Adaptable Sensor Fusion Using Multiple Kalman Filters

Louis Drolet[1,2], François Michaud[2] and Jean Côté[1]

[1]Robotics Division - IREQ
Institut de recherche d'Hydro-Québec
Varennes (Québec) Canada J3X 1S1
drolet@gel.usherb.ca, jean@robot.ireq.ca

[2]Department of Elec. & Comp. Engineering
Université de Sherbrooke
Sherbrooke (Québec) Canada J1K 2R1
michaudf@gel.usherb.ca

Abstract

This paper presents an innovative sensor fusion strategy for the positioning of an underwater ROV. The use of multiple Kalman filters makes the system highly adaptable by allowing different combinations of sensors without any modification of the models. This algorithm can handle any number of redundant sensors by using multi-filter fusion and can work asynchronously with different sensor data rates through a filter switching process.

1. Introduction

As a part of the Hydro-Québec dam security program, an underwater remotely operated vehicle (ROV) has been under development for the past few years at the Institut de recherche d'Hydro-Québec (IREQ). Such a ROV may be used in various inspection tasks taking place in underwater environments. The main purpose of this vehicle is the inspection of dam surfaces. Over the years, the exposed surfaces of concrete dams suffer from fatigue and the attack of the elements. Precise monitoring of the state of the dams is of extreme importance to avoid major failures that could have catastrophic consequences.

As the ROV inspects the dam, cracks or defaults on the surface must be identified using the on-board camera, precisely locating and reproducing them in a virtual environment consisting of a complete scaled model of the dam [5]. Later on, maintenance crews will want to return to the identified defaults to follow their evolution in time or to perform restoration work. To perform such tasks, the ROV must be equipped with a precise navigation system, giving at all time its exact position relative to the dam.

The ROV navigation system consists of two parts : a variety of sensors giving information on the vehicle's position and movement, and a navigation algorithm. The group of sensors may include an acoustic positioning system, accelerometers, gyroscopes, rate sensors, inclinometers, Doppler velocity sensors, and any other sensor contributing to the positioning of the vehicle. The navigation algorithm takes input from the different sensors and evaluates the best possible estimate of the vehicle's pose. The pose is expressed by the *x*, *y* and *z* coordinates of the vehicle as well as its orientation in terms of *roll*, *pitch* and *yaw*. Sensor fusion is used to combine information from a number of sensors to obtain a more precise evaluation of important ROV parameters.

To meet the needs of the ROV designers and users, the navigation system must be easy to use and adaptable. The users of the vehicle may, at different times, equip the ROV with different combinations of sensors, according to what they have in hand or to carry out specific inspection missions. The navigation algorithm must be adaptable in that it must be able to fuse information from these different combinations of sensors, without requiring any modification of the algorithm itself.

In order to accomplish this, we propose an approach that allows addition and removal of sensors from the navigation system by the selection of an appropriate model from a bank of Kalman filters. The use of a bank of filters in the Kalman filtering process has been suggested before for the purposes of sensor failure accommodation [7] or adaptive Kalman filtering [6]. Inspired by these approaches, we have designed a multiple Kalman filter algorithm to accommodate different combinations of sensors. One innovative aspect of our approach is that it accepts asynchronous information from the sensors. In real systems, all sensors do not send data at the same rate. Our approach includes a filter switching process so that only the newest data is used to evaluate the new pose estimate. Our approach can also combine information from any number of redundant sensors.

The paper is organized as follows. Section 2 describes the Kalman filter, which is the most widely used estimator in sensor fusion. However, the Kalman filter also suffers from certain shortcomings. Section 3 presents the multiple Kalman filter sensor fusion strategy we have designed to overcome these limitations, followed in Section 4 by results obtained using this new approach.

2. Traditional Kalman Filter

The Kalman filter, explained in detail in [4], is an optimal linear estimator based on an iterative and recursive process. It is used in a wide variety of applications, and applies particularly well to sensor fusion. It recursively evaluates an optimal estimate of the state of a linear system. In our case, the state of the system is the pose of the vehicle. At each iteration of the filter, a new estimate of the state (pose) is evaluated, using the new information (measurements) available to the filter. The Kalman filter process consists of two sub-processes, repeated iteratively at a time step of *dt*: the time update and the measurement update.

An easy way to explain the Kalman filter is to say that it incorporates two sets of sensor measurements, one in each sub-process. In the time update process, a *prior* estimate $X_{prior}(k)$ is computed based on the previous state estimate $X(k\text{-}1)$ and sensors indirectly related to the state (e.g. accelerometers or velocity sensors when the state is position, also called dead-reckoning sensors). Then, in the measurement update process, this prior estimate is blended with direct measurements of the state (position) coming from other sensors, thus obtaining the new updated state estimate $X(k)$.

2.1 Time Update Process

As described by Equations 1 and 2, the optimal estimate from the previous iteration, noted $X(k\text{-}1)$ is projected in time through the state transition matrix *A*, and the noisy inputs $U(k)$ (indirect sensors) are fed to the system through matrix *B*, relating the inputs to the state.

$$X_{prior}(k) = A \cdot X(k-1) + B \cdot U(k) \quad (1)$$

$$P_{prior}(k) = A \cdot P(k-1) \cdot A' + Q \quad (2)$$

To illustrate, we take as an example our ROV navigation system. In Equation 1, the state *X* is the linear position of the vehicle and *U* is the input from a linear velocity sensor. Matrices *A* and *B* reflect the cinematic of the system, *A* being equal to 1 and *B* to *dt*, the time interval between iterations. Equation 2 projects in time the error covariance matrix *P*, representing the variance of the error on the estimate *X*. *Q* is the covariance matrix associated with the process noise from the measurements $U(k)$.

2.2 Measurement Update Process

This process is formalized by Equations 3 to 5, in which direct noisy state measurements $Z(k)$ coming from sensors are compared with the prior state estimate $X_{prior}(k)$, yielding a correction to apply to this prior estimate to obtain the new estimate $X(k)$. Matrix *H* relates the measurements to the state.

$$K = P_{prior}(k) \cdot H' \cdot (H \cdot P_{prior}(k) \cdot H' + R)^{-1} \quad (3)$$

$$X(k) = X_{prior}(k) + K \cdot (Z(k) - H \cdot X_{prior}(k)) \quad (4)$$

$$P(k) = (I - K \cdot H) \cdot P_{prior}(k) \quad (5)$$

The importance of each estimation (the prior estimate $X_{prior}(k)$ and the measurement $Z(k)$) is determined by the Kalman gain *K*. This gain is in turn determined by matrices *Q* and *R*, which respectively represent the process noise covariance (indirect measurements) and the measurement noise covariance (direct measurements). The Kalman gain *K* takes a value between 0 and 1, 0 representing the use of the indirect measurements only, 1 the direct measurements only. The error covariance matrix *P*, modified in Equation 2, is again corrected in Equation 5 to reflect the measurement update process.

Continuing with the previous example, $Z(k)$ in Equation 4 represents a measurement coming from a position sensor and *H* is equal to 1 since there is direct correspondence between *Z* and *X*.

2.3 Strengths and Weaknesses

The Kalman filter is popular in sensor fusion applications because its formulation makes it easily adaptable to sensor fusion. As illustrated in the previous example, dead-reckoning sensors (velocity or acceleration) can be blended with sensors measuring direct position to give an optimal estimate. This combination brings out the good sides of all the sensors. This is illustrated through Figures 1 to 3, showing the position estimate errors over time for different combinations of noisy sensors. Figure 1 shows the position error resulting from the use of an accelerometer only. The noise from the accelerometer is accumulated over every iteration because the acceleration is integrated to obtain position. This results in a diverging position error and therefore this sensor is unusable over long time periods. Figure 2 shows the position error of a typical direct position sensor, such as the GPS or an SBL system (an acoustic positioning system often used in underwater applications). These systems are subject to intermittence and have larger noise levels over short time periods. However, their position estimates do not drift over time (the error is zero-mean), which makes them more reliable on the long run. Figure 3 shows the position error from the Kalman filter combining both of these sensors. It takes the absence of drift from the position sensor and the smoothness from the accelerometer, to obtain the best of both worlds.

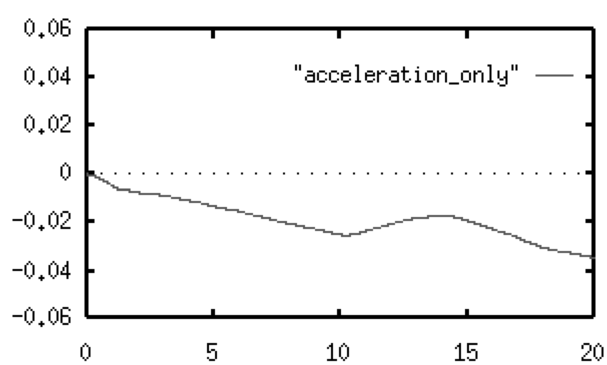

Figure 1 : Position error using only an accelerometer.

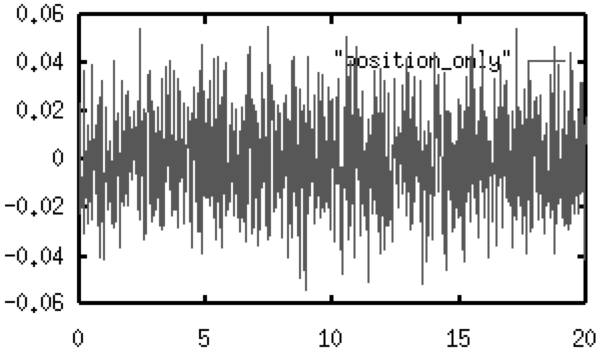

Figure 2 : Position error using only position.

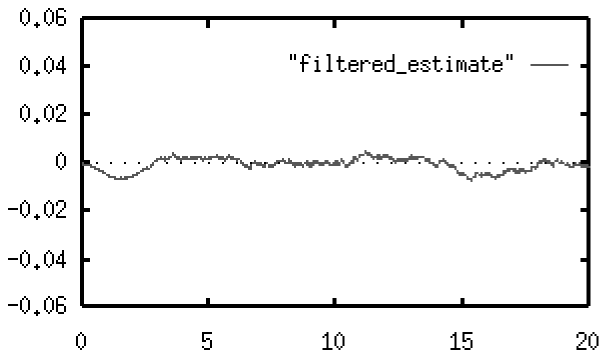

Figure 3 : Position error using the Kalman filter.

However, the Kalman filter works well within a certain framework, and under certain conditions. In some applications such as ours, a Kalman filter alone cannot meet the requirements and needs of the users of the ROV. In our case, one requirement is that it must be possible to add and remove sensors from the vehicle without any modification of the navigation algorithm. The Kalman filter does not allow this kind of flexibility because it models one specific system. Once the model represented in the filter matrices has been defined to accommodate certain sensors, all those sensors and no others must give information at each iteration of the filter. Another problem that cannot be handled by the traditional Kalman filter is sensor intermittence. If one of the sensors does not provide new information for the current iteration, the output of the filter can be corrupted.

3. Multiple Kalman Filter Strategy

Our multiple Kalman filter strategy is a simple and adaptable sensor fusion algorithm. The idea is to use a bank of Kalman filters to represent various combinations of sensors. A model selection process chooses the appropriate filters that model the real system. The approach also allows asynchronous inputs from the sensors by switching between filters according to sensor data availability. This way, we overcome the two major problems of the Kalman filter described in the previous section.

The number of distinct Kalman filters in the filter bank is determined based on the following characteristics:

• **Number of parameters to perceive**. For the ROV navigation system, five parameters are required : position in Cartesian coordinates; linear velocity; linear acceleration; orientation in Euler angles or any other representation; and angular velocity. Various sensors can be used on the ROV, but all of the positioning related sensors that can be installed on the vehicle give, directly or with minor transformations, a measure of one of these parameters. If the distinction between linear and angular measures could be dropped, this number would be reduced to three : position, velocity and acceleration.

• **Number of dimensions represented by each filter**. The position of the ROV is given by a vector in six dimensions : three representing the linear position along each of the *x*, *y* and *z* Cartesian axes and three giving the orientation of the vehicle in the Euler angles of *roll*, *pitch* and *yaw*. Sensors can measure any of the five parameters, in any number of the six dimensions. Designing a filter for every possible combination of parameters and number of dimensions would yield an impracticably high number of complicated filters. For this reason, we decided that each of the six dimensions should be filtered separately. Each filter works in only one dimension.

• **Transformations applied to the data before sending it to the filter**. Data from particular sensors require some transformations before being sent to the Kalman filters. For example, a sonar system giving a range and an angle should be transformed into Cartesian coordinates, and an inertial rate sensor should be corrected to remove the influence of the Earth's rotation. Furthermore, to simplify and reduce the number of Kalman filters, we decided that all filters would compute the pose in the same reference frame, namely a global reference frame having its origin at some fixed point in the environment. Therefore, some sensor measurements require coordinate transformations before being incorporated to the Kalman filters. For instance, a set of accelerometers fixed to the vehicle have to be corrected with the known orientation of the vehicle. All of these transformations take place outside of the Kalman filters themselves. They are part of a prior acquisition and transformation process.

These characteristics of our approach reduce the Kalman filter models to the simplest form of the cinematic equation (given in Equation 6 with p as the position, v as the velocity, a as the acceleration and dt as the time interval between iterations), and therefore all filters fusing the same parameters, in any dimension, are identical. There is no difference between fusing Cartesian position and linear velocity in x and fusing orientation and angular velocity in pitch. The same filter can then be used.

$$p(k) = p(k-1) + v(k)\cdot dt + a(k)\cdot dt^2/2 \qquad (6)$$

Taking all of this into consideration, only seven different filter models are required. All sensor combinations fall within one of the seven following one-dimensional models : position-only ; velocity-only ; acceleration-only ; position-velocity ; position-acceleration; velocity-acceleration; position-velocity-acceleration. Note that the first three models are not complete Kalman filters, since there is only one sensor (and therefore no sensor fusion). But specific models must nevertheless be defined for these cases, because they will occur in the usage of the ROV even if only for short periods. The following subsections describe the use our approach makes of these models.

3.1 Model Selection

The model selected to do the sensor fusion depends on the number and nature of the sensors currently used by the vehicle. For the navigation algorithm, a sensor is not defined in terms of its name, its brand or its part number: it is simply a black box sending measurements of a certain parameter in certain dimensions. In the initialization phase of the ROV, the number and the type of active sensors are given to the navigation system. Knowing what type of sensors it is dealing with, the navigation system can easily choose the appropriate model from the seven available models.

The algorithm also provides an additional degree of flexibility in allowing dynamic addition and removal of sensors at any time during the operation of the vehicle. If, for a certain task, the use of an additional sensor is required, a command is sent to the navigation system and the sensor fusion model is redefined to include this new sensor. The same process is available in case a sensor has to be removed from the fusion process, for example in case of sensor failure.

3.2 Asynchronous Filtering

Sensory data acquisition is done at different rates on the various sensors, and our sensor fusion strategy takes that into account by switching models according to what new data is available at each iteration. This relieves the Kalman filter from the rigid necessity of receiving information from each of the sensors in its model at every iteration. For instance, consider a ROV equipped with a position sensor and an accelerometer. The accelerometer provides data at a rate 10 times higher than that of the position sensor. When both sensors provide new data, the position-acceleration Kalman filter model is used. However, 9 times out of 10 only the accelerometer provides new data, and so the acceleration-only model is used. Continuing to use the old value from the position sensor until a new one is available would introduce a bias error in the estimate. This filter switching process is what we call asynchronous filtering.

3.3 Dealing with Redundant Information

The Kalman filter is designed in its basic form to fuse information from sensors measuring different parameters and output a better state estimate. For instance, it is used to find a better position estimate by fusing data from an accelerometer with data from an acoustic positioning system. But what if there are two redundant measures of absolute position coming from two different positioning systems? Our sensor fusion algorithm handles such redundancy by using two identical Kalman filters to fuse each of the two position measurements with the data from the accelerometer. The outputs of both Kalman filters are then fused together to obtain an optimal global estimate that combines a maximum of information. The method is inspired from the decentralized Kalman filtering process described in [4].

Keeping with the example in the previous paragraph, each one of the two filters has its own error covariance matrix P giving an estimate of the correctness of the filter's estimate. It is these error covariance matrices that are used to fuse the filters together. Each filter contributes to the global estimate in a way inversely proportional to its error covariance matrix. Let X_1 be the estimate from filter 1, X_2 the estimate from filter 2 and P_1 and P_2 their respective error covariance matrices, the global fused estimate X_g is given by Equation 7. The smaller the error covariance of an estimate, the larger its contribution to the global estimate.

$$X_g = \frac{X_1/P_1 + X_2/P_2}{1/P_1 + 1/P_2} \qquad (7)$$

4. Results

To demonstrate the effectiveness of the multiple Kalman filter sensor fusion algorithm, this paper presents two examples, each giving the results of a separate feature of the algorithm, namely the asynchronous filtering and the multi-filter fusion. All the results are obtained using the ROV simulator developed at the IREQ [3]. The vehicle is given a destination position and the closed loop controller guides the vehicle to its objective. Noisy sensor data collection is simulated by adding normally distributed noise to the real values of the parameters being measured. For the sake of simplicity, all examples show the results in only one-dimension, as similar results are obtained in the other dimensions.

4.1 Asynchronous Filtering

To demonstrate asynchronous filtering, the vehicle is equipped with two sensors : an acoustic positioning system, giving a direct measurement of the position of the vehicle at a rate of 5 Hz, and an accelerometer, giving the acceleration of the vehicle at a rate of 50 Hz. The basic filter used in this case is the position-acceleration filter.

The control loop of the vehicle is running at 50 Hz, so running the Kalman filter at 5 Hz and ignoring 9 out of 10 accelerometer measurements is not a solution, even though it would remove the need to switch filters. Since the position measurement remains the same for each cycle of 10 iterations, the error on this measurement progressively grows larger for 9 iterations, and then is reset at the 10^{th}. This results in the saw-tooth error waveform shown in Figure 4.

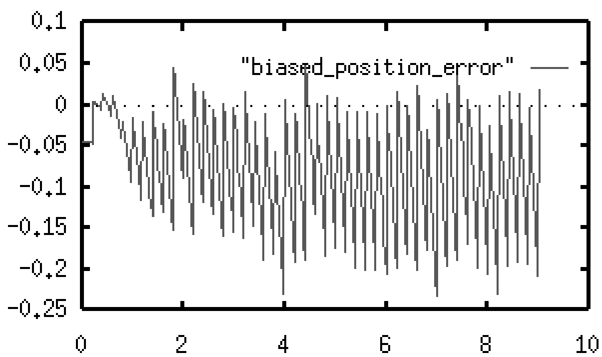

Figure 4 : Position error of the low-frequency positioning system.

Without the filter switching process, the position-acceleration model is used at every iteration with a position measurement that does not reflect the latest movement of the vehicle 9 times out of 10. Remember that the Kalman filter gives a good estimate because the position measurements incorporated in the measurement update process have a zero-mean error distribution. If this is not the case, as in Figure 4, the estimation error of the Kalman filter will be biased, as can be seen on the lower plot of Figure 5. The bias is equal to the mean of the position error of Figure 4.

The plot closest to zero on Figure 5 shows the optimal estimation error resulting from the use of the asynchronous filtering process. This estimation error is unbiased, as it should be when using the Kalman filter. In this case, the position-acceleration model is used 1 time out of 10, and the acceleration-only model the other 9 times, when no new position measurement is available.

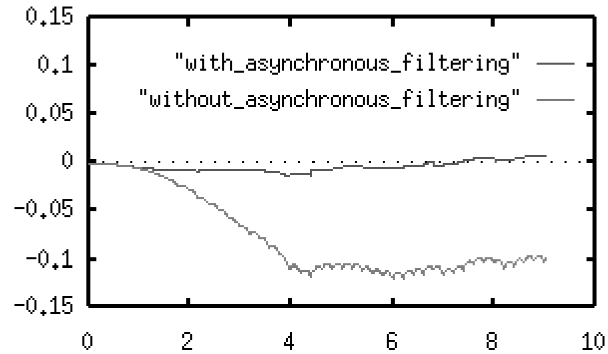

Figure 5 : Kalman filter estimation error with and without asynchronous filtering.

4.2 Multiple Filter Fusion

In this simulation the navigation system is composed of four sensors giving double redundancy : two accelerometers and two positioning systems (giving direct position measurements). Accelerometers number 1 and 2 have respective noises of $0.01 m/s^2$ and $0.05 m/s^2$ while positioning systems 1 and 2 have noises of 0.02m and 0.015m. All sensors send data at the control loop frequency of 50 Hz. This is a case of multiple sensors giving measurements of the same parameters. Two position-acceleration Kalman filters are created by the sensor fusion algorithm, each fusing one of the accelerometers with one of the positioning systems. Figure 6 shows the estimation errors of each of the two filters taken individually. Filter 1 fuses both number one sensors while filter 2 fuses the number two sensors.

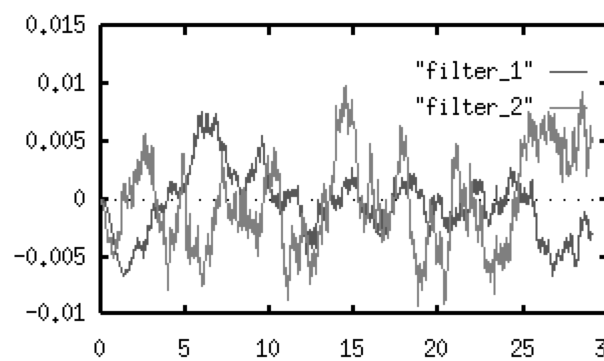

Figure 6 : Estimation error of the two Kalman filters.

The two filters have different error covariance matrices *P*, because they fuse sensors with different noise levels. The error covariance of filter 2 (1.13x10e-5) is larger than that of filter 1 (7.92x10e-6). In accordance with the multi-filter fusion scheme (Equation 7), filter 1, having the smallest error covariance, has a stronger contribution to the global estimate than filter 2. This is confirmed by comparing closely Figure 7 (which shows the global estimate resulting from the fusion of filters 1 and 2) with Figure 6. The global positive effect of this multi-filter fusion process is to give an estimate with a lower average error than each filter taken individually.

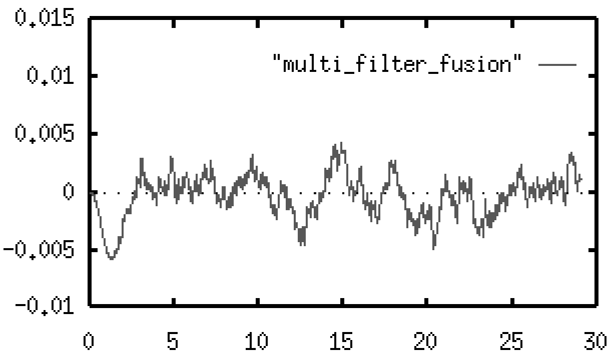

Figure 7 : Multiple filter fusion.

5. Related Works

An important amount of work related to multi-sensor fusion in underwater robotic applications can be found in the literature that cannot be referenced in this paper. Interesting schemes have been devised using the Kalman filter or other innovative fusion methods. The support of different combinations of sensors using different Kalman filters is suggested in [2], but the models (GPS/INS and INS/SONAR) are predefined and specific to the sensor combinations supported by the application. Our algorithm goes further by allowing any combination of sensors without modification of the Kalman filters. A very interesting asynchronous data fusion technique is also proposed in [1] for use in a fusion algorithm called the heuristic position estimator. Our approach differs in that it combines data asynchronously for use in a Kalman filter switching process.

6. Conclusions

This paper describes an adaptable sensor fusion algorithm using multiple Kalman filters for the navigation system of a ROV. Our approach has four important advantages over traditional Kalman filtering. First of all, the algorithm can accommodate various combinations of sensors without the need to modify the models. This is done by selecting the appropriate filters from a bank of Kalman filters according to the sensors that are being used. Only a separate acquisition and data transformation system needs to be altered to support specific sensors. Second, the approach uses a limited number of simple filters by transposing all data into the same reference frame before the fusion takes place. Third, our algorithm can fuse data from sensors operating at different data rates, by switching between different models to use only the newest data. Finally, the strategy supports any number of sensors measuring the same parameter by creating separate filters having the same model and by optimally fusing together these filters according to their error covariance to obtain a better global estimate.

Work is in progress to bring further adaptability to the algorithm by rendering the Kalman filters adaptive.

Acknowledgments

Research for this project has been made possible thanks to funding from the FCAR and Hydro- Québec. F. Michaud is also supported by NSERC, FCAR and CFI.

References

[1] An, P.E., Healey, A.J., Park, J., Smith, S.M. "Asynchronous Data Fusion for AUV Navigation Via Heuristic Fuzzy Filtering Techniques", *MTS/IEEE Oceans '97 Conference*, 1 : 397-402.

[2] Bennamoun, M., Boashash, B., Faruqi, F., Dunbar, M. "The Development of an Integrated GPS/INS/SONAR Navigation System for Autonomous Underwater Vehicle Navigation", *IEEE Symposium on Autonomous Underwater Vehicle Technology 1996*, 256-261.

[3] Bernier, M.-A., Blain, M. "Modélisation et contrôle d'un véhicule sous-marin", *3rd International Conference on Industrial Automation*, Montreal, 1999, 4.9-4.12

[4] Brown, R.G., Hwang, P.Y.C. *Introduction to Random Signals and Applied Kalman Filtering*, 3rd edition, John Wiley & Sons, 1997.

[5] Côté, J., Lavallée, J. "Augmented Reality Graphic Interface for Upstream Dam Inspection", *SPIE - Telemanipulator and Telepresence Technologies II*, Philadelphia, October 1995.

[6] Magill, D.T. "Optimal Adaptive Estimation of Sampled Stochastic Processes", *IEEE Transactions on Automatic Control*, AC-10(4) : 434-439, October 1965.

[7] Mehra, R., Seereeram, S., Bayard, D., Hadaegh, F. "Adaptive Kalman Filtering, Failure Detection and Identification for Spacecraft Attitude Estimation", *4th IEEE Conference on Control Applications*, 1995, 176-181.