

# HIGHWAY EVALUATION OF TERRAIN-AIDED LOCALIZATION USING PARTICLE FILTERS

Adam J. Dean, Pramod K. Vemulapalli, Sean N. Brennan\*

Department of Mechanical Engineering  
Pennsylvania State University  
318 Leonhard Building  
University Park, Pennsylvania 16802  
Email: ajdean, pkv106, sbrennan@psu.edu

## ABSTRACT

*This work evaluates a real-time algorithm to localize a vehicle on a highway in the direction of travel without the use of GPS. The algorithm uses a particle filter to estimate vehicle position along a map of road grade using real-time pitch measurements from an in-vehicle pitch sensor as the input. Experiments over 60 kilometers along Interstate I-80 and US Route 220 in Pennsylvania are used to demonstrate the algorithm, observe the speed of convergence, and evaluate several methods of implementation. The results indicate that the method can localize a vehicle with a position accuracy of 5 meters after traveling about 1 kilometer within the 60 kilometer map.*

## INTRODUCTION

For reasons of safety and efficiency, there is a great deal of interest in localizing road vehicles. Today, the Global Positioning System (GPS) serves as the primary means to determine vehicle position. However, due to poor GPS signal reception in some locations, the ease of jamming a GPS signal in battlefield operation, and the need for sensor redundancy in vehicle automation and driver assist applications, there has been great interest in localization technologies independent of GPS.

This study directly follows the work done in [1] and the preliminary discussion, methodology, and experimental validation can be found therein. This study extends the previous work of vehicle localization along a one-mile test track to implementation along actual highways with an on-vehicle map of over 60 kilometers in length. The purpose of this paper is to demonstrate the feasibility of the algorithm to localize a vehicle in an un-

controlled environment within which the algorithm is meant to be implemented, like an interstate highway, versus a closed test track as shown in [1].

Highway data is perhaps the most challenging demonstration of the algorithm's potential to localize a vehicle across large terrain maps. Not only is the vehicle traveling very quickly, and thus the wheelbase filtering effect is quite prominent, but the highway roadway surface itself represents the smoothest roadways available and hence the least likely to excite the pitch variations that are used for correlation.

Several methods have been used to localize a vehicle without GPS or during short GPS outages including fusion of GPS with odometry [2], inertial measurements, vision [3], laser scans [4], or using a network of beacons [5]. Of particular relevance to this work is the fusion of GPS with map data. A demonstration of this capability was accomplished in real-time [2] where a Kalman filter was used to combine GPS data with odometry measurements.

Similar to the work done in [6] where an aircraft's elevation profile is matched to a digital elevation map, this work demonstrates the use of a terrain map for real-time vehicle localization with the goal to obtain GPS-accurate position resolution of a vehicle's longitudinal position. Similar terrain-aided applications include missile guidance systems [7] and underwater robotics [8]. This work uses a particle filter, which is gaining wide use for localization [6, 9, 10], tracking [11], and even vehicle localization during GPS outages [12].

In this study it is assumed that the lane of travel has been previously mapped and that on-vehicle storage of the resulting terrain information is available. The first assumption is quite realistic given the large number of ongoing research projects focused on mapping terrain, whereas the second assumption is increas-

\*Address all correspondence to this author.

ingly valid given the exponentially decreasing costs of data storage and recent integration of similar on-vehicle map databases into commercial products, for example the “TomTom” navigation devices.

## PARTICLE FILTER

Details presented in this section are a summary of the algorithm used in [1] with significant portions repeated here for completeness; however, the parameter values within the algorithm are adjusted from the previous work in order to extend the application to long-range highway data.

Particle filters are Monte-Carlo estimators that are known to be quite robust to non-Gaussian variance distributions similar to what would occur in this work due to similarities in the road profile along different segments. A Kalman filter could not be used to estimate the vehicle position because the vehicle can start anywhere along the map, hence the initial probability distribution is uniform whereas a Kalman filter requires a gaussian probability distribution.

The algorithm begins by converting the time-dependant data to the spatial domain, or more plainly as a function of distance from the starting point. Other than wheelbase filtering which is dependant on velocity, this removes velocity dependence on the pitch data. A set of  $N$  equally weighted and randomly distributed particles are located along the terrain map. The pitch estimate of each particle location is determined from the pitch map; particles that lie between the discretely mapped locations are determined via linear interpolation.

