Hindawi Publishing Corporation
International Journal of Navigation and Observation
Volume 2012, Article ID 576807, 16 pages
doi:10.1155/2012/576807

Research Article

Augmented Kalman Filter and Map Matching for 3D RISS/GPS Integration for Land Vehicles

Matthew Cossaboom,[1] Jacques Georgy,[2] Tashfeen Karamat,[3] and Aboelmagd Noureldin[1,3]

[1] *Navigation and Instrumentation Research Group (NavINST), Electrical and Computer Engineering Department, Royal Military College of Canada, Kingston, ON, Canada K7K 7B4*
[2] *Trusted Positioning Inc., Calgary, AB, Canada T2L 2K7*
[3] *Navigation and Instrumentation Research Group (NavINST), Electrical and Computer Engineering Department, Queen's University, Kingston, ON, Canada K7L 3N6*

Correspondence should be addressed to Jacques Georgy, jgeorgy@trustedpositioning.com

Received 1 July 2011; Revised 1 October 2011; Accepted 28 October 2011

Academic Editor: Olivier Julien

Copyright © 2012 Matthew Cossaboom et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Owing to their complimentary characteristics, global positioning system (GPS) and inertial navigation system (INS) are integrated, traditionally through Kalman filter (KF), to obtain improved navigational solution. To reduce the overall cost of the system, microelectromechanical system- (MEMS-) based INS is utilized. One of the approaches is to reduce the number of low-cost inertial sensors, decreasing their error contribution which leads to a reduced inertial sensor system (RISS). This paper uses KF to integrate GPS and 3D RISS in a loosely coupled fashion to enhance navigational solution while further improvement is achieved by augmenting it with map matching (MM). The 3D RISS consists of only one gyroscope and two accelerometers along with the vehicle's built-in odometer. MM limits the error growth during GPS outages by restricting the predicted positions to the road networks. The performance of proposed method is compared with KF-only 3D RISS/GPS integration to demonstrate the efficacy of the proposed technique.

1. Introduction

Low-cost navigation applications are highly dependent on satellite navigation systems, primarily global positioning system (GPS). It is composed of a constellation of 24 (with room to spare for some additional) satellites covering the globe in a manner that ensures continuous worldwide coverage. To obtain accurate positioning data, one must be in direct line of sight with at least four satellites. The main advantage of the GPS is that it can determine one's location, accurate to within a range of 30 m when using a single point positioning technique, and to a few centimeters when using a differential GPS technique [1–4]. However, the satellite signal can be blocked in GPS-denied environments such as urban canyons and tunnels. This is a major problem because there will be an interruption in the real-time positioning information. To overcome this navigational data gap, GPS is usually integrated with an inertial navigation system (INS) because it does not rely on any external sources [1–3]. The INS is a self-contained system consisting of three accelerometers and three gyroscopes which is mounted on the moving platform to monitor linear accelerations and angular velocities. Given the initial values of navigation parameters, the measurements from INS can be processed to determine current position, velocity, and attitude of the moving platform with respect to a certain frame of reference [4, 5]. Since higher-end INS are very expensive therefore not suitable for low-cost applications, contemporary research is focused on microelectromechanical system- (MEMS-) based INS [6–8]. They are the key to the navigation applications where size, weight, and cost are the main concern, such as land vehicle and pedestrian navigation. However, the MEMS-based INS sensors suffer from noise, bias, and drift errors which are much more serious than the higher-grade sensors [9, 10]. Therefore, when MEMS-based INS works unaided, the performance will degrade very quickly compared to

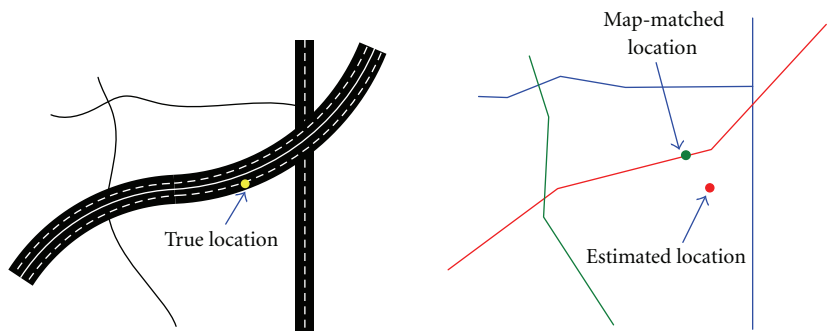


Figure 1: The map matching problem (adapted from [31]).

higher-grade INS [11]. Since a bias in accelerometer contributes to an error in position which is proportional to t^2 and a bias in gyroscope causes an error in position which is proportional to t^3, this research utilizes one gyroscope and two accelerometers along with vehicle's built-in odometer to get a full 3D navigational solution [12].

