A REAL-TIME VISUAL MOSAICKING AND NAVIGATION SYSTEM

Kristof Richmond[1], **Stephen M. Rock**[1,2]
{kristof,rock}@stanford.edu

[1]**Aerospace Robotics Laboratory, Stanford University**
Durand Bldg, Rm 028, 496 Lomita Mall, Stanford, CA 94305-4035

[2]**Monterey Bay Aquarium Research Institute**
7700 Sandholt Road, Moss Landing, CA 95039-9644

Abstract

A real-time visual mosaicking and navigation system for use near the sea floor has been developed and deployed as a pilot aid on MBARI ROVs. This system provides high-precision, environment-relative vehicle positioning and control without the use of external positioning arrays. The system uses a live video feed from a camera on the ROV combined with velocity data from an acoustic Doppler velocity log (DVL) to build and display in real time a mosaic depicting the sea floor beneath the vehicle. The mosaic is composed of individual video frames (tiles) snapped at appropriate intervals and placed in a larger composite image. The tile location is determined by image disparities from frame-to-frame image correlation combined with DVL navigation data. The result is a visual map showing the current vehicle position, which is robust to visual outages, such as dust clouds or periods out of visual range of the seafloor. This map can be used by the ROV pilot to navigate the seafloor environment, either via direct joystick control of the vehicle or through an automatic control interface allowing a point of interest in the mosaic to be selected and autonomously moved to. To date, this system has been used as a pilot aid for ROV operation. With some extensions, it has the potential to be deployed on AUVs as well.

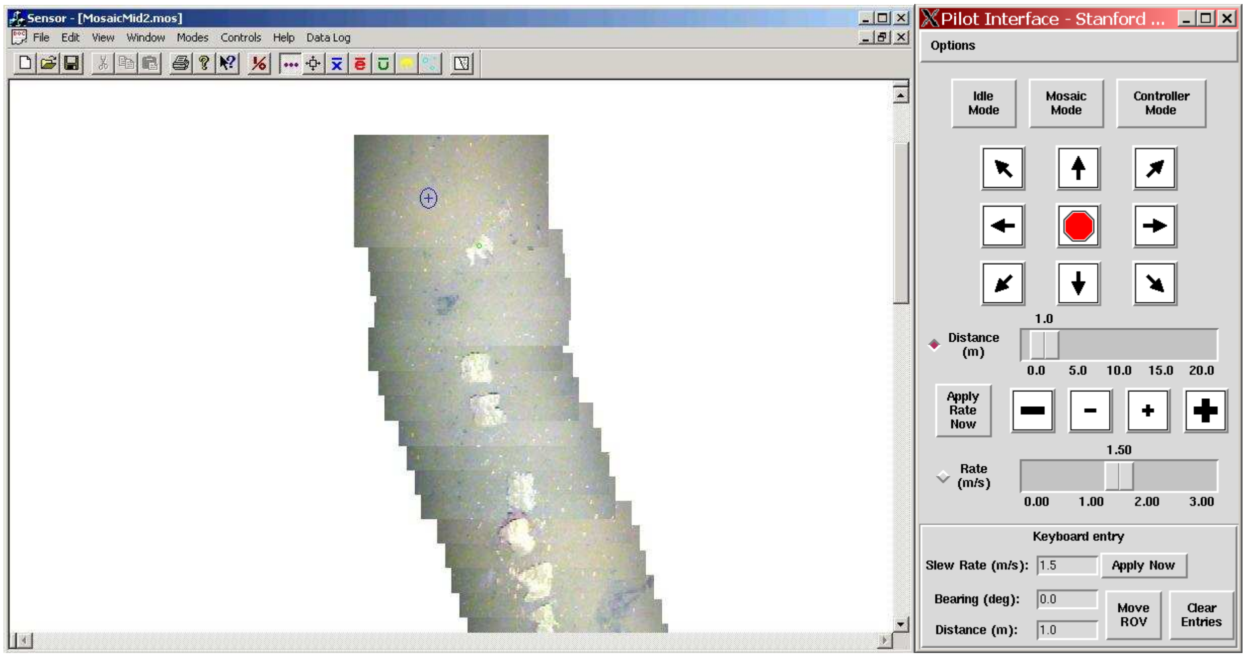
This paper reviews the high-speed vision algorithms used to calculate in real time camera displacement in the challenging visual environment at the sea floor (documented in part in previous papers) and the use of these algorithms in an automatic stationkeeping and online mosaicking system for visual odometry. The paper then focuses on new results from continued development of the system. Issues encountered during field testing are presented, along with the methods developed to address these issues. In particular, the incorporation of DVL measurements is described. These measurements provide a means to compute vehicle position during vision outages and allow the online mosaic to be continued when vision is restored. Using this complementary sensor has not only increased the robustness of the system to outages in the primary sensor, but has also improved general vehicle control by providing direct velocity feedback to the vision sensor. The combined system provides direct measurements with respect to the vehicle environment along with an intuitive and information-laden display for human users.

1 Introduction

The Stanford Aerospace Robotics Lab (ARL), in cooperation with the Monterey Bay Aquarium Research Institute (MBARI), has developed, demonstrated and deployed a real-time visual mosaicking and navigation system on the ROV *Ventana* for use as a pilot aid. The system provides high-precision, environment-relative vehicle positioning and control without the use of external positioning arrays. The user interface (see Figure 1) provides the user with an evolving view of the area being explored by the vehicle and the capability to control the vehicle relative to the environment. The goal is to provide a real-time navigation system enabling direct interaction with objects in the local environment through an intuitive, information-laden interface.

This paper reviews the high-speed vision algorithms used to calculate in real time camera displacement in the challenging visual environment at the sea floor. These algorithms form the core of the visual sensor which creates the real-time mosaics. The camera displacements from the vision sensor are then merged with measurements from other sensors on the vehicle to calculate the full vehicle state. The computed state can be used to provide automatic position control of the vehicle.