The particle filter algorithm is based off of Algorithm 3 in [11] and iterates every  $dX$  distance of vehicle travel by repeating the following: First, the position estimates, denoted by  $X$ , at interval  $k$  are updated from the previous estimate by

$$X^k = X^{k-1} + dX + dO \quad (1)$$

where  $dO$  is gaussian noise of variance  $R_O$ , equal to the variance of the odometry measurement.

Second, the weights of the position particles are updated by measuring the actual vehicle pitch and comparing it to the particle’s pitch estimate using a standard particle weighting function. The importance density is assumed to be the prior density and the pdf is assumed to be gaussian:

$$q_i^k = \frac{\exp\left(-\frac{1}{2R} \cdot (\theta_a - \theta_{p,i})^2\right)}{\sum_{i=1}^N \left(\exp\left(-\frac{1}{2R} \cdot (\theta_a - \theta_{p,i})^2\right)\right)} \quad (2)$$

Here  $R$  is the measurement noise variance on pitch,  $\theta_a$  is the measured pitch, and  $\theta_{p,i}$  is the  $i^{th}$  particle’s pitch corresponding to its position along the terrain map.

Third, the particles are resampled following Algorithm 2 described in [11] where the number of effective position particles

$N_{eff}$  is calculated as

$$N_{eff} = \frac{1}{\sum_{i=1}^N (q_i^k)^2} \quad (3)$$

and when  $N_{eff}$  is below a threshold of  $N_T$ , the position particles are re-sampled by

$$\begin{aligned} c &= \text{cumsum}(q^k) \\ u_1 &= \text{rand}(1) \cdot N^{-1} \\ i &= 1 \\ \text{for } j &= 1 \dots N \\ u_j &= u_1 + (j-1) \cdot N^{-1} \\ \text{while } u_j &> c_i \\ i &= i + 1 \\ \text{end} \\ X_j^k &= X_i^k \\ q_j^k &= N^{-1} \\ \text{end} \end{aligned} \quad (4)$$

where  $\text{rand}(1)$  is an evenly distributed random number in  $[0, 1]$  and  $\text{cumsum}$  is the cumulative sum.

Fourth, the vehicle’s position is estimated as the mean of the position particles. This use of the entire population to characterize the estimate is fairly conservative since the position estimate of the “best” particle is in general far better than that of the population mean. However, for this study on the feasibility of the algorithm itself, convergence of the population to the correct solution is a far better indicator of algorithm performance than is analysis of the best particle estimate.

Fifth, as a means of measuring the accuracy of the algorithm the error in the prediction is calculated from the true vehicle position as measured by GPS. Because a driver is not capable of driving over the exact center of the lane, a path error is introduced called the lane-keeping error. An estimate of error between the predicted position to the actual vehicle position would include the lane-keeping error. In order to remove this error the measured vehicle position is projected to the nearest position on the map. The corrected error is calculated as the distance from the predicted position to the corrected measured position as

$$E^k = |\bar{X} - x_c^k| \quad (5)$$

where  $\bar{X}$  is the mean location of the position estimates and  $x_c$  is the vehicle’s GPS measured location projected to the map. The lane-keeping error is calculated as

$$E_l^k = |x_c^k - x_m^k| \quad (6)$$

where  $x_m$  is the vehicle’s measured GPS location. An example of the projection and error measurement is shown in Fig. 1 [1].

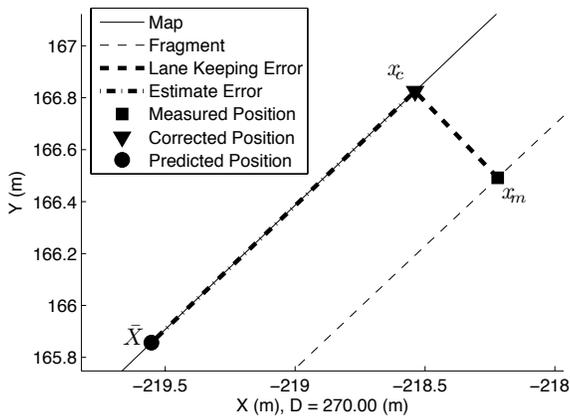


Figure 1. LANE KEEPING ERROR AND ITS CORRECTION.