Integrating INS with GPS has several advantages because they possess complementary error characteristics. GPS bounds the INS drift in the long run whereas INS fills the GPS data gaps during GPS signal interruption. The traditional method of INS/GPS integration is Kalman filtering (KF) which can be implemented in a loosely coupled or a tightly coupled manner. The loosely coupled scheme of integration requires at least four satellites for GPS measurement update whereas tightly coupled integration can benefit from GPS even when only one satellite is available. However, tightly coupled approach is much more complex to implement and hardly any superior when more than three satellites are visible. Both the aforementioned approaches can be implemented in open- and closed-loop fashion. Open-loop filters do not use feedback; the input data does not use corrections whereas closed-loop filters use the previous corrections to minimize the approximation errors [13, 14].

KF uses a linearized system and measurement models. KF techniques suffer from divergence during outages due to approximations during the linearization process, especially when utilizing MEMS-based inertial measurement units (IMUs). This problem can be avoided by using particle filter (PF) which enhances the performance of the MEMS-based INS by including the nonlinearities in the system and measurement models [15, 16]. Particle filtering is a nonlinear filtering technique that does not require the system model to be linearized. It can accommodate for arbitrary sensor characteristics, motion dynamics, and noise distribution because of its ability to deal with nonlinear non-Gaussian models [17, 18]. Other methods of integration have been investigated based on artificial intelligence (AI), also known as neural network (NN) [6, 19–24]. The major challenge of PF or the AI method is the fact that they are computationally expensive and may not be useful in some applications.

Map matching (MM) is the process of utilizing a digital road network map database to improve the predicted position errors during integration [25–29]. Motivated by the simplicity and drawbacks of KF, this research will focus on reducing the KF integration errors by utilizing MM. The goal of MM is to match the estimated location with the road network map [30]. Figure 1 gives a good representation of the MM approach [31]. The left diagram displays the person's actual location on the actual streets whereas the right diagram displays the set of estimated arcs (digital road networks) with the estimated location and the MM location. This example uses a piecewise linear solution to estimate the arcs in the roads.

2. RISS/GPS Integration Using KF

KF uses a linearized system model and has several limitations. It requires a stochastic model of the inertial sensor errors and a priori information about the data covariance provided by both inertial system and GPS [6, 32]. KF techniques suffer from divergence during outages due to approximations during the linearization process, especially when utilizing MEMS-based IMUs. As a result, the inertial-based position and velocity errors could grow quite significantly. The details of traditional KF and derivation of its equations can be found in some excellent texts such as [33–36]. However, a brief overview of KF equations is presented.

KF operates in two distinctive stages: (1) prediction stage and (2) update stage. In the prediction stage, a new prediction of the error states (1) and error covariance (2) are determined for the next time step. The equations for the prediction stage are as follows [35, 36].

Prediction of error states:

$$\hat{x}_k^- = \Phi_{k,k-1}\hat{x}_{k-1}^+, \tag{1}$$

where $\hat{x}_k$ is the error state vector, $(-)$ indicates the *a priori* estimate while *posteriori* estimate is indicated with $(+)$. The Φ is the state transition matrix. The predicted error covariance is expressed as follows.

Prediction of the error covariance:

$$P_k^- = \Phi_{k,k-1}P_{k-1}^+\Phi_{k,k-1}^T + G_{k-1}Q_{k-1}G_{k-1}^T, \tag{2}$$

where G is the noise distribution matrix and Q is the covariance of the system noise.

In the update stage, the KF makes corrections to the predicted state estimate based on new information from the GPS measurements. These corrections are appropriately weighed

though Kalman gain (3) which determines if the prediction or the measurement should be trusted more. Then the Kalman gain is used to update the state estimate (4) and error covariance matrix (5) as the posteriori estimate for the next prediction stage.

Kalman gain:

$$K_k = \left(P_k^-\right) H_k^T \left(H_k P_k^- H_k^T + R_k\right)^{-1}, \tag{3}$$

where R is the covariance of the measurement noise and H is the measurement design matrix.

Updating of error states:

$$\hat{x}_k^+ = \hat{x}_k^- + K_k\left(Z_k - H_k \hat{x}_k^-\right), \tag{4}$$

where Z is the difference between the INS and GPS position and velocity components.

