

A Robotic Sentinel for Benthic Sampling along a Transect

Jnaneshwar Das and Gaurav S. Sukhatme

Abstract—This paper presents the design of a novel robotic system capable of long-term benthic sampling along a transect. The robot is built to traverse back and forth along a mechanical guide-rail at the bottom of a water body. We present results from localization tests with the robot in a laboratory tank and a shallow swimming pool. A pilot deployment was made at a marina to accurately observe the rate and direction of water flow across a section of the marina inlet. Results from the experiment demonstrate the potential of this platform for monitoring various aquatic phenomena of interest.

I. INTRODUCTION

Underwater observing systems are used for making physical, chemical and biological measurements of aquatic environments. Automating such systems can lead to significant savings in scientist time, while increasing the possibility of new discoveries in limnology and oceanography.

We introduce a novel *benthic robotic observing system* designed to periodically patrol a transect while capturing images of the water above it. We describe the electromechanical design of the robot and its propulsion and support system. Using a combination of inertial sensing, absolute position measurements at the transect endpoints, and a simple dynamic model, we are able to apply a Kalman smoother to obtain accurate position estimates for the robot. These estimates are used *a posteriori* to align sensor scans of the water made by the main observational instrument on the robot - an Acoustic Doppler Current Profiler (ADCP). The aligned scans produce a map of unprecedented accuracy and coverage (presently such measurements are typically made by manually lowering the ADCP or winching it down).

Our focus in this work is on the development of a system for long-term deployment. We envisage a robotic sentinel that is able to maintain a presence underwater for weeks or months at a time allowing seasonal-scale repeatable measurements. Mobility is energy expensive, and a long-term presence may naturally suggest a Lagrangian approach where the robot moves with the water mass (*e.g.*, a set of drifters or a glider). For increased precision and repeatability we considered an alternative approach wherein a guide-rail is used to mechanically support the motion of the robot as it moves to and fro between the transect endpoints. The advantages of this approach are discussed below.

Our design is energy-friendly since energy is needed only to move, not to hold station. It is also unobtrusive relative

to the water surface. Importantly, the approach is potentially highly *repeatable* and *precise*. The main contributions of this paper are 1. the design of the prototype robot, and 2. the experimental evidence, including data from a field experiment, in support of its precise and repeatable positioning ability, leading directly to the mapping of water column properties with unprecedented accuracy and repeatability.

Since this is our first report on this novel robot, we make the following assumptions (we plan to relax some of these in future work). For the purposes of the work reported here the guide-rail is rigid and has negligible slack. The yaw of the robot is mechanically restricted. The robot is not equipped with sensors to directly sense its precise location on the guide-rail. Finally, we do not envisage real-time, fine-grained location estimates being necessary for robot operation.

We utilize a state estimation technique that uses frequent measurements from inertial sensors, and infrequent, periodic absolute position information every time the robot reaches the end of the guide-rail. This is achieved by using a forward-backward Kalman filter and a smoother. Our results show that we are able to precisely localize the robot underwater using the smoother (root mean squared error $\sim 3\text{cm}$ on a traversed length $\sim 65\text{cm}$; 5% of the total span). A pilot deployment done at commercial marina demonstrates accurate observation of water flow rate and water flow direction across a section of the marina inlet.

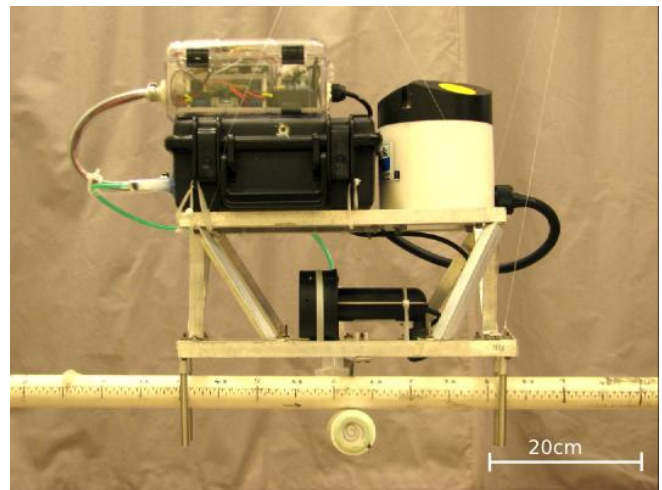


Fig. 1. The robot positioned on a PVC guide-rail.

Jnaneshwar Das (jnaneshd@usc.edu) and Gaurav S. Sukhatme (gaurav@usc.edu) are with the Robotic Embedded Systems Laboratory and Department of Computer Science at the University of Southern California (USC). This work was funded in part by NSF (grants CNS-0540420, CNS-0325875, CCR-0120778); ONR (grant N00014-08-1-0693); NOAA (grant NA05NOS4781228); and a gift from the Okawa Foundation.

There is significant interest and activity in the construction of large (continent-scale) earth observing systems. Prominent examples include the National Ecological Observatory Network (NEON) [1] (a network of observatories connected to-

gether), and the NEPTUNE ocean observatory [2] (a heavily instrumented, cabled ocean observing system to characterize information from the ocean and the ocean floor). NEON and NEPTUNE both envisage robots (tethered and free-swimming) as part of their design for semi-autonomous data gathering and sample collection.

Autonomous underwater vehicles (AUVs) are an obvious choice for underwater sampling because of their mobility, a good example being the REMUS system [3]. However, AUVs cannot operate for extended periods of time because of energy constraints. Autonomous sea gliders [4] are a more energy-friendly alternative, generating forward propulsion by varying the buoyancy, allowing them to span thousands of kilometers with a single charge cycle. More recent developments have resulted in gliders which harvest propulsive energy from the heat flow between the vehicle engine and the thermal gradient of the temperate and tropical ocean [5]. However both platforms essentially use dead-reckoning for state estimation (occasionally bounding the error with accurate absolute position information on resurfacing). Significant recent work on AUV autonomy is being done in other areas. In [6] two approaches are investigated: a geometric approach in which a mobile robot moves within a field of static nodes and all nodes are capable of estimating the range to their neighbours acoustically; and visual odometry from stereo cameras. The Networked Aquatic Microbial Observing Systems (NAMOS) [7] is an example of an Autonomous Surface Vehicle (ASV) collaborating with a static set of sensor nodes to form an observing system for large-scale aquatic sampling.

The localization technique exploited in this work is the forward-backward Kalman filter and the Kalman smoother which has found numerous applications in robotics. Examples include the 3D attitude estimation for localization of planetary rovers [8], and [9] to maintain acceptable precision of vehicle positioning even during GPS satellite blackout (e.g. when the vehicle enters a tunnel). A Kalman smoother has been used underwater for tasks such as object tracking [10].

Finally, we note that the work reported here owes a significant piece of its intellectual heritage to the Networked Infomechanical Systems (NIMS) project [11], [12] which has produced a set of robotic systems supported by cableway structures suspended within the environment. Various NIMS systems have been used to sense aspects of the forest canopy, a river confluence, and a lake. These systems have all featured a gondola carrying the sensor payload suspended from a cableway and a pulley mechanism for gondola mobility. In contrast, the system reported here hovers above a guide-rail due to buoyant force, and uses a thruster for propulsion.

II. SYSTEM DESCRIPTION

A. Packaging and Support Infrastructure

The components of the robot are packaged into two separate levels, each being an off-the-shelf drybox for underwater use (Figure 2). The upper level is a small (19 cm x 3.6 cm x 3.3 cm), transparent unit which houses the computing

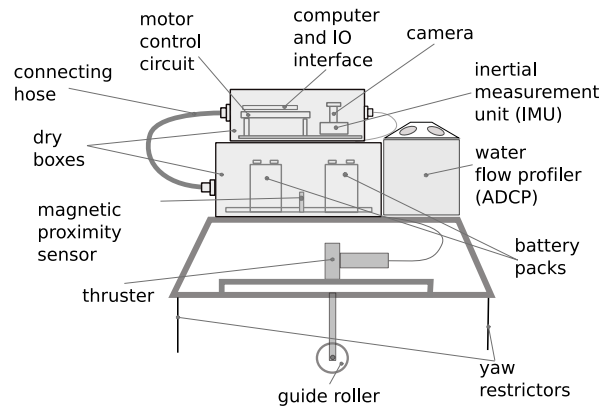


Fig. 2. Schematic of the robot

hardware, inertial measurement unit, power converter, safety fuses, motor control board and an upward looking camera. The lower level is a larger drybox (30.4 cm x 22.8 cm x 15.2 cm) which holds the batteries, magnetic power switches, and hall-effect sensor for marker detection. A communication and power link between the upper and the lower units is established via cables running through a tygon hose, clamped to the dry-boxes using tapped hose fittings. The dryboxes are fixed to a custom-made aluminum frame, on which the thruster is mounted. At the very bottom of this frame is a roller assembly that links the robot to the guide rail. Stainless steel guide rods at each end restrict yaw. The robot is powered by two 7Ah, 12V sealed lead-acid batteries housed within the lower deck. One battery is dedicated for the computing and sensing hardware, whereas the other battery is used by the thruster. A DC-DC voltage converter supplies the required voltages to the computer, motor control board and the sensor suite.

Hermetically sealed reed switch assemblies were mounted within the robot housing to serve as power switches. They are triggered externally with small neodymium magnets, allowing easy power cycling and emergency shutdown.

The guide-rail is a multi-segment PVC pipe, with embedded neodymium magnets at the endpoints to serve as markers. The pipe is mounted to supporting structures at either end using aluminum structural fittings. Figure 10 illustrates the deployment setup for an experiment at a marina.

B. Computing

The robot uses a Gumstix [13] 600MHz single board computer for all tasks. Additional expansion cards provide IO and storage functionalities (RS232, I²C, Ethernet, 802.11b and microSD). The software has been written using C and C++ to run on the Linux operating system.

C. Sensing

The robot's sensor suite consists of accelerometers, inclinometers, a proximity sensor, a camera, and an Acoustic Doppler Current Profiler (ADCP).

Inertial measurement unit: The MicroStrain 3DM-G Inertial Measurement Unit (IMU) with a tri-axial accelerometer, inclinometer and rate-gyro provides acceleration, angular velocity, and attitude information respectively. We obtain data from the IMU by polling it at a rate of 16Hz over the RS-232 link.