## EXPERIMENTAL RESULTS

The particle filter algorithm used in this work was implemented off-line using data previously recorded. The data was collected using an instrumented vehicle equipped with a portable sensor integration platform designed for research applications in robotics, surveying and vehicle dynamics. The onboard NovAtel Synchronized Position Attitude Navigation (SPAN) system, based on an OEM4 DL4-PLUS dual frequency receiver and the Honeywell HG1700 military grade IMU, can acquire data up to 100Hz. The position errors in the latitude and longitude data, with full satellite visibility, are about 2 meters (one sigma) and the errors in the orientation angles are 0.017 and 0.02 degrees (one sigma) for the roll and pitch angles respectively. The Honeywell HG1700 military grade IMU is based on the Honeywell GG1308 ring laser gyro and Honeywell RBA500 accelerometers and has a performance range consistent with requirements of tactical missiles and smart munitions systems. The IMU has only a 10 deg/hr gyro bias and 3 milli-g acceleration bias.

To test the algorithm, the pitch response of a Ford Explorer was measured as the vehicle traveled for over 60 km along Interstate I-80 and US Route 220 in Pennsylvania and stored for a terrain map. The vehicle was then used a second time to travel along a small portion of the previous route, without regard to following the exact path, in order to collect the data to be used for localization, hereafter called the fragment data. Figure 2 shows a few kilometers of the map and fragment pitch data, demonstrating visible variations in pitch between the data sets due to the differences in speed, inexact path tracking, and possibly vehicle loading differences between measurements.

The particle filter was implemented with  $N$  equal to 39,842 particles (1,000 particles per mile on the map [1]), iterated every  $dX = 100$  meters of travel, the map was decimated to 5 meters in order to reduce the computational load, and  $N_T = 0.95 \cdot N$ . The value of the pitch noise variance,  $R$ , was relaxed to a value of 0.1 degrees<sup>2</sup>, much greater than the variance in the IMU pitch mea-

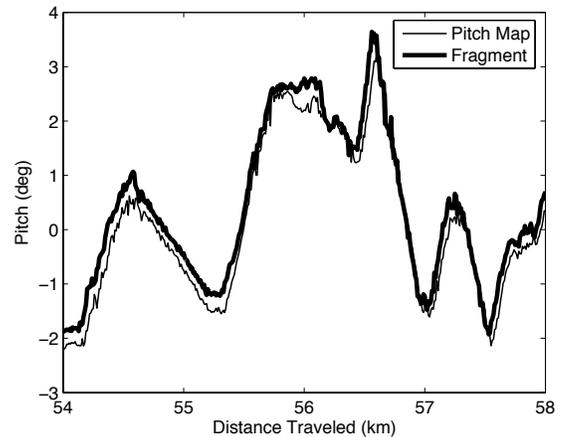


Figure 2. A FEW KILOMETERS OF THE TERRAIN MAP AND TEST FRAGMENT PITCH DATA ALONG I-80 AND ROUTE 220 IN PA.

surement of 0.000169 degrees<sup>2</sup> [13].  $R$  is called a variance, but it is actually an indicator of the amount of trust is placed in the accuracy of the measurement; a small  $R$  means the measurement is trusted to accurately indicate the position of the vehicle along the map. In this case,  $R$  was relaxed to account for the mean offset between the pitch data, as shown in Fig. 2, likely due to different loading conditions in the vehicle between the data sets; otherwise the algorithm would converge too quickly to an erroneous solution at another portion of the map where the pitch value would be equivalent to the biased pitch measurement.

The value of  $R_O$  was calculated using the results of an effective tire radius study [14], where a tire with a specified radius of 321.65 mm was measured to have a nominal effective radius of 310.4 mm. Under different loading and tread conditions the effective radius was shown to vary by as much as 0.8%, which causes an equivalent error in a dead-reckoning odometry measurement. Thus, to be conservative, the variance in the odometry measurement was chosen to be  $R_O = (0.01 \cdot dX)^2$  m<sup>2</sup>, or 1% of the distance traveled between iterations. The vehicle used in this study could not be equipped to measure odometry, so the odometry measurement was instead extracted from GPS by calculating the shortest distance between data points; however, there are many alternatives to measuring odometry without using GPS.

The algorithm's convergence results are shown in Fig. 3, an overhead view of the highway with the position estimates as dots, the mean estimate as a circle, and the actual "true" position measured from GPS shown as a box. It can be seen that as the vehicle travels, the position estimates converge to the measured vehicle location to within the accuracy of the GPS/INS system.

The error in the vehicle's position estimate (Eq. 5) as a function of the distance traveled is shown in Fig. 4. It can be seen that the algorithm localized the vehicle to the accuracy of the map after traveling about two kilometers. Also shown in this figure and following figures is a line representing the mapping interval of

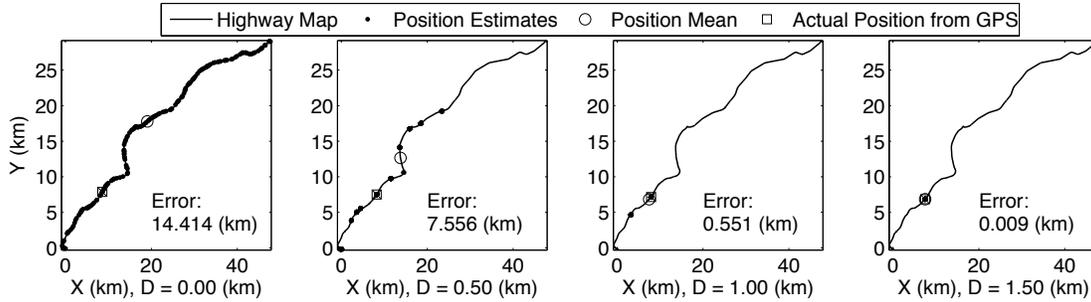


Figure 3. POSITION ESTIMATES AFTER THE VEHICLE TRAVELED A DISTANCE D.

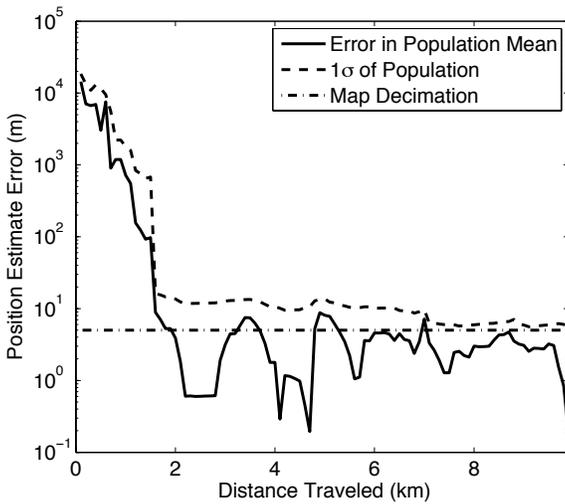


Figure 4. POSITION ESTIMATE ERROR AS A FUNCTION OF THE DISTANCE TRAVELED.

the terrain map, 5 meters, a value which places a lower-limit on the achievable estimator accuracy.

The lane-keeping error and position estimate error are calculated at every iteration following Eqs. 5 and 6. Figure 5 demonstrates the position estimate error as a function of the lane-keeping error from once the localization error is reduced to the map decimation until the end of the algorithm.

It can be seen that there is no direct correlation between the lane-keeping error and the algorithm accuracy, contrary to the off-line correlation study in [13]. Also, it is evident that the algorithm was able to localize a vehicle within the accuracy of the map even though the vehicle had as much as 5 meters of lane-keeping error. This is probably due to fact that the lane-keeping error in this work is mostly due to GPS system innaccuracy rather than true error. In the off-line correlation study differential GPS was used that could resolve lane-keeping error to centimeter level accuracy.

This also demonstrates that, while the high frequency pitch

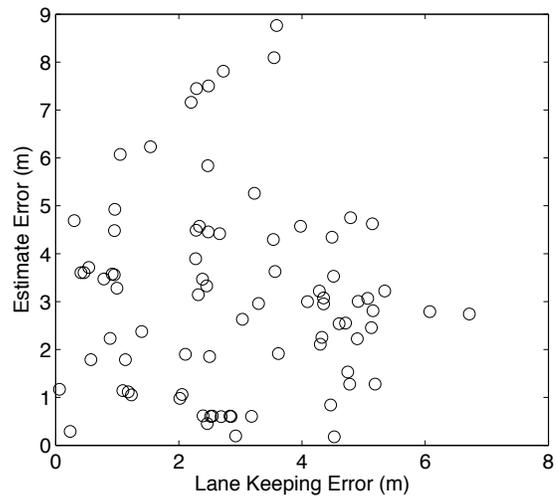


Figure 5. POSITION ESTIMATE ERROR AS A FUNCTION OF THE LANE KEEPING ERROR.

data may vary significantly between travel lanes, the low frequency data can still be used to sufficiently localize a vehicle after the solution has converged. Thus a driver using this localization method would not need to be concerned with following the exact same path as the map and could even move within a few meters of the mapped lane. This gives the possibility of only having to map the center lane of the highway in order to localize a vehicle across a three lane highway.