Updating of error covariance matrix:

$$P_k^+ = (I - K_k H_k) P_k^-, \tag{5}$$

where I is the identity matrix.

This research used a loosely coupled 3D RISS/GPS integration approach. Loosely coupled integration helps assessing the map matching better because if we use the tightly coupled integration then during the partial GPS outage there will be two or three satellites which will help the solution as well as the map matching. Therefore to be able to better assess the map matching, we focus on loosely coupled integration because in loosely coupled integration, there is no satellites at all during the outage and enhancement contribution will come from map matching algorithm. As mentioned earlier, due to the complex error characteristics of MEMS-based sensors, this paper uses a different configuration of inertial sensors where only one gyroscope and two accelerometers along with the vehicle's built-in odometer are used to obtain a three-dimensional navigation solution [12, 37]. This is termed as *reduced inertial sensor system* (RISS) as opposed to *full IMU* which uses three gyroscopes and three accelerometers. The 2D RISS was first introduced in [38] where a KF was used for 2D RISS/GPS integration, a PF for 2D RISS/GPS integration was proposed in [39]. The 3D RISS was first introduced in [12], together with its full derivation, and its detailed advantages over a full-IMU-based solution and over 2D solutions. A tightly coupled KF 3D RISS/GPS integration solution was proposed in [37]. As explained in the aforementioned literature, there are only three sensors contributing to the errors versus six. A gyroscope is mounted so that its axis is aligned with the vertical axis of the vehicle to obtain the azimuth, and the vehicle odometer provides the forward speed [37]. Two accelerometers, instead of gyroscopes, are used to compute the pitch and roll angles. They are aligned with the forward and transversal axes of the vehicle body frame. The pitch and azimuth angles are used to calculate the velocities and then the position components can be calculated.

The azimuth angle is calculated by integrating the gyroscope measurement ω_z, as shown in (6). This measurement includes the component of the Earth rotation and rotation of the local level frame on the Earth's curvature, these quantities are removed from the measurement before integration [40],

$$\dot{A} = -\left[\omega_z - \omega^e \sin\varphi - \frac{v_e \tan\varphi}{R_N + h}\right], \tag{6}$$

where ω^e is the Earth's rotation rate, φ is latitude, v_e is the east velocity, R_N is the normal radius of the earth ellipsoid, and h is altitude.

When the vehicle is moving, the forward accelerometer measures the forward vehicle acceleration as well as the component due to gravity. Therefore, the following relationship is used to calculate the pitch angle:

$$p = \sin^{-1}\left(\frac{f_y - a_{\text{od}}}{g}\right), \tag{7}$$

where f_y is the forward accelerometer measurement, a_{od} is the odometer-derived acceleration, and g is the Earth's gravity.

The transversal accelerometer measures the normal component of the vehicle acceleration and the component due to gravity. Therefore roll angle is computed as follows:

$$r = -\sin^{-1}\left(\frac{f_x + v_{\text{od}}\omega_z}{g\cos p}\right), \tag{8}$$

where f_x is the transversal accelerometer measurement and v_{od} is the odometer measurements.

The three velocities (east v_e, north v_n, and up v_u) are calculated using A, p, and v_{od} through the following relationship:

$$\mathbf{v} = \begin{bmatrix} v_e \\ v_n \\ v_u \end{bmatrix} = \begin{bmatrix} v_{\text{od}} \sin A \cos p \\ v_{\text{od}} \cos A \cos p \\ v_{\text{od}} \sin p \end{bmatrix}. \tag{9}$$

Then the time rate of change of the position components can be obtained as follows:

$$\dot{\mathbf{r}}^l = \begin{pmatrix} \dot{\varphi} \\ \dot{\lambda} \\ \dot{h} \end{pmatrix} = \begin{pmatrix} 0 & \dfrac{1}{R_M + h} & 0 \\ \dfrac{1}{(R_N + h)\cos\varphi} & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v^e \\ v^n \\ v^u \end{pmatrix} = D^{-1}\mathbf{v}^l, \tag{10}$$

where λ is the longitude and R_M is the meridian radius of the earth ellipsoid.

When RISS is integrated with GPS using a KF to create a 3D position solution, the error state vector has nine error states. They are latitude, longitude, and altitude errors $(\delta\varphi, \delta\lambda, \delta h)$, the east, north and up velocity errors $(\delta v_e, \delta v_n, \delta v_u)$, the azimuth error δA, the gyroscope error $\delