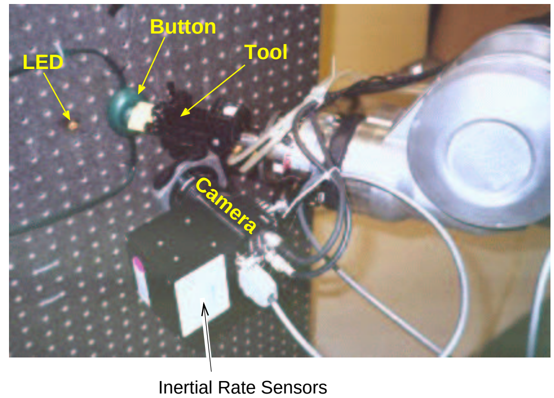

Fig. 5. Manipulator endpoint with camera and inertial rate sensors positioned close to the IR-LED and the push button

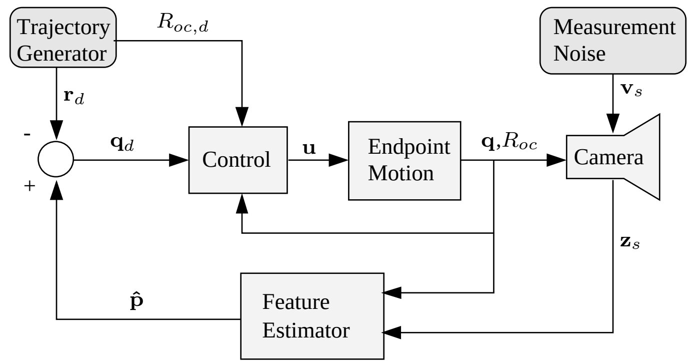

Fig. 6. Block Diagram showing the Vehicle, Controller, Sensors, and Estimators

the manipulator endpoint as the underwater vehicle and use the manipulator to simulate the motion of the vehicle due to thruster and disturbance forces. All of the manipulator joints are instrumented with encoders, so that the exact position of the endpoint can be computed. This provides a truth measurement which is usually not available when using operational vehicles.

While vision processing for underwater environments remains a challenging problem, many researchers have already developed useful algorithms that can be used to track a feature on the object of interest. In our current research, we assume that a robust point-feature can be tracked, and focus instead on integrating this type of measurement into a position estimator. For this experiment, we use an infrared LED to simulate a visual feature. The camera uses an infrared filter which blocks most of the ambient light. When the LED is in the camera's field-of-view, the resulting image contains a bright spot which can be tracked by simple threshold methods. To measure acceleration and rotational velocity, we use the DMU-6X Inertial Measurement Unit by Crossbow[2].

Our experiment includes a simple manipulation task to demonstrate the usefulness of our position estimator. Most autonomous manipulation tasks are difficult to specify and involve complex modeling, planning, and execution phases as well as task-specific sensors. The task we have chosen is much easier to implement because it only requires the manipulator to push a button to toggle a light switch. Specifying the position of the button and the pushing direction completely defines the task. The size of the button determines the required accuracy of the position measurement.

Fig. 5 shows the manipulator endpoint positioned close to a board on which the button is mounted. The endpoint has a tool used to push the button, a camera with an infrared filter, and the inertial rate sensors. An infrared LED, which serves as the visual feature, is mounted on the board to the left of the button. The position of the LED is estimated relative to the camera, and the manipulator tool is used to push the button. The necessary offset between estimated feature position and desired tool position is a constant that can be determined by a simple calibration procedure.

[2]http://www.xbow.com

V. Simplified Estimation Problem

We have used the manipulator to demonstrate a simplified version of the estimation problem in which the camera motion is known in inertial space. We simulate this scenario by providing the estimator with $\mathbf{q}$ and R_{oc} derived from the manipulator encoders. The estimator then computes $\hat{\mathbf{p}}$ based on vision measurements, which provide the projection of $\mathbf{r}$ onto the image plane.

Fig. 6 shows a block diagram for this system. The trajectory generator specifies a desired camera offset $\mathbf{r}_d$ from the feature and a desired camera orientation $R_{oc,d}$. This leads to a desired camera position

$$\mathbf{q}_d = \hat{\mathbf{p}} - \mathbf{r}_d. \tag{6}$$

The controller computes a command $\mathbf{u}$ that drives the camera to position $\mathbf{q}$ and orientation R_{oc}. The control design is beyond the scope of this paper. At this position, the camera generates a measurement $\mathbf{z}_s$ using (3). This measurement is used to update $\hat{\mathbf{p}}$ in the estimator.

We implemented the estimator as an EKF [9] with $\mathbf{p}$ as the state vector and a trivial time update $\dot{\mathbf{p}} = 0$. Thus the EKF time-update equations are given by

$$\hat{\mathbf{p}}_{i+1|i} = \hat{\mathbf{p}}_{i|i} \tag{7}$$
$$P_{i+1|i} = P_{i|i} + Q/T. \tag{8}$$

Let T be the sample interval with measurements recorded at time $t = iT$, $i = 0, 1, \ldots N$. The notation $\hat{p}_{a|b}$ indicates an estimate at time-step a using measurements up to time-step b. P is an estimate of the error covariance and Q represents fictitious process noise to keep the estimator from falling asleep.

The measurement update equations use a non-linear function $h(\mathbf{p})$ and a time-varying matrix H_i to capture the non-linearity of the measurement equation.

$$L_i = P_{i|i-1} H_i^T \left(R + H_i P_{i|i-1} H_i^T\right)^{-1} \tag{9}$$
$$\hat{\mathbf{p}}_{i|i} = \hat{\mathbf{p}}_{i|i-1} + L_i \left(\mathbf{z}_{s,i} - h\left(\hat{\mathbf{p}}_{i|i-1}\right)\right) \tag{10}$$
$$P_{i|i} = \left(I - L_i H_i\right) P_{i|i-1} \tag{11}$$

We can derive $h(\mathbf{p})$ from (3). We first label the rows of $R_{co,i}$

$$R_{co,i} = \begin{bmatrix} R_{1,i} \\ R_{2,i} \\ R_{3,i} \end{bmatrix}, \tag{12}$$

which allows us to rewrite (3) as

$$\mathbf{z}_{s,i} = h(\mathbf{p}_i) + \mathbf{v}_{s,i} \tag{13}$$
$$h(\mathbf{p}_i) = \begin{bmatrix} h_1(\mathbf{p}_i) \\ h_2(\mathbf{p}_i) \end{bmatrix} \tag{14}$$
$$h_1(\mathbf{p}_i) = \frac{R_{1,i}(\mathbf{p}_i - \mathbf{q}_i)}{R_{3,i}(\mathbf{p}_i - \mathbf{q}_i)} \tag{15}$$
$$h_2(\mathbf{p}_i) = \frac{R_{2,i}(\mathbf{p}_i - \mathbf{q}_i)}{R_{3,i}(\mathbf{p}_i - \mathbf{q}_i)} \tag{16}$$

To compute H_i, we write the first-order Taylor expansion of $h(\mathbf{p}_i)$ about the prior estimate $\hat{\mathbf{p}}_{i|i-1}$, which is based on the assumption that $\Delta\mathbf{p}_i$ and the higher order terms of the expansion are small.

$$h(\mathbf{p}_i) \cong h\left(\hat{\mathbf{p}}_{i|i-1}\right) + H_i \Delta\mathbf{p}_i \tag{17}$$
$$\Delta\mathbf{p}_i = \mathbf{p}_i - \hat{\mathbf{p}}_{i|i-1} \tag{18}$$
$$H_i = \left.\frac{\partial h}{\partial \mathbf{p}}\right|_{\hat{\mathbf{p}}_{i|i-1}} \tag{19}$$
$$= \frac{1}{\hat{\rho}_i^2} \begin{bmatrix} \hat{\rho}_i R_{1,i} - R_{1,i}\left(\hat{\mathbf{p}}_{i|i-1} - \mathbf{q}_i\right) R_{3,i} \\ \hat{\rho}_i R_{2,i} - R_{2,i}\left(\hat{\mathbf{p}}_{i|i-1} - \mathbf{q}_i\right) R_{3,i} \end{bmatrix} \tag{20}$$
$$\hat{\rho}_i = R_{3,i}\left(\hat{\mathbf{p}}_{i|i-1} - \mathbf{q}_i\right) \tag{21}$$

Because of the non-linearities in the estimation problem, the camera trajectory is important in achieving a good estimate. The camera needs to observe the feature from a variety of directions for the triangulation procedure implicit in the estimator to generate an accurate feature location.

The trajectory that we have chosen has two parts: an exploratory phase and an approach phase. During the exploratory phase, the camera moves along an arc of constant radius centered on the current estimate of the feature. An arc maximizes the range of directions from which the feature is observed. Because the feature estimate changes during the trajectory, the final motion might not represent an exact arc in inertial coordinates.

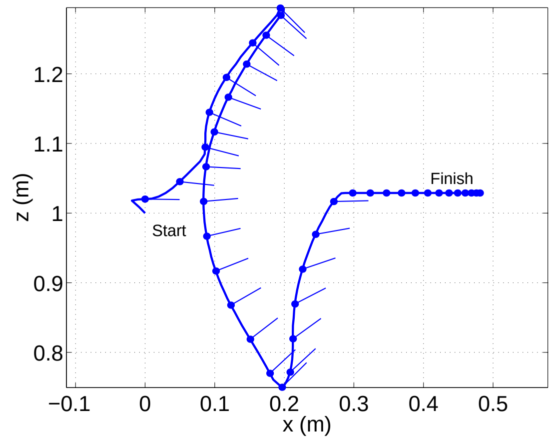

Fig. 7. Camera Motion in the x-z plane. The guides are spaced at 0.5 s intervals and indicate the optical axis of the camera.

During the approach, the camera moves primarily towards the feature, so the camera measurements acquired during this phase provide little new information. At the same time, the measurement update (i.e., equation 20) becomes poorly conditioned as $\hat{\rho}_i$ becomes smaller. Therefore, we stop the estimator at the beginning of the approach phase and complete the task in a blind manner.

VI. Experimental Results

This section presents results obtained from experiments with the K-1607 manipulator. We implemented the estimator as part of the closed-loop control of the manipulator arm. The estimator used camera measurements $\mathbf{z}_s$ of the feature and known camera motion (i.e., $\mathbf{q}$ and R_{oc}) to estimate the feature position $\hat{\mathbf{p}}$.

Before each experiment, the estimator is reset to a new initial estimate $\hat{\mathbf{p}}_0$ of the feature position. The system then moves the camera through a desired trajectory and estimates the position $\mathbf{p}$ of the feature. The trajectory is relative to the current feature estimate and terminates at a predetermined offset to the final feature estimate $\hat{\mathbf{p}}_f$. If this estimate is sufficiently accurate, the manipulator succeeds in pressing the button beside the feature.

To simulate changing conditions, we chose different initial camera positions $\mathbf{q}_0$ for each experiment. The initial estimate of the feature $\hat{\mathbf{p}}_0$ was initialized at a fixed distance in front of the camera.

Fig. 7 shows a plot of the camera motion in the x-z plane for one experiment. Although the camera also moves in the y-direction, the design of the trajectory concentrates most of the motion is in the x-z plane. The camera starts at a position $x = 0.0\,m, y = 0.0\,m$, and $z = 1.0\,m$ and finishes by pressing the button. The task is set up such that the manipulator approaches the button always in the positive x-direction. At the beginning of the task, the only unknown is the position of the LED feature and the button, which are related by a fixed offset. The guides in Fig. 7 indicate the position of the camera and the orientation of its optical axis in $0.5\,s$ intervals.

Fig. 8 shows the estimator results for the same experiment. The estimator is initialized at $x = 0.4\,m, y = 0.0\,m$, and

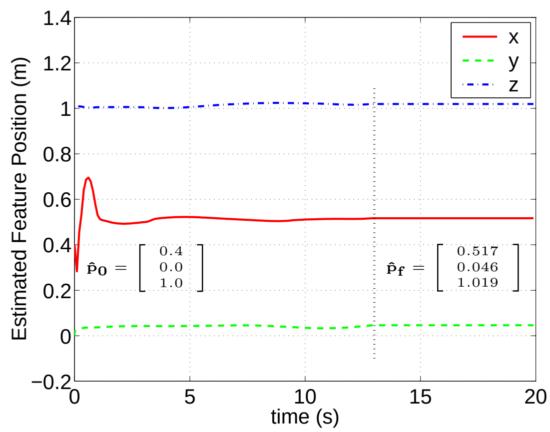

Fig. 8. Estimated Feature Position for one run, with initial and final estimates. The vertical line indicates the beginning of the approach phase when the estimator is turned off.

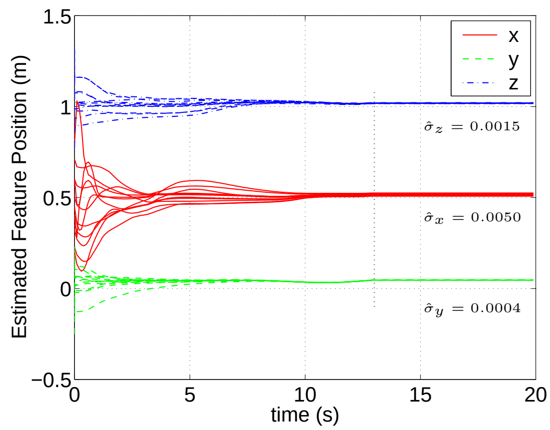

Fig. 9. Estimated Feature Position for twelve runs each with different initial conditions. The estimated standard deviation of the final estimate is shown for each coordinate.

$z = 1.0\,m$, which is simply $0.4\,m$ in front of the camera. The parameters used for the estimator are:

$$T = 0.1\,s \tag{22}$$

$$P_0 = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{bmatrix} \tag{23}$$

$$Q = \begin{bmatrix} 10^{-8} & 0 & 0 \\ 0 & 10^{-8} & 0 \\ 0 & 0 & 10^{-8} \end{bmatrix} \tag{24}$$

$$R = \begin{bmatrix} 0.0001 & 0 \\ 0 & 0.0001 \end{bmatrix} \tag{25}$$

The Q and R matrices are selected to capture any noise and bias errors in the camera measurements and robot joint encoders, errors in the kinematic model of the manipulator, and biases induced by the estimator.

The vertical line in Fig. 8 separates the exploratory phase of the trajectory from the approach phase. The estimator is only active during the exploratory phase. The plot also shows the initial and final estimates for this experiment.

Fig. 9 shows the estimator results for 12 experiments, each with different initial conditions. The initial feature estimate ranges from 0.2 to $0.8\,m$ in x, -0.25 to $0.23\,m$ in y, and 0.75 to $1.34\,m$ in z. Table I shows the mean and standard deviation of the state estimates for these 12 experiments. The robot succeeded in pressing the button every time.

TABLE I

MEAN AND STANDARD DEVIATION OF ESTIMATES

	mean (m)	standard deviation (m)
$\hat{\mathbf{p}}_x$	0.5153	0.0050
$\hat{\mathbf{p}}_y$	0.0460	0.0004
$\hat{\mathbf{p}}_z$	1.0189	0.0015

VII. CONCLUSIONS

We have proposed a relative position estimator for free-floating underwater vehicles capable of performing autonomous manipulation tasks. Such vehicles pose a unique sensor requirement: a robust, 6-DOF, direct estimate of the relative position between the vehicle and a stationary visual feature.

We have defined a sensor strategy that merges vision and inertial rate sensors, discussed its key advantages, and identified some of the challenges of implementing this technique.

In our initial experiments, we used a fixed-base manipulator with an endpoint camera to demonstrate the use of vision and known camera motion to estimate the position of a single stationary feature. The position estimates were used to close the loop on an autonomous manipulation task in which the robot presses a button identified by an infrared LED.

Our future work will focus on four main topics. First, the estimator needs to be expanded to merge inertial rate sensor measurements to estimate camera motion. Some of our work in this direction has already been described in [8], where we have presented an estimator for a two-dimensional world. The next step is to derive a two-step estimator for the full 3D problem. Second, when camera motion is unknown, the quality of the models that predict disturbance dynamics (e.g., from ocean currents) will limit the performance of the estimator. More realistic modeling of disturbance processes will improve the performance of an operational system. Third, we currently use *ad hoc* trajectories to generate sufficient sensor motion. The issue of blending optimal maneuvers to improve observability with the execution of useful manipulation tasks remains to be explored. Finally, we expect to demonstrate the estimator as part of an autonomous manipulation task on a free-floating underwater vehicle.

ACKNOWLEDGMENTS

This research was supported in part by the Packard Foundation under Grants 98-3816 and 98-6228. We also would like to thank NASA Ames Research Center for providing access to the K-1607 manipulator.

REFERENCES

[1] Dana R. Yoerger, Albert M. Bradley, Marie-Helene Cormier, William B. F. Ryan, and Barrie B. Walden, “High resolution mapping of a fast spreading mid ocean ridge with the autonomous benthic explorer,” in *Proceedings of*

the 11th International Symposium on Unmanned Untethered Submersible Technology, Durham, NH, August 1999, Autonomous Undersea Systems Institute.

[2] Christopher E. Smith and Nikolaos P. Papanikolopoulos, "Grapsing of static and moving objects using a vision-based control approach," *Journal of Intelligent and Robotic Systems*, vol. 19, no. 3, July 1997.

[3] Issa A. D. Nesnas, Mark W. Maimone, and Hari Das, "Rover maneuvering for autonomous vision-based dexterous manipulation," in *IEEE International Conference on Robotics and Automation*, San Francisco, CA, April 2000, IEEE, vol. 3, pp. 2296–2301.

[4] Matthias Seitz, "Towards autonomous robotic servicing: Using an integrated hard-arm-eye system for manipulating unknown objects," *Robotics and Autonomous Systems*, vol. 26, no. 1, pp. 23–42, January 1999.

[5] S. A. Stansfield, "Robotic grasping of unknown objects: A knowledge-based approach," *The International Journal of Robotics Research*, vol. 10, no. 4, pp. 314–326, August 1991.

[6] H. H. Wang, S. M. Rock, and M. J. Lee, "OTTER: The design and development of an intelligent underwater robot," *Autonomous Robots*, vol. 3, no. 2-3, pp. 297–320, June-July 1996.

[7] N. Jeremy Kasdin, "The two-step optimal estimator and example applications," in *Guidance and Control 2000*, Robert D. Culp and Eileen M. Dukes, Eds., San Diego, CA, 2000, American Astronautical Society, vol. 104 of *Advances in the Astronautical Sciences*, pp. 15–34, Univelt Incorporated.

[8] Andreas Huster and Stephen M. Rock, "Relative position estimation for intervention-capable AUVs by fusing vision and inertial measurements," in *Proceedings of the 12th International Symposium on Unmanned Untethered Submersible Technology*, Durham, NH, August 2001, Autonomous Undersea Systems Institute.

[9] Arthur Gelb, Ed., *Applied Optimal Estimation*, The M.I.T. Press, Cambridge, Massachusetts, 1974.