Extensive field testing showed that the visual sensor alone was not sufficiently robust for regular use in both



Distance (m) 1.0

Rate (m/s) 1.50

Slew Rate (m/s): 1.5

Bearing (deg): 0.0

Distance (m): 1.0

Figure 1: The user interface to the visual navigation system. On the left, the visual mosaic and is displayed in real time as the vehicle maneuvers over the sea floor. Current vehicle position is indicated by the crosshair. When automatic control is engaged, vehicle navigation is enabled by selecting a desired position on the mosaic with a touch on a touch screen monitor or by selecting a bump displacement via the arrow-button interface on the right.

mosaicking and navigation. It became apparent that, during the course of normal operation, vision outages such as dust clouds are not uncommon. In addition, it is restrictive to constantly maintain an altitude low enough to have clear sight of the bottom. These considerations limited the practical use of the system.

In order to overcome these limitations, a DVL was incorporated into the system. Use of the DVL navigation solution allows the system to track position through periods of obscured vision. The direct measurement of velocity made by the DVL has also improved the estimate of vehicle state and thus control. Addition of the DVL has rendered the system robust enough to enable deployment as a practical pilot aid for benthic navigation and control.

1.1 Background

Visual mosaicking—stitching together snapshots to produce larger composite images—is a powerful tool for benthic exploration [10]. The limited propagation of light in water constrains the possible size of single-frame images that can be acquired. In order to gain a large-scale overview of the seafloor, a composite mosaic image must be constructed.

Usually, the desired product is a mosaic optimized for viewing: images are shuffled, equalized, warped, and blended in a batch process after the vehicle has surveyed the site in order to come up with the most consistent, accurate picture possible. The process is often seeded with navigation information from an LBL array or DVL unit.

The system described in this paper, however, uses vision for real-time feedback to a vehicle user and/or automatic control system. Here, the focus is on providing immediate feedback on the position of the vehicle, sacrificing some of the accuracy of a fully-optimized, batch-processed mosaic.

The use of vision as a position sensor on the sea floor was pursued early on by Marks and Fleischer [2, 7]. The algorithms they developed form the core of the system described in this paper. Their work has also been extended by Garcia, et. al. into a Kalman filter framework demonstrated in a test tank [4]. Additional work in using vision as the *sole* sensor to measure the complete vehicle state has been pursued by Gracias, et. al. [5] and Negahdaripour, et. al. [8].

The automated visual mosaicking system described in this paper builds mosaics of the seafloor incrementally in real time.

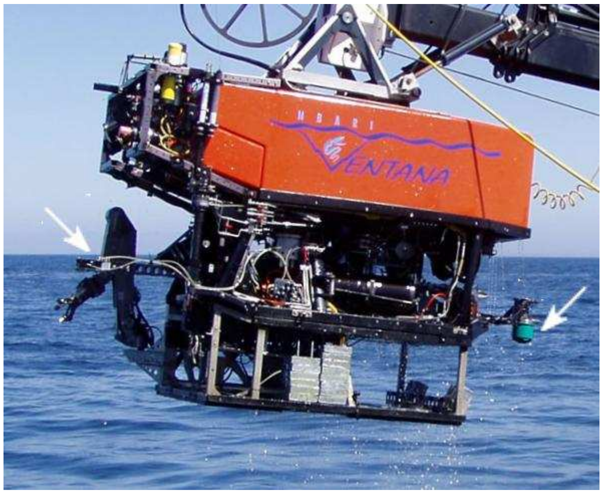


Figure 2: The MBARI ROV *Ventana*. Arrows highlight the primary sensors used in the real-time mosaicking and navigation system: a downward-looking video camera mounted on an arm at the front of the vehicle, and a DVL mounted at the rear.

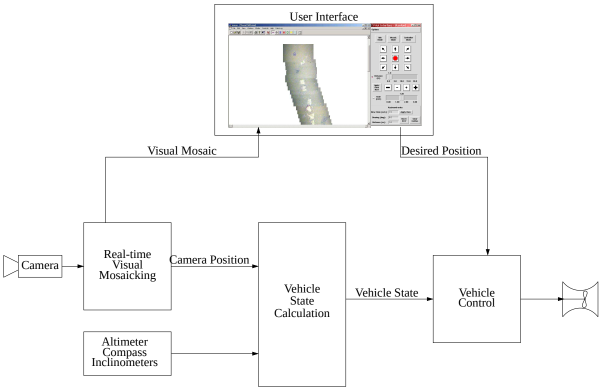


Figure 3: Block diagram of the system. A continuous video feed from a downward-looking camera is passed into the visual mosaicking component. The computed camera position is passed to the vehicle state component and combined with other sensor measurements to obtain the full vehicle state. This is then passed to the control component to use for navigation and control of the vehicle thrusters. Activity of the components is monitored and controlled from a graphical user interface showing the current real-time mosaic.

2 System Overview

The visual navigation and control system can be divided into three main components: the real-time visual mosaicking component, the vehicle state calculation component, and the vehicle control component. Figure 3 shows how these components communicate with each other and with the user interface.

The system has been developed and demonstrated primarily on the ROV *Ventana* (Figure 2). All three components have been interfaced with the telemetry and control system of this vehicle. The mosaicking component can also be used standalone. In this case, its interface consists solely of an incoming video feed and an outgoing computer display. This has facilitated the demonstration of real-time mosaicking on other vehicles, such as the MBARI ROV *Tiburon* (e.g. Figure 8), and the ROV *Triton* of the Santa Clara University Robotic Systems Laboratory [6].

2.1 Real-Time Visual Mosaicking

The primary sensor for the system is the real-time visual mosaicking component. It takes as input a live video stream from a downward-looking camera. On *Ventana* this is generally an Insite Pacific, Inc. Orion, a color, variable zoom camera. This camera is mounted on an arm at the front of the vehicle in order to provide a good view of the bottom and be out of the way of other tools and operations at the front of the vehicle (see Figure 2). The system has also been used with a down-sampled signal from the main Sony HDTV camera tilted down to be perpendicular to the sea floor and various other cameras.

The visual mosaicking component has been described in detail elsewhere [3, 7]. Briefly, live images from the camera are continuously processed and compared to a previously-obtained reference image. The relative offset between the live and reference images indicates the camera location relative to the reference image. When the camera moves far enough from the center of the reference image, the current live image is automatically added to the mosaic and becomes the new reference image. The first image seen when the system is initialized is the initial reference and is taken as the origin of the mosaic coordinate system.

The reference and live images are first filtered using the signum of Laplacian of Gaussian operation [9], which extracts textures in the images. The visual offset is then calculated via a fast sum-of-xor correlation. Only two parameters of the motion are calculated, namely the translation in the plane parallel to the bottom. The remaining components of the motion are taken from other sensors, as explained in Section 2.2. Constraining the potential motion to two dimensions greatly increases the speed of the correlation computation.

The location of the 2D correlation peak gives the translational offset, and its magnitude is a direct function of the variance in the offset measurement. This maximum correlation value is referred to as the correlation confidence. It is thresholded at an empirically-determined value to determine whether the system has visual lock [1].

The vehicle position is then computed by summing the relative offsets between all reference images starting with the origin. This position is displayed on the user interface as a crosshair overlay on the mosaic. If a new reference image has been snapped, the mosaic is first updated with the new tile before displaying the position.

Visual offsets can be calculated at the frame rate of 30 Hz. Depending on the speed and altitude of the vehicle, and the seafloor visibility, the sampling of new live images can be slowed down. As explained in Section 3, on *Ventana*, the vision component is generally run at 5 Hz. At this sample rate, visual offsets and real-time mosaics are still reliably produced, (given the restrictions mentioned in Section 2.4).

Position error growth in the system is unbounded in practice. Though some work has been done to try to recognize crossover in the vehicle path and correct for error accumulated around the loop [1], it has not yet been implemented in the field. Thus, as the vehicle moves around in the environment, the visual mosaicking component currently deployed has steady error growth in the position calculation. Error growth is on the same order as DVL navigation solutions (see Section 4).

2.2 Vehicle State Computation

The position measurements produced by the vision component, while useful for producing mosaics, are not enough to provide navigation and control capability. As with any monocular solution, the scale of the measurement cannot be determined from vision alone. The measurement is thus appropriately scaled using an altimeter reading and the field-of-view parameters of the camera.

The remaining elements of the full 6-DOF vehicle position vector are taken from the vehicle altimeter and inclinometers. This vector can also be differentiated to determine vehicle velocities for vehicle control.

2.3 Navigation and Control

Once the full vehicle state is calculated, it is passed to the control component. Here it is compared to a desired state provided by the user interface and this error signal feeds a PD controller.

Automatic control of the vehicle can be enabled or disabled. This allows the system to be used either in “piggyback” mode—providing the user with visual feedback as the vehicle is flown under joystick control—or in automatic mode to control the vehicle. In the latter mode, the user interacts with the vehicle via higher-level positioning commands through the user interface (see Figure 1), in a manner similar to GPS-based dynamic positioning systems for surface vessels. Intuitive, position-level control is enabled by providing a touch-screen interface to the real-time mosaic. The user can see the environment around the vehicle and readily move to locations of interest simply by touching the mosaic. An interface to enable displacements of known length and direction is also provided. Additionally, the commands from the controller are summed with pilot joystick commands, so that a pilot can always intervene if necessary.

2.4 Limitations

The visual navigation system described above was demonstrated on several field trials. While collection and display of small real-time mosaics and control of the vehicle for stationkeeping for limited periods of time (∼10 min) were generally possible, the system suffered from two major limitations.

First, during practical operations, it is not always possible to maintain visual lock on the sea floor. Visual occlusions, such as dust clouds, unexpectedly come into the field of view. In addition, in order to see the bottom consistently, the vehicle is required to operate at a limited altitude (∼2 m for *Ventana*). However, mission parameters may require periods of time out of this visual range of the sea floor, necessitating a vision outage. Such outages are generally indicated by low correlation and vision loss-of-lock. They render the vision component useless for their duration.

Second, the noise in the vision measurement is significant. Variations in the image offsets are compounded by multiplicative noise in the altimeter measurement which is used to scale them. Differencing these noisy position measurements to obtain the velocity components of the vehicle state exacerbates the problem.

3 Incorporation of a DVL

In order to address the limitations of the visual mosaicking and navigation system described in Section 2, a Doppler velocity log (DVL) was integrated into the system. It complements well the vision sensor. First, it operates in a complementary medium (sound *vs*. electromagnetic), and thus is not necessarily subject to the same occlusions and disruptions as vision. Second, it measures a different, yet related quantity (velocity *vs*. position). This means that it can not only augment estimates in those states which have the least information from vision—namely the velocities—but it can also fill in position measurements with integrated velocities during periods of vision loss-of-lock. This allows the real-

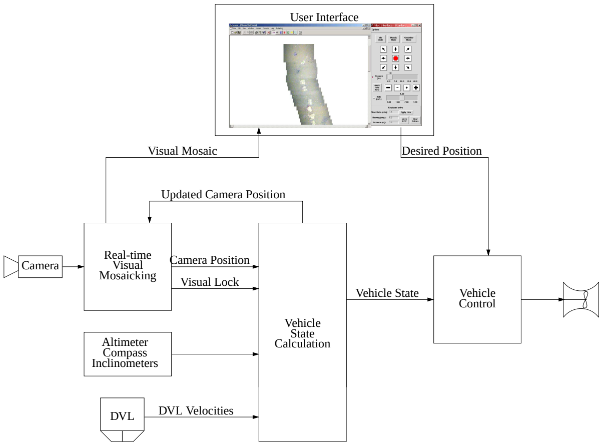


Figure 4: Block diagram of the system with DVL incorporated into the system. The vehicle state computation was extended to merge in the measurements from the complimentary sensor. This required additional knowledge of the vision lock state. Finally, the merged position is fed back to the visual mosaicking component to allow continuation of the mosaic from the current location after periods of loss-of-lock.

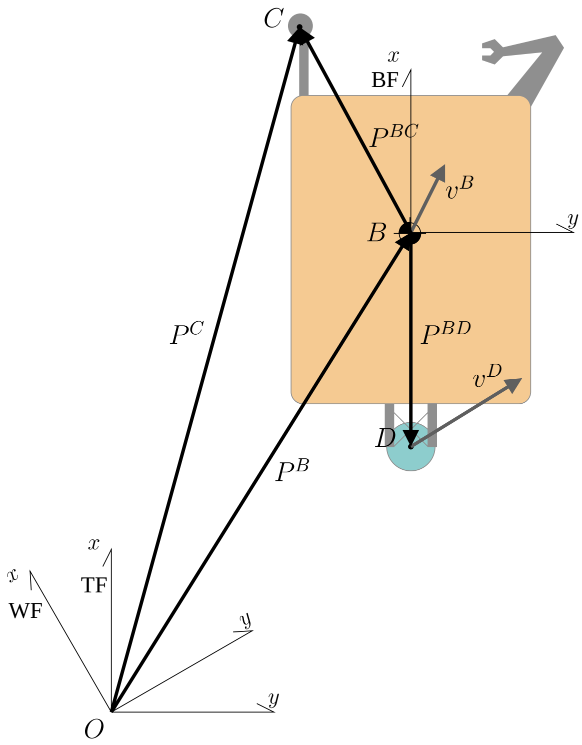


Figure 5: Reference frames, points and vectors used in Section 3. TF: Terrain frame. WF: World frame. BF: Body frame. O: Origin of mosaic (inertial). B: Vehicle center of mass. C: Location of downward-looking camera. D: Location of DVL. Position vectors are indicated in black, velocity vectors in gray.

time mosaic to be continued from an updated estimate of position when vision lock is restored. Finally, DVLs are often included in the standard suite of navigation sensors for underwater vehicles, as is the case for *Ventana*.

The *Ventana* DVL is a 1200 KHz RDI Workhorse Navigator. It is mounted at the rear of the vehicle, about 1 m up from the base (see Figure 2). This mounting location was chosen to minimize interference with the manipulators, cameras and tools at the front of the vehicle. Recent demonstrations were also successful when substituting a 300 KHz Workhorse unit for the standard 1200 KHz unit, while operating in the same regime ($\sim$2 m off bottom).

The DVL measurements are merged with the vision component measurements in the vehicle state computation component (see Figure 4). The DVL has a maximum update rate of 5 Hz. In order to avoid the complications of a multi-rate system, the vision component is slowed from 30 Hz (frame rate) to 5 Hz, and all computation proceeds at this rate. The remainder of this section explains in more detail how the vehicle state is computed.

3.1 Definitions

Figure 5 gives an overview of the frames of reference, points in space, and vector quantities used in the following discussion. The follow notation is used: a superscript following a vector indicates to which point in the vehicle the measurement applies (e.g. v^D is the velocity of the DVL, P^{BC} is the position vector from the center of mass to the camera), a leading superscript indicates in which reference frame a vector is expressed (e.g. ${}^{TF}P$ is a position coordinatized in the terrain frame), a following subscript indicates which sensor is the source of a particular measurement (e.g. P_V is a position as measured by vision). Notation for rotation matrices is an exception: ${}^{AF}R_{BF}$ denotes the rotation bringing a vector expressed in the B frame into the A frame. Generally, the minimal number of sub- and superscripts needed to clarify the discussion will be used.

The frames are defined as follows. All navigation occurs in a limited area, so a flat, inertial earth is assumed. The terrain frame (TF) has its origin where the first image in the mosaic was taken. Its axes are aligned with the image axes, with the z-axis pointing vertically down into the sea floor, which latter is assumed to be horizontal. The terrain frame is the frame in which the variables used in navigation and control are expressed. The world frame (WF) has the same origin and z-axis as the terrain frame, but is rotated so the x-axis is aligned with true north. Finally, the body frame (BF) has its origin at the vehicle center of mass, with standard aircraft coordinate axes (x forward, z down).

In general, all velocities are taken with respect to an

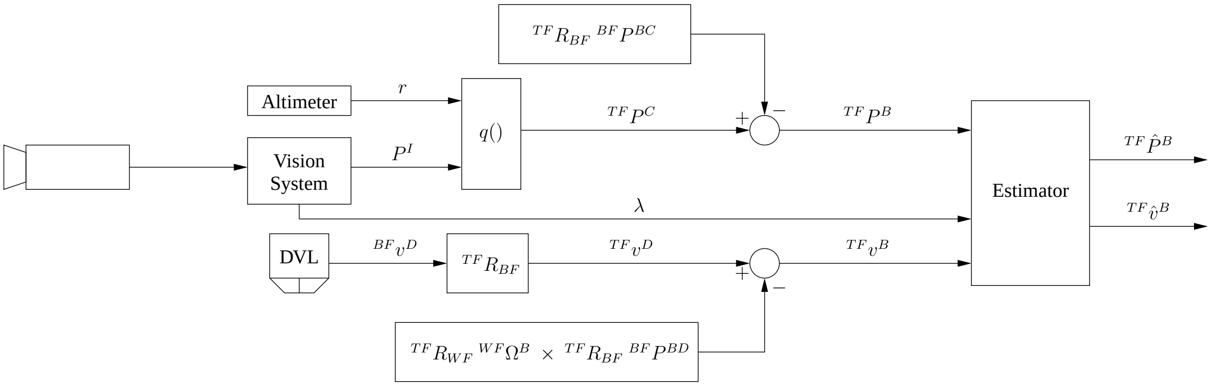


Figure 6: Transformation of sensor measurements to common frame of reference.

inertial frame.

3.2 Fusion

The vehicle state computation requires that the measurements from the two sensors be brought into the same frame and that they have the same scale. Figure 6 gives an overview of this process.

The two sensors measure two different quantities. The vision system calculates the position, P^I, of the current image in the mosaic. As explained in Section 2.2, this is scaled using the altimeter measurement, r, to determine the camera location,

$$P^C = q(P^I, r). \tag{1}$$

Here $q()$ maps positions in the image to positions in the world,

$$q(P, r) = \begin{bmatrix} \alpha_1 p_x r \\ \alpha_2 p_y r \\ r \end{bmatrix}, \tag{2}$$

where α_1 and α_2 are constant scaling factors given by the camera field of view.

The DVL calculates its velocity, v^D from the Doppler returns of the four beams. Given P^C and v^D, and given the known positions of the sensors on the vehicle and the vehicle angular velocity from the angle sensors (compass and inclinometers), the vehicle state can be found.

$$v^B = v^D - \Omega^B \times P^{BD}, \tag{3}$$

$$P^B = P^C - P^{BC}. \tag{4}$$

For control, it is convenient to express the motion of the vehicle in the terrain frame. Thus, for calculation, all quantities must be expressed in this frame. Given the angle sensors, the rotations to bring all vectors into the same frame of expression can be easily found, so that the full computation of vehicle state becomes

$$\begin{aligned} {}^{TF}v^B = {}&{}^{TF}R_{BF}{}^{BF}v^D \\ &- {}^{TF}R_{WF}\ {}^{WF}\Omega^B \times {}^{TF}R_{BF}\ {}^{BF}P^{BD}, \end{aligned} \tag{5}$$

$${}^{TF}P^B = q\left(P^I, r\right) \ - {}^{TF}R_{BF}\ {}^{BF}P^{BC}. \tag{6}$$

These quantities can now be fused in an estimator to provide a continuous position and velocity measurement during periods of loss of vision lock. Figure 7 presents this process. As explained in Section 2.1, the vision system returns an indication of its confidence in its measurement, λ. In this discussion, λ is normalized to lie in the unit interval, with 0 indicating no confidence and 1 indicating complete confidence.

At a time step k, the change in vehicle position ΔP is then computed from the position measurement derived from vision, P_V, and the velocity measurement derived from the DVL, v_D,

$$\Delta P_k = \lambda\left(P_{V_k} - P_{V_{k-1}}\right) + (1-\lambda)v_{D_k}T, \tag{7}$$

where T is the sample time. The fused vehicle position, $\hat{P}$ is then

$$\hat{P}_k = \sum_{i=1}^{k} \Delta P_i. \tag{8}$$

In practice, $\lambda \in \{0, 1\}$ and its value is taken from the indication of vision loss-of-lock. Thus the estimator becomes a switch between the two sensors. This implementation has the advantage of simplicity and is quite robust. In addition, before merging, the vision measurement, P_V, is scaled by a parameter, η, which adaptively matches the vision and DVL measurements to account for changes in the vehicle configuration and other unmodeled factors.

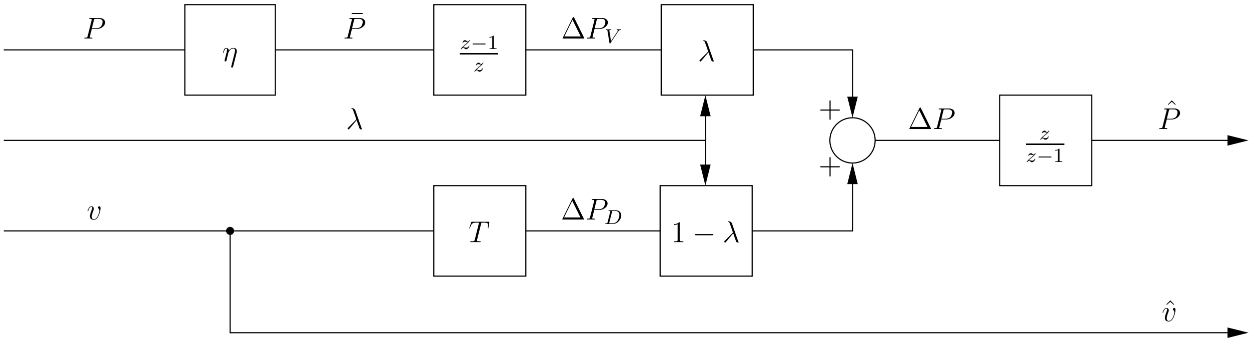


Figure 7: Inside the estimator: computation of vehicle state from two sensor measurements. For clarity, superscripts are dropped in this diagram.

4 Results

Real-time mosaicking has been demonstrated on a variety of platforms and in a number of environments. The full navigation and control system has been demonstrated on *Ventana*.

Figure 8 shows a mosaic generated using the video feed from an overhead pilot camera on the ROV *Tiburon* while flying over a whalefall in Monterey Bay. This mosaic was created without a DVL, and care had to be taken to keep the vehicle in a narrow layer close enough to the bottom to maintain visual lock and far enough to avoid jutting bones. Work is underway to interface our navigation system with the telemetry and control system on this vehicle to enable communication with the DVL and to implement automatic control.

Figure 9 shows a mosaic created from *Ventana* using vision and DVL. The pilot was asked to fly a box maneuver, returning to the point of origin. For one leg of the box, the pilot lifted the vehicle out of visual range of the sea floor, and vehicle position computation switched to using the DVL. After returning close to the bottom, the system re-acquired vision lock, switched back to vision position updates and the pilot flew back to the starting point, using the real-time mosaic as a guide.

Figure 10 shows another box maneuver mosaic, this time created using the automatic control functionality. After initializing the system and engaging automatic control, the pilot moved the vehicle away from the starting position using the arrow-button interface. To move back to an object of interest near the starting position (the feature pointed to in the mosaic), the pilot tapped the desired location on the touch screen displaying the mosaic. Vision lock was maintained throughout. Actual accumulated error, as measured by the offset in the feature, is 0.36 m or about 2% of distance traveled. This agrees well with the DRMS error (63% probability) of 0.35 m predicted based on correlation confidences.

In addition to allowing the real-time mosaic to be continued through periods of vision loss-of-lock, the DVL has also improved control by providing a clean velocity signal. Figure 11 compares the DVL velocity with that computed by differencing the vision position signal. The reduction in velocity noise has allowed more damping to be applied in the controller, resulting in smoother control.

5 Conclusion and Future Work

The system presented in this paper is a field-demonstrated system for visual navigation near the sea floor. It provides real-time mosaicking capability, which gives the user visual feedback on the vehicle position and the relative position of objects in local environment. It is not dependent on external navigation arrays, only on instruments carried on board the vehicle. To increase the robustness of the sensing to vision outages, a DVL has been incorporated into the system. This has also improved the general state estimate by providing a direct velocity measurement, which has in turn improved automatic control. The result is a system which has been deployed as a pilot aid on MBARI ROVs.

Future work will concentrate on bounding error growth. The fact that error growth in visual and DVL odometry is similar suggests that a better estimate can be obtained by appropriately mixing the measurements in an optimal estimator. This is one step to reducing errors, but greater gains can be achieved by effectively recognizing and taking advantage of crossover in the vehicle path to zero out the error around loops. Effectively recognizing such loops in mosaic data and updating the mosaic in real time will help ensure that the vehicle can always return to previously visited locations. Such a system can provide precise, environment-relative positioning and task-level command capability for vehicles with increased autonomy.

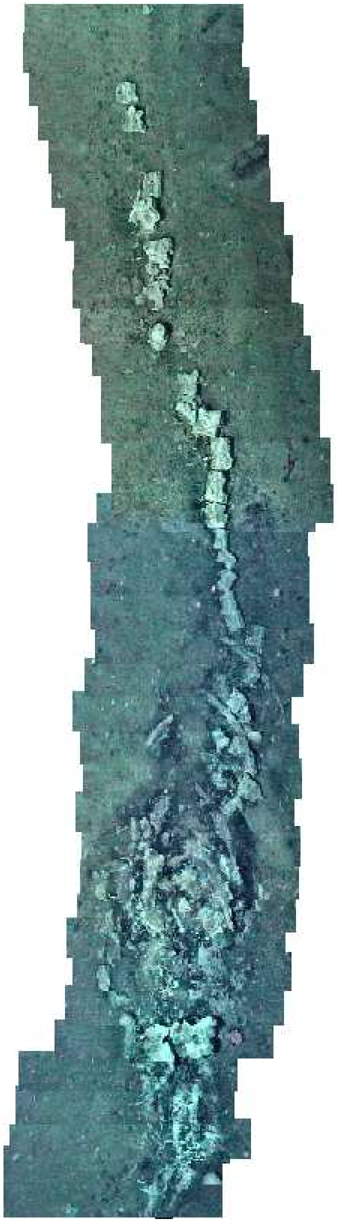


Figure 8: Mosaic of whalefall on the floor of Monterey Bay. The mosaic was produced in real time from the MBARI ROV *Tiburon* as it flew over the skeleton under pilot control. It has not been post-processed. Skeleton length is approx. 30 m. Courtesy R. Vrijenhoek, MBARI.

References

[1] S. D. Fleischer. *Bounded-Error Vision-Based Navigation of Autonomous Underwater Vehicles*. PhD thesis, Stanford University, Stanford, CA 94305, May 2000.

[2] S. D. Fleischer, R. L. Marks, S. M. Rock, and M. J. Lee. Improved Real-Time Video Mosaicking of the Ocean Floor. In *Proceedings of the OCEANS 95 Conference*, pages 1935–1944, San Diego, CA, October 1995. MTS/IEEE.

[3] S. D. Fleischer, H. H. Wang, S. M. Rock, and M. J. Lee. Video Mosaicking Along Arbitrary Vehicle Paths. In *Proceedings of the Symposium on Autonomous Underwater Vehicle Technology*, pages 293–299, Monterey, CA, June 1996. OES/IEEE.

[4] R. Garcia, J. Puig, P. Ridao, and X. Cufi. Augmented state Kalman filtering for AUV navigation. In *International Conference on Robotics and Automation, ICRA 2002*, Washington, DC, May 2002. IEEE.

[5] N. R. Gracias, S. van der Zwaan, A. Bernardino, and J. Santos-Victor. Mosaic-based navigation for autonomous underwater vehicles. *IEEE Journal of Oceanic Engineering*, 28(4):609–624, Oct. 2003.

[6] C. T. Lopez, R. A. Schweickert, M. M. Lahren, J. Howle, C. Kitts, J. M. Ota, and B. Richards. Submarine geology within the western part of Lake Tahoe, California. In *Geological Society of America Annual Meeting*, Denver, Colorado, Nov. 2004. GSA.

[7] R. Marks, S. Rock, and M. Lee. Real-time video mosaicking of the ocean floor. *IEEE Journal of Oceanic Engineering*, 20(3):229–241, July 1995.

[8] S. Negahdaripour and P. Firoozfam. Positioning and photo-mosaicking with long image sequences; comparison of selected methods. In *Oceans 2001*, volume 4, pages 2584–2592. MTS/IEEE, Nov. 5–8 2001.

[9] H. K. Nishihara. Practical realtime imaging stereo matcher. *Optical Engineering*, 23(5):536–545, Oct. 1984. Also in *Readings in Computer Vision: issues, problems, principles, and paradigms*.

[10] H. Singh, J. Howland, D. Yoerger, and L. Whitcomb. Quantitative photomosaicking of underwater imagery. In *Oceans 1998*, volume 1, pages 263–266. IEEE, Sept. 29–Oct. 31 1998.

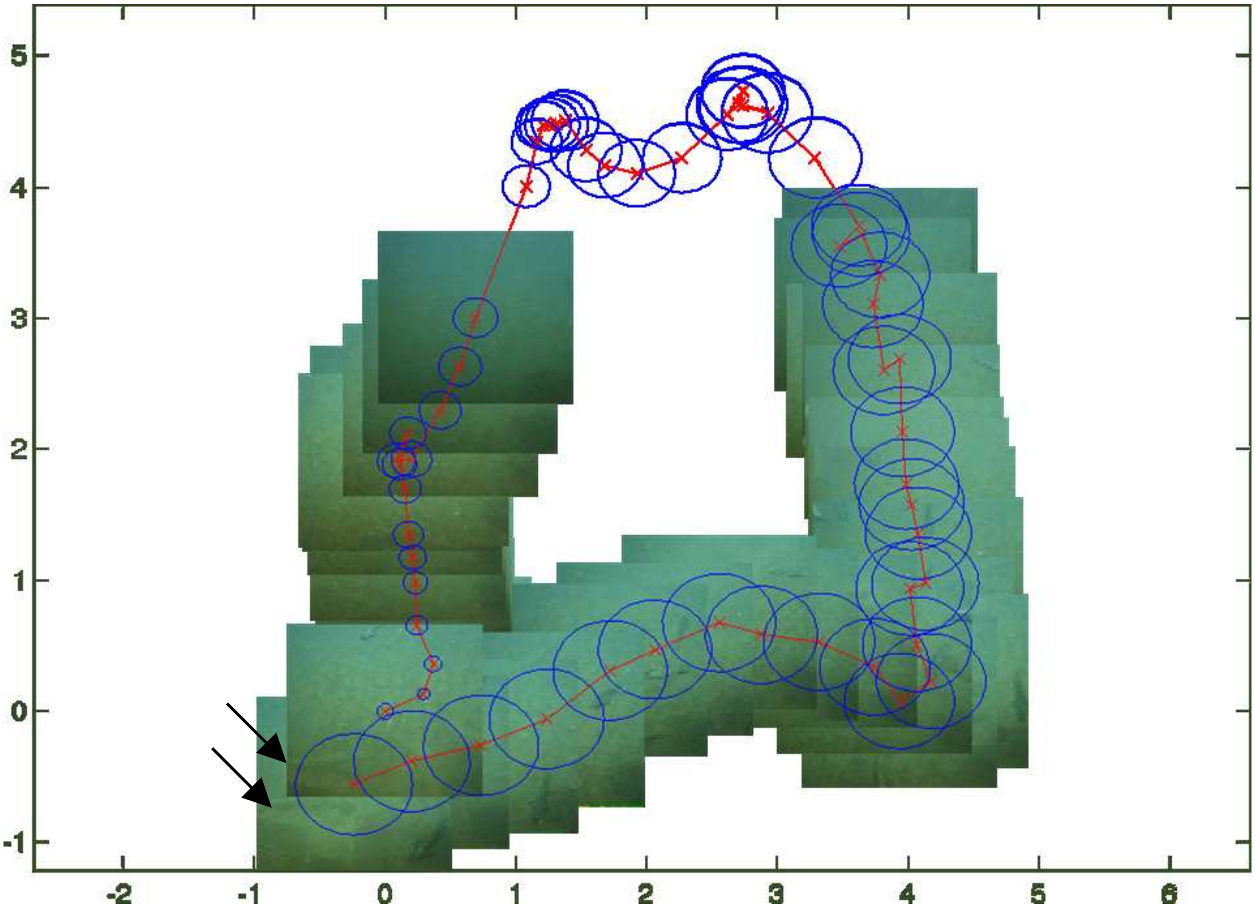


Figure 9: “Box maneuver” mosaic created with a period of vision loss-of-lock. The vehicle started at $(0, 0)$. The pilot then flew a rectangular trajectory in the clockwise direction. The vehicle trajectory as calculated by the system is shown in red. The blue ellipses indicate the computed 1-σ error in the position estimate. The portion of the trajectory without image tiles is the region where visual lock was lost. The initial image tile is brought to the foreground to show the location of a seafloor feature in the starting and ending tiles (black arrows). Scale is in m.

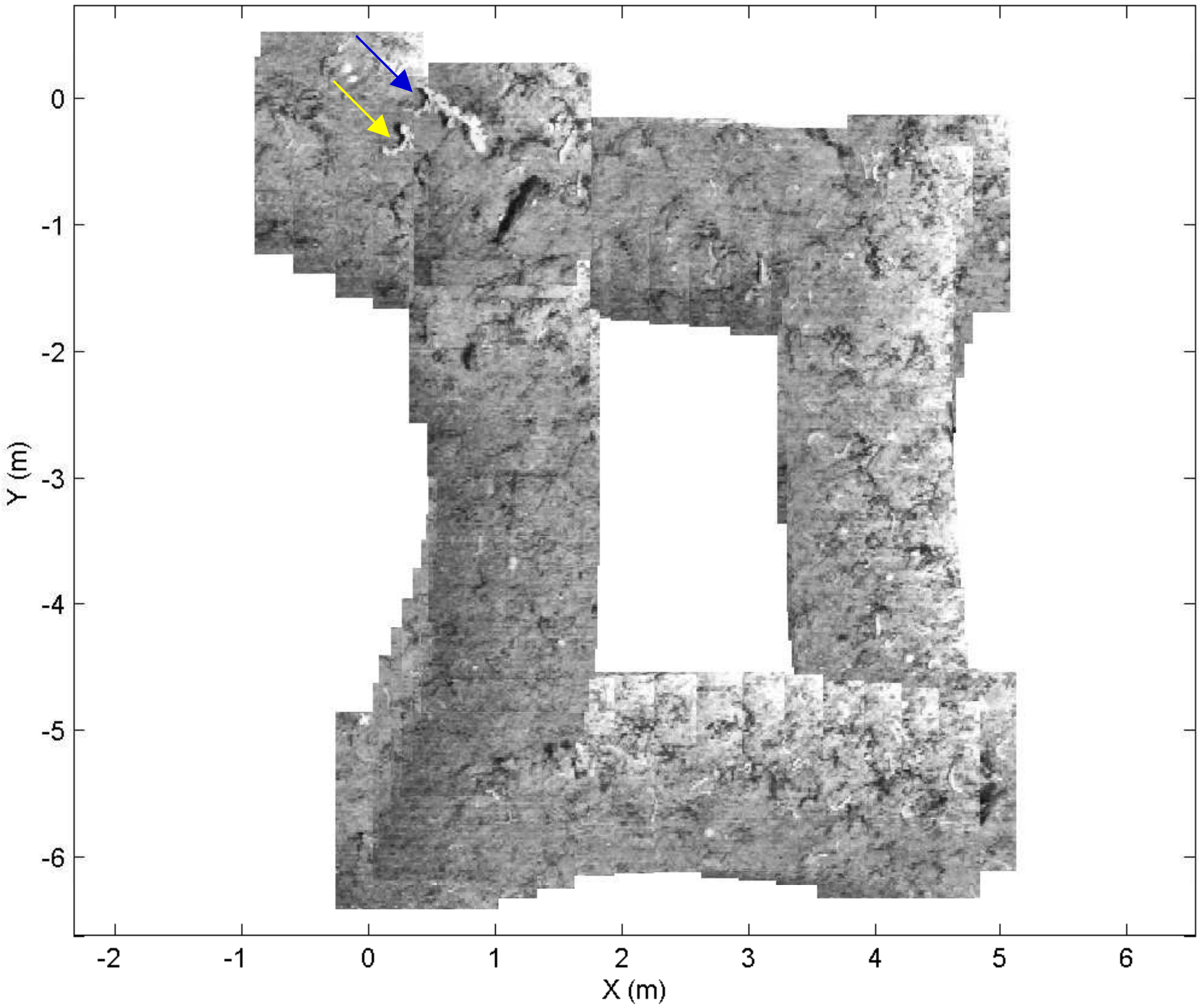


Figure 10: “Box maneuver” mosaic created under automatic control. The vehicle started at $(0, 0)$. The user interface was used to bump the vehicle around in the clockwise direction, and then to return to an object of interest seen near the beginning (yellow arrow). The feature is seen again in the final image tiles (blue arrow). Error is about 2% of distance traveled. Vision lock was maintained throughout. Image is monochrome as a color camera was not available on this day. Scale is incorrect as intrinsic parameters are not known for the camera used.

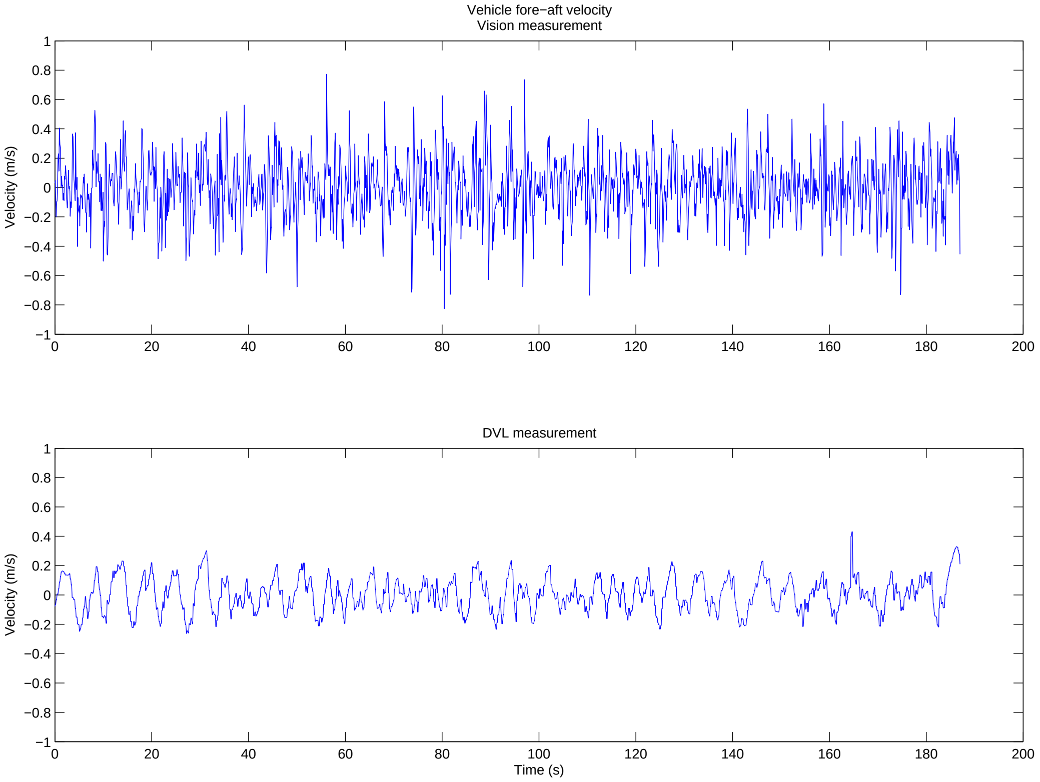


Figure 11: Comparison of velocity time series from the two sensors. Incorporating velocity measurements from the DVL into the vehicle state resulted in improved control.