## ROLL DATA

An alternate approach to localizing the vehicle is to use the roll measurement instead of a pitch measurement. The roll data was collected in the same manner and in the same location as the pitch data. The roll data, as shown in Fig. 6, varies as much as the pitch data.

The algorithm is implemented again using roll data while

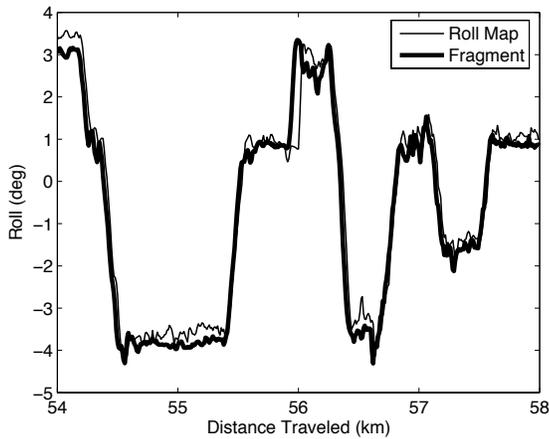


Figure 6. A FEW KILOMETERS OF THE TERRAIN MAP AND TEST FRAGMENT ROLL DATA ALONG I-80 AND ROUTE 220 IN PA.

excluding the pitch data, thus Eq. 2 is modified to

$$q_i^k = \frac{\exp\left(-\frac{1}{2 \cdot R_r} \cdot (\phi_a - \phi_{p,i})^2\right)}{\sum_{i=1}^N \left(\exp\left(-\frac{1}{2 \cdot R_r} \cdot (\phi_a - \phi_{p,i})^2\right)\right)} \quad (7)$$

where  $R_r$  is the measurement noise variance on roll, which was set to equal  $R$ ,  $\phi_a$  is the measured roll angle, and  $\phi_{p,i}$  is the  $i^{\text{th}}$  particle's roll corresponding to its position along the roll map.

The resulting error convergence is shown in Fig. 7. It can be seen that localization using the roll data is about as accurate as using the pitch measurement; however, in order to converge to the accuracy of the map decimation the vehicle needed to travel twice as far as when the pitch data was used.

It can also be seen that the first four kilometers of travel resulted in poor localization, which is due to the little variation of the roll data from the mean roll measurement within that distance, as shown in Fig. 6. This demonstrates the tradeoff of relaxing  $R_r$ ; if  $R_r$  is reduced to the variance of the roll measurement sensor, then the algorithm would be more sensitive to the slight changes in roll and possibly decrease the time of convergence; however, the algorithm would then become very sensitive to bias errors between the roll measurement and the roll map and in the presence thereof could result in an erroneous position estimate. This encourages a further investigation, which is beyond the scope of this study, into an adaptive algorithm that could vary the value of  $R_r$  as a function of the error in the measured roll and the corresponding roll of the estimated position along the map.

## PITCH AND ROLL DATA

In an attempt to improve the performance of the algorithm, a redundancy can be used in the position estimation by including

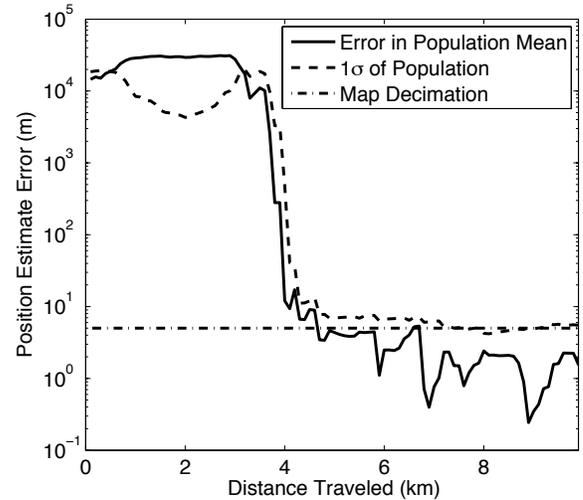


Figure 7. POSITION ESTIMATE ERROR AS A FUNCTION OF THE DISTANCE TRAVELED USING ONLY A ROLL MAP.

the roll map with the pitch map. Thus Eq. 2 is modified to

$$q_i^k = \frac{Q_i^k}{\sum_{i=1}^N (Q_i^k)} \quad (8)$$

where

$$Q_i^k = \exp\left(-\frac{1}{2 \cdot R} \cdot (\phi_a - \phi_{p,i})^2 - \frac{1}{2 \cdot R_r} \cdot (\theta_a - \theta_{p,i})^2\right) \quad (9)$$

and  $R_r$  is again set to equal  $R$ .

The results of using this modified algorithm are shown in Fig. 8. Because the accuracy of the map is limited to the decimation distance of 5 meters, which in turn is limited by the accuracy of uncorrected GPS, we cannot determine if the redundancy improved the overall accuracy. However, the added measurement did increase the rate of convergence and thus decreased the distance required to converge from 2 kilometers to nearly 1 kilometer. This suggests that both pitch and roll measurements could be used initially in the algorithm in order for the estimate to converge quickly, and once the position estimate has converged to a desired accuracy, the algorithm could switch to using only the pitch or roll measurement.

## FAULTY SENSOR DETECTION

The algorithm demonstrated in this study can also be used to detect when a sensor is faulty. For example, after the algorithm has converged to a solution and is tracking the vehicle accurately along the highway, the pitch and roll measurements can be predicted using the map data. If a sensor has diverged abnormally

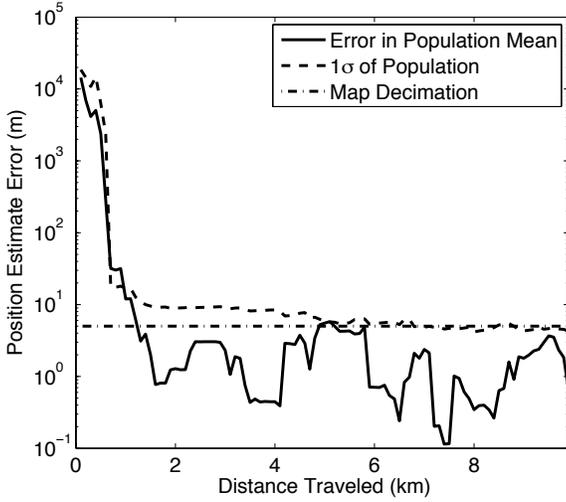


Figure 8. POSITION ESTIMATE ERROR AS A FUNCTION OF THE DISTANCE TRAVELED USING PITCH AND ROLL MAPS.

beyond the expected value, while the other sensor’s measured values continue to be in accord with the expected value, then the diverging sensor can be assumed to be faulty or accruing error abnormally. The sensor can be then switched off or noted for replacement or maintenance.

To demonstrate, the algorithm was used again using only the pitch measurement for estimation and an abnormal amount of bias error, 5 degrees, was added to only a portion of the roll measurements. Throughout the algorithm the error in the pitch and and roll sensor data are calculated as

$$E_{\theta} = |\theta_a - \theta_{p,e}| \quad (10)$$

$$E_{\phi} = |\phi_a - \phi_{p,e}| \quad (11)$$

where  $\theta_{p,e}$  and  $\phi_{p,e}$  are the pitch and roll values at the estimated vehicle location along their respective maps. The errors are calculated at each iteration, and shown in Fig. 9 where it can be seen that the error in the roll measurement is clearly predicted.

## CONCLUSIONS AND FUTURE WORK

This work shows that a vehicle’s longitudinal position can be estimated along a long stretch of highway given a terrain map, pitch measurements, and odometry. The convergence of the estimate is seen to occur within approximately 1 to 2 kilometers of highway driving, with converged longitudinal positioning error of 5 meters in accuracy, or the decimation of the terrain map, as compared to a GPS system. It has also been shown that localizing a vehicle using its pitch measurements is as accurate as using the roll measurements or both of them combined. A method of faulty sensor detection has also been demonstrated. Further

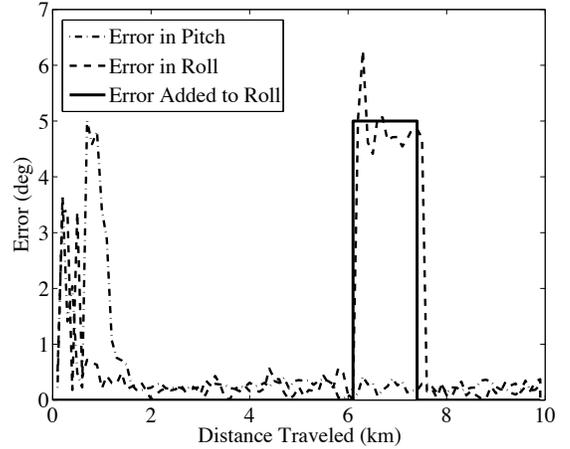


Figure 9. ERROR IN THE MEASURED AND PREDICTED PITCH AND ROLL MEASUREMENTS AND THE REGION OF THE INTRODUCED BIAS ERROR IN THE ROLL MEASUREMENTS.

study is under way to increase the localization accuracy and decrease the computational load of the particle filter algorithm.

## REFERENCES

- [1] Dean, A., Martini, R., and Brennan, S., 2008. “Terrain-based road vehicle localization using particle filters”. In Proceedings. 2008 American Control Conference.
- [2] Najjar, M. E. E., and Bonnifait, P., 2005. “A road-matching method for precise vehicle localization using belief theory and kalman filtering”. *Auton. Robots*, **19**(2), pp. 173–191.
- [3] Oskiper, T., Zhu, Z., Samarasekera, S., and Kumar, R., 2007. “Visual odometry system using multiple stereo cameras and inertial measurement unit”. In Computer Vision and Pattern Recognition, 2007. IEEE Conf. on, pp. 1–8.
- [4] Jabbour, M., Bonnifait, P., and Cherfaoui, V., 2006. “Enhanced local maps in a GIS for a precise localisation in urban areas”. In Intelligent Transportation Systems, 2006. Proceedings. 2006 IEEE, pp. 468–473.
- [5] Chu, H.-C., and Jan, R.-H., 2005. “A GPS-less self-positioning method for sensor networks”. In 2005. Proceedings. 11th International Conference on Parallel and Distributed Systems, Vol. 2, pp. 629–633.
- [6] Gustafsson, F., Gunnarsson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., and Nordlund, P. J., 2002. “Particle filters for positioning, navigation, and tracking”. *Signal Processing, IEEE Trans. on*, **50**(2), Feb., pp. 425–437.
- [7] Golden, J. P., 1980. “Terrain contour matching/TERCOM/-A cruise missile guidance aid”. *Image processing for missile guidance*, pp. 10–18.
- [8] Williams, S., Dissanayake, G., and Durrant-Whyte, H., 2001. “Towards terrain-aided navigation for underwater robotics”. *Advanced Robotics*, **15**(5), pp. 533–549.

- [9] Caron, F., Davy, M., Duflos, E., and Vanheeghe, P., 2007. "Particle filtering for multisensor data fusion with switching observation models: Application to land vehicle positioning". *Signal Processing, IEEE Transactions on*, **55**, June, pp. 2703–2719.
- [10] Chausse, F., Laneurit, J., and Chapuis, R., 2005. "Vehicle localization on a digital map using particles filtering". In *Intelligent Vehicles Sym.*, 2005. Proc. IEEE, pp. 243–248.
- [11] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., 2002. "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking". *Signal Processing, IEEE Transactions on*, **50**(2), Feb., pp. 174–188.
- [12] Boucher, C., and Noyer, J. C., 2006. "Hybrid filtering for map-aided vehicle navigation". In *Computational Intelligence for Measurement Systems and Applications, Proceedings of 2006 IEEE Int'l. Conf. on*, pp. 46–50.
- [13] Martini, R. D., 2006. "GPS/INS sensing coordination for vehicle state identification and road grade positioning". MS, Pennsylvania State University, August.
- [14] Carlson, C. R., and Gerdes, J. C., 2005. "Consistent nonlinear estimation of longitudinal tire stiffness and effective radius". *IEEE Transactions on Control Systems Technology*, **13**(6), Nov., pp. 1010–1020.