Proximity sensor: We use the Philips KMZ51 hall-effect sensor to detect magnetic markers embedded at the endpoints of the guide-rail. The sensor was mounted inside the lower dry-box and calibrated such that the robot reliably detected magnetic markers at a distance of 20 cm with a field of view of 6 cm. Figure 3 shows the readings from this sensor from tests of robot traversals in a pool.

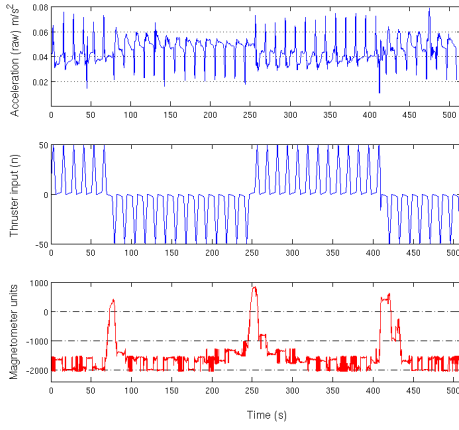


Fig. 3. A segment of the raw accelerometer readings, thruster input and magnetometer readings from a pool trial

Camera: The robot uses an upward looking digital camera to take pictures of the water surface. We plan to stitch these images together using position estimates from the robot’s offline state estimator to produce a visual-panorama of the water surface. The camera is interfaced to the Gumstix stack through a USB host interface.

Acoustic Doppler Current Profiler (ADCP): The Argonaut-XR manufactured by Sontek is an advanced doppler sonar for precise measurement of water velocity along three axes [14]. The ADCP was mounted on the robot pointing up to measure the water velocity at various depths. It is interfaced to the Gumstix stack over the RS232 interface.

D. Actuation

The robot is actuated using a single Seabotix BTD150 bidirectional underwater thruster. The thruster is controlled using a Roboteq AX500 motor control board which receives commands from the Gumstix stack over the RS232 interface.

E. Workflow and Control

We implemented the state transition sequence shown in Figure 4. Traversal in each direction is implemented as a series of ‘hops’, with a specified time interval between each hop. Open-loop control was chosen over other techniques

to keep the system simple (recall that the focus here is to test the accuracy at which we can reconstruct location estimates for the robot post-traverse). A triangular thruster actuation pattern was experimentally identified to be an ideal control primitive, since it allowed smooth traversal between locations.

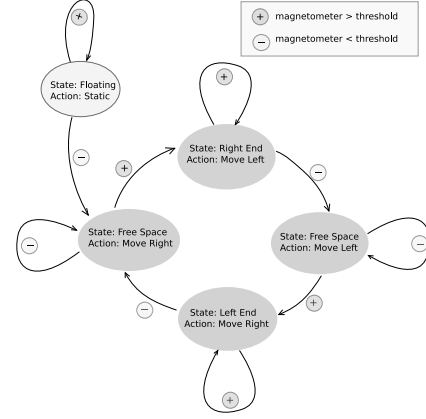


Fig. 4. The state transition diagram of the robot. The robot maintains internal state-action pairs which are updated based on magnetometer measurements. The location on the pipe is maintained as coarse belief states viz. right end, free space, and left end. The fine-grained locations are computed offline by the robot’s state estimator.

III. SYSTEM MODEL

Since the robot is constrained to a guide-rail, its movement is restricted to the x-axis, giving it only one translational degree of freedom. Additionally, by restricting yaw we eliminate one of the three rotational degrees of freedom. Rolling is practically negligible when environmental disturbances like underwater currents are absent. Hence, for all practical purposes the robot possess only two degrees of freedom: translation along the x-axis, and rotation around the pitch or y-axis. The free-body diagram shown in Figure 5 illustrates various forces acting on the robot. We neglect added mass (inertia added due to accelerating or decelerating fluid that moves along with the robot body), and assume hydrodynamic damping to be linear because of slow speeds involved. Since the center of buoyancy is above the center of mass, the robot experiences a restoring torque which tries to keep it upright.

The equations of motion of the system follow by balancing all forces and moments about the robot axis.

$$\ddot{x} = \frac{F \cos \phi - C_d \dot{x} - f - Ml\dot{\phi}^2 \sin \phi + Ml\ddot{\phi} \cos \phi}{M} \quad (1)$$

$$\ddot{\phi} = \frac{Mgl \sin \phi + F_b l_b \sin \phi + Ml\ddot{x} \cos \phi - F(l - l_{th})}{MI^2 + I} + \frac{F_d(l + l_d) \cos \phi}{MI^2 + I} \quad (2)$$

M is the robot mass, C_d is the hydrodynamic damping constant, ϕ is the pitch angle, F is the thruster force, l_d and l_{th} are the distances of the center of drag and point of thruster force from the center of mass. During experimental trials, the robot was observed to pitch by a maximum angle of 1.5°

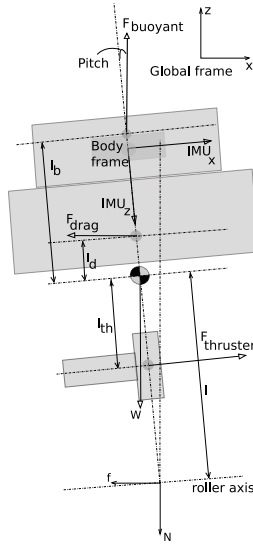


Fig. 5. Free body diagram for the system

from the mean orientation. Since this angle is quite small, the pitching effect can be ignored to obtain a simplified linear model given by the following equations,

$$M\ddot{x} = F - C_d\dot{x} \quad (3)$$

$$\ddot{x} = \frac{F}{M} - \frac{C_d}{M}\dot{x} \quad (4)$$

We model the robot as a discrete linear dynamical system in the state space, represented as

$$x_k = Ax_{k-1} + Bu_{k-1} \quad (5)$$

The state vector is $x_k = [s_k, v_k, a_k]^T$ where s_k , v_k and a_k are the position, velocity and acceleration at timestep k . The control input u_k is the force from the thruster which is given by

$$u_k = \phi(n_k) \quad (6)$$

ϕ is a function which maps the input command 'n' to a force. A is the state transition matrix and B is the control matrix. Using a first order solution to Equation 4 with a discretization of Δt , Equation (5) can be expanded as follows representing the simplified discrete dynamical model of the system used in the following section on location estimation.

$$\begin{bmatrix} s_k \\ v_k \\ a_k \end{bmatrix} = \begin{bmatrix} 1 & \Delta t & 0 \\ 0 & 1 & \Delta t \\ 0 & \frac{-C_d}{M} & 0 \end{bmatrix} \begin{bmatrix} s_{k-1} \\ v_{k-1} \\ a_{k-1} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{M} \end{bmatrix} u_{k-1} \quad (7)$$

The damping constant and the mass of the system were determined empirically to be $C_d = 17.15Ns/m$, and $M = 12.83kg$ respectively. We use a discretization $\Delta t = 0.12s$.

IV. LOCATION ESTIMATOR

We are interested in determining the position of the robot on the guide-rail as a function of time. Since our robot is a data-logging sentinel, we are not particularly interested in online estimates of the state. Instead, we want to compute the estimates offline from the logged sensor data, which could then be used to position-stamp the payload data. This allows

us to accurately align the ADCP flow velocity measurements and in the future will allow us stitch images together or generate spatio-temporal distributions of aquatic parameters of interest. Our goal is to use an estimation technique which would not require absolute sensors like wheel encoders since they do not perform reliably underwater (slip, no contact with a rigid surface, sensor failure). Instead, we fuse frequent data from internal sensors (accelerometers), and infrequent absolute position data available every time the robot reaches an endpoint on the guide rail. Since we want to obtain location estimates offline, we use a forward-backward Kalman filter and a smoother. A forward Kalman filter propagates forward in time, estimating the state of the robot at every timestep. Once the filter reaches the next absolute measurement (at the end of the guide-rail), a backward filter is run. This filter uses the control inputs and acceleration in a manner similar to the forward filter, but propagates backward in time. The backward filter eventually reaches the timestep where the forward filter started estimating. The smoother combines the estimates from these filters resulting in position estimates with reduced uncertainty.

A. The Kalman Filter

The Kalman filter is a recursive filter used extensively as a state estimator for dynamical systems in the presence of noisy measurements [15]. It follows a predictor-corrector cycle; first predicting the state of the robot using the process model, and then updating it with the measurements from the sensors. The discrete Kalman filter uses the state space given by,

$$x_k = Ax_{k-1} + Bu_{k-1} + q_{k-1} \quad (8)$$

$$y_k = Hx_k + r_k \quad (9)$$

where,

x_k is the state of the system at timestep k

y_k is the measurement at timestep k

u_{k-1} is control input at timestep $k-1$

$q_{k-1} \sim N(0, Q_{k-1})$ is the process noise at timestep $k-1$

$r_k \sim N(0, R_k)$ is the measurement noise at timestep k

A is the state transition matrix of the dynamical model

B is the control matrix

H is the measurement model matrix

Predict step:

$$x_k^- = Ax_{k-1} + Bu_{k-1} \quad (10)$$

$$P_k^- = AP_{k-1}A^T + Q_{k-1} \quad (11)$$

Update step:

$$v_k = y_k - Hx_k^- \quad (12)$$

$$K_k = \frac{P_k^- H^T}{HP_k^- H^T + R_k} \quad (13)$$

$$x_k = x_k^- + K_k v_k \quad (14)$$

$$P_k = P_k^- - K_k S_k K_k^T \quad (15)$$

where, x_k^- and P_k^- are the estimated mean and covariance of the state before seeing the measurement at timestep k , v_k is the measurement residual at timestep k , K_k is the filter gain, which tells how much the predictions should be corrected at timestep k , x_k and P_k are the estimated mean and covariance of the state respectively after seeing the measurement at timestep k .

B. Backward filter

The backward filter provides us with the optimal state estimates by propagating the system back in time starting from the last timestep, all the way to the first timestep. Equations 8 and 9 can be modified to propagate the system backwards in time as follows,

Predict step:

$$x_{k-1}^- = A^{-1}x_k - (A^{-1}B)u_{k-1} \quad (16)$$

$$P_k^- = A^{-1}P_{k-1}A^{-1T} + Q_{k-1} \quad (17)$$

Given the current state and the previous input, the process model propagates the system backward to determine the state at the previous timestep. The measurement model remains unchanged.

Update step: The update step remains the same, given by Equation 15. The backward filter estimates are the best closer to the last timestep and progressively increase in uncertainty.

C. Filter Design

The filter uses the discrete state space model for the robot (Equation 7) as the process model. Acceleration measurements from the IMU are corrupted with components of the gravity field. The corrected acceleration along the x-axis of the global frame is given by,

$$a_{true} = z_x \cos \theta + z_z \sin \theta \quad (18)$$

a_{true} is the true acceleration along the x axis in the global frame, z_x is the measured acceleration on the x axis of the robot body frame, z_z is the acceleration measured along the z axis of the robot body frame, and θ is the pitch angle of the robot in the global frame. The measurement model (Equation 9) is given by,

$$z_k = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_k \\ v_k \\ a_k \end{bmatrix} + r_k \quad (19)$$

z_k is the corrected measured acceleration along the global x-axis. The process and measurement noise covariances were empirically determined.

$$Q = \begin{bmatrix} \sigma_s^2 & 0 & 0 \\ 0 & \sigma_v^2 & 0 \\ 0 & 0 & \sigma_a^2 \end{bmatrix} = \begin{bmatrix} (0.00072)^2 & 0 & 0 \\ 0 & (0.006)^2 & 0 \\ 0 & 0 & (0.05)^2 \end{bmatrix} \quad (20)$$

$$R = \sigma_{imu}^2 = (0.03)^2 \quad (21)$$

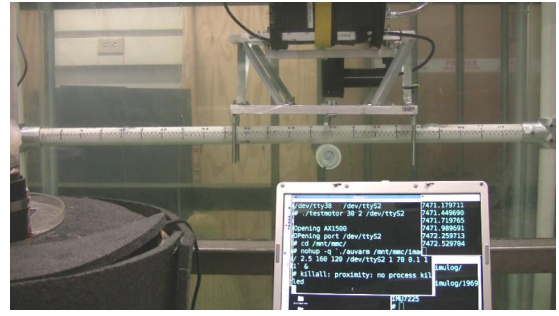


Fig. 6. The experimental tank setup

D. Smoother

The estimates from the forward and backward Kalman filters are combined using the smoother. It computes estimates with lower uncertainty by weighting the estimates from each filter with their covariances [16] as follows.

$$P_s = (P_f^{-1} + P_b^{-1})^{-1} \quad (22)$$

$$x_s = (P_f^{-1}x_f + P_b^{-1}x_b)P_s^{-1} \quad (23)$$

P_f , P_b and P_s are the covariances for the estimates from the forward filter, backward filter and the smoother respectively. x_f , x_b and x_s are the state estimates from the forward filter, backward filter and the smoother respectively.

V. EXPERIMENTAL RESULTS

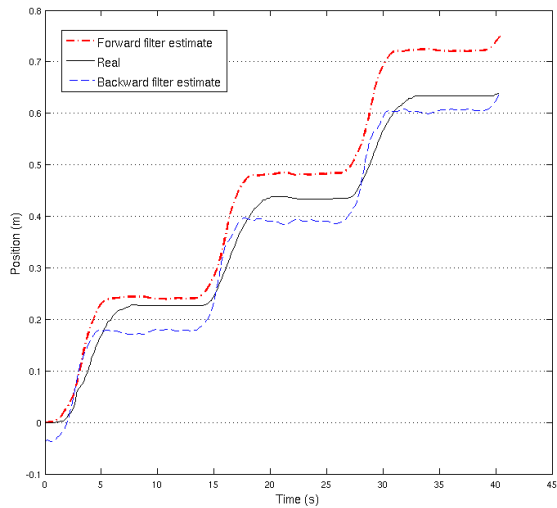
The system was tested at three locations, an indoor experimental tank with dimensions 2 m x 1.2 m x 2.3 m (Figure 9), a swimming pool (Figure 11), and a section of Redondo Beach Marina on the Southern California coastline.

We performed the tank experiments in three phases. The first phase involved the developmental iterations to test the robot structure, actuation, workflow and control. The second phase was to determine the process model, and the third phase was to implement and evaluate the location estimator.

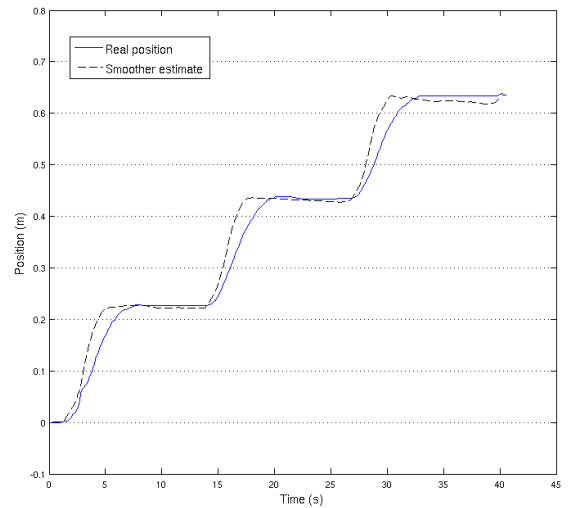
The robot was mounted on a graduated guide-rail supported by appropriate structural fittings. A directional wireless antenna outside the tank in close proximity to the tank wall made it possible to maintain a continuous communication link between the robot and a laptop external to the tank. All experiments were video recorded so that ground truth position information could be obtained by manually processing the imagery. The forward Kalman filter, the backward Kalman filter, and the smoother were run on the the data sets obtained from the tank experiments. Figure 7 shows the resulting location estimates for one of the trials.

The pool served as a larger testbed with a traversable span of 3.45 m. On an average, the robot needed thirteen thrusts to traverse the length. Images of the water surface was captured by the robot after every hop. A measuring tape with periodic markers was suspended on the surface as a source of images for the camera on the robot, and also provide the true position of the robot after each hop.

For the tank experiment, the root mean squared error from the forward filter was 0.042 m, the backward filter 0.044 m,

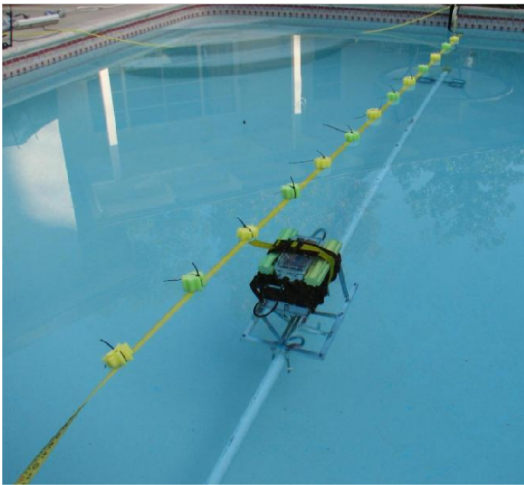


(a) Forward-backward filter estimates for a trial in the tank

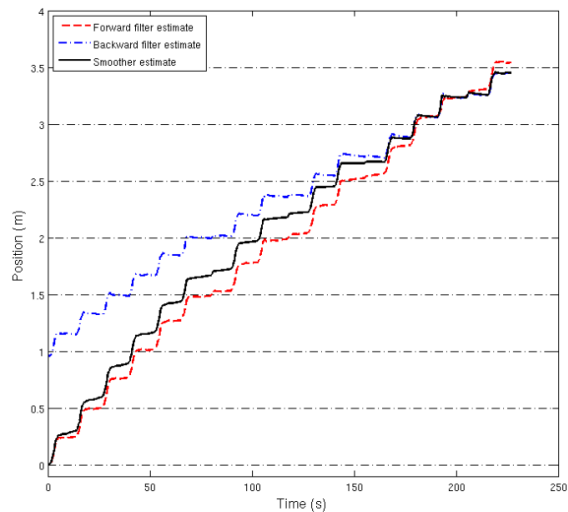


(b) Smoother estimates for the trial in the tank

Fig. 7. Results from the tank experiment



(a) The swimming pool setup



(b) Forward filter, backward filter and smoother estimates for the pool trial

Fig. 8. Pool experiment setup and results

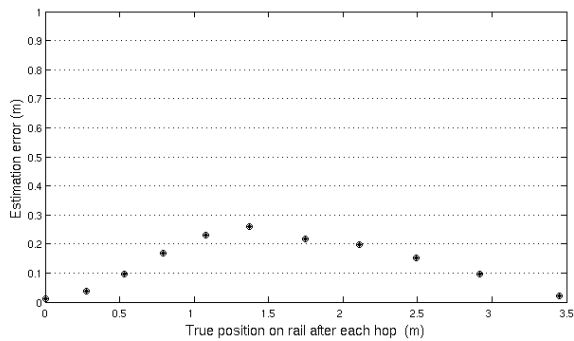
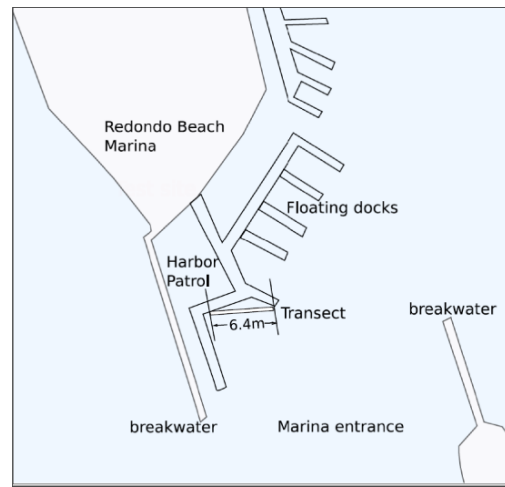


Fig. 9. The error in location estimates for each hop of a pool traversal

and the smoother 0.037 m. The estimation error after each hop of the pool experiment is shown in Figure 9. The error builds up and reaches its peak of 25cm close to the center of the rail and reduces as the robot reaches the endpoint. The pool error is considerably higher than in the tank experiment. This is not surprising since the estimator performance on longer transects is affected more by the cumulative error of the IMU. The robot also faces the additional problem of pitch in the guide-rail. The estimator was designed with the assumption of negligible pitch in the rail, but for long transects, the guide-rail develops considerable slack due to its weight ($\sim 5^\circ$ in the pool). With the lack of brakes on the original prototype, a small component of the positive buoyancy results in a mild drift when the robot is not actuating. The small accelerations for the drift is usually within the band of IMU noise, and hence not very effective



(a) Satellite image of Redondo Beach Marina. The test site is shown in detail in (b)



(b) The deployment plan for the marina trial

Fig. 11. The test site at Redondo Beach Marina

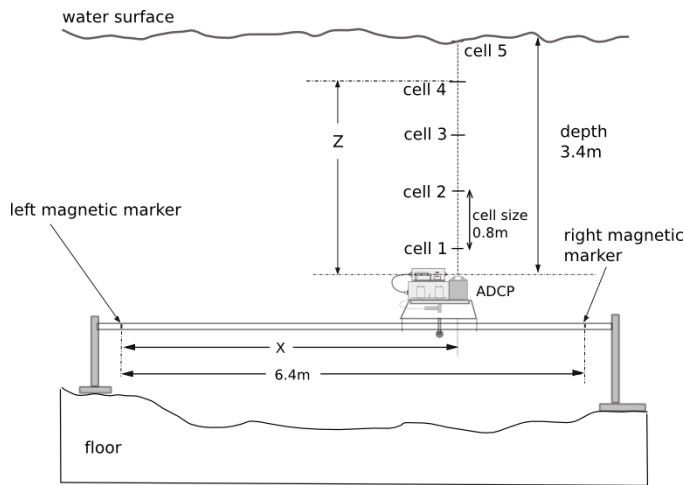


Fig. 10. A cross-sectional view of the deployment at Redondo Beach Marina, the ADCP sampling locations, and the corresponding flow data axes used for the visualizations in Figure 12

in the estimation process. Driven by the observations from the pool test, parking brakes were implemented and used in the tests at the marina. For future trials on long transects($\sim 40m$), we plan to use intermediate magnets on the guide-rail as landmarks to bound the error and improve the estimator performance. These magnets can be spaced approximately 4m apart since the observed maximum error of 25cm for the pool experiment is within acceptable range for aquatic sensing.

A pilot deployment was performed at Redondo Beach Marina to observe water flow at a section of the marina inlet (Figure 11). The system had a traversable span of 6.4 m and was deployed at a depth of 3.4m (Figure 10 shows a schematic in profile). The ADCP was programmed to record water flow at five depths. On average, the robot sampled seven points along the transect during each traversal. The ADCP data and the location estimates were then used

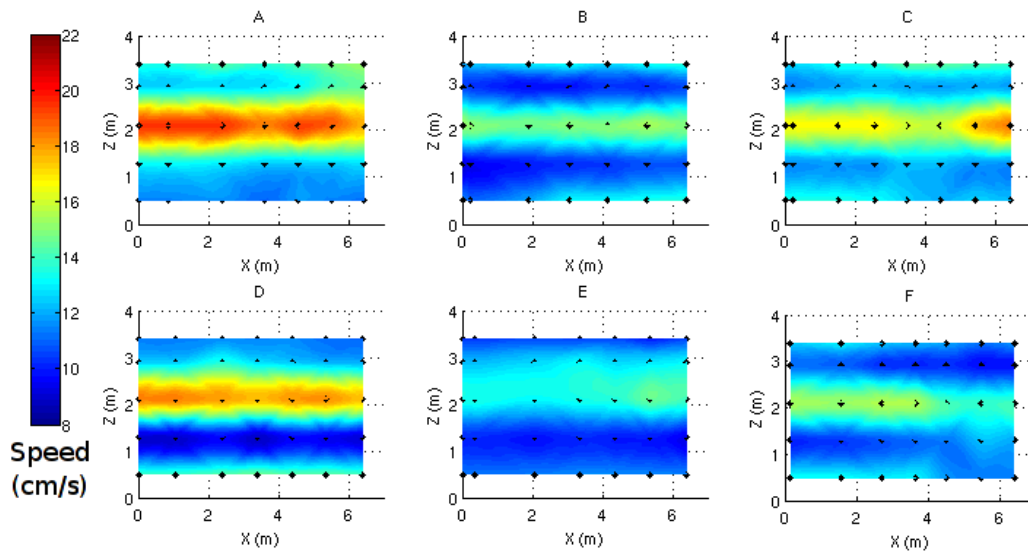
to generate a snapshot of the flow during each traversal. Data logged over a period of an hour were used to generate registered flow plots for six traversals approximately 8 minutes apart. Figure 12 shows the water flow speed and flow direction observed during the course of an hour-long experiment.

VI. CONCLUSION AND FUTURE WORK

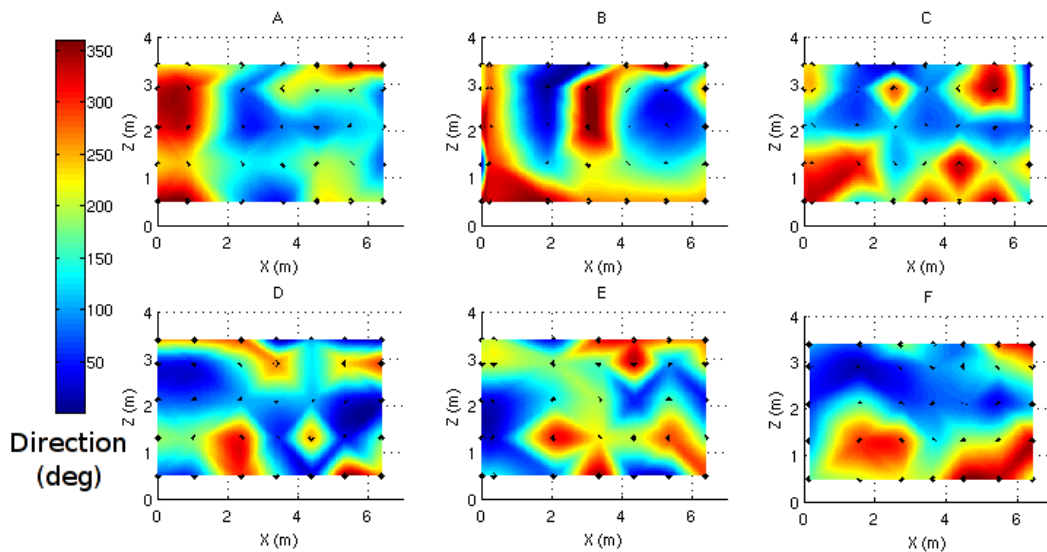
In this paper, we presented a novel underwater robotic system for benthic sampling along a transect. A state estimation approach that relies on frequent inertial measurements, and infrequent position updates was proposed and implemented. Controlled experiments performed at a tank and a pool show the accuracy of our approach. In preliminary field trials at a commercial marina we show unprecedented, accurate, water flow measurements using the system. We are actively working on long-term deployments in the field with the system as well as the ability to change its mission in real-time based on measured data.

REFERENCES

- [1] "National Ecological Observatory Network Project (NEON)." [Online]. Available: <http://www.nsf.gov/bio/neon/start.htm>
- [2] "North East Pacific Time-integrated Undersea Networked Experiments (NEPTUNE)." [Online]. Available: <http://www.neptune.washington.edu>
- [3] J. Bellingham, C. Goudey, T. Consi, J. Bales, D. Atwood, J. Leonard, and C. Chrysostomidis, "A second generation survey auv," *Proceedings of the Symposium on Autonomous Underwater Vehicle Technology (AUV '94)*, pp. 148–155, 19-20 Jul 1994.
- [4] R. E. Davis, C. C. Eriksen, and C. P. Jones, "Autonomous buoyancy-driven underwater gliders," In: *Griffiths, G. (ed), Technology and applications of autonomous underwater vehicles. Taylor and Francis, London*, pp. 37–58, 2003.
- [5] D. Webb, P. Simonetti, and C. Jones, "SLOCUM: an underwater glider propelled by environmental energy," in *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, Oct 2001, pp. 447–452.
- [6] P. Corke, C. Detweiler, M. Dunbabin, M. Hamilton, D. Rus, and I. Vasilescu, "Experiments with Underwater Robot Localization and Tracking," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'07)*, 10-14 April 2007, pp. 4556–4561.



(a) Flow speed



(b) Flow direction (on the horizontal plane)

Fig. 12. Plots showing water flow for six traversals on the marina transect. The traversals were spaced by 8 minutes, with a dwell time of 25 seconds at each location. The robot sampled 7 points on the transect in each traversal. The ADCP sampled flow data for 5 vertical points for each dwell point on the transect. The offline estimator was run to find the position of the robot for each flow datapoint and the data was then interpolated to generate the flow visualization.

- [7] G. S. Sukhatme, A. Dhariwal, B. Zhang, C. Oberg, B. Stauffer, and D. A. Caron, "The Design and Development of a Wireless Robotic Networked Aquatic Microbial Observing System," *Environmental Engineering Science*, vol. 24, no. 2, pp. 205–215, 2006.
- [8] S. Roumeliotis, G. Sukhatme, and G. Bekey, "Smoother based 3D Attitude Estimation for Mobile Robot Localization," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'99)*, vol. 3, 1999, pp. 1979–1986 vol.3.
- [9] D. Bouvet and G. Garcia, "Civil-engineering articulated vehicle localization: solutions to deal with GPS masking phases," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'00)*, vol. 4, pp. 3499–3504 vol.4, 2000.
- [10] R. B. Nicklas, "An Application of a Kalman Filter Fixed Interval Smoothing Algorithm to Underwater Target Tracking," Master's thesis, 1989.
- [11] P. H. Borgstrom, M. J. Stealey, M. A. Batalin, and W. J. Kaiser, "NIMS3D: A Novel Rapidly Deployable Robot for 3-Dimensional Applications," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'06)*, Oct. 2006, pp. 3628–3635.
- [12] B. Jordan, M. Batalin, and W. Kaiser, "NIMS RD: A Rapidly Deployable Cable Based Robot," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'07)*, 10-14 April 2007, pp. 144–150.
- [13] "Gumstix Inc." [Online]. Available: <http://gumstix.com/>
- [14] "Argonaut-XR Current Meter." [Online]. Available: <http://www.sontek.com/argonaut-xr-current-meter.htm>
- [15] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice*. Prentice-Hall, 1993.
- [16] S. Reynolds, "Fixed interval smoothing: Revisited," *Journal of Guidance*, vol. 13, no. 5, 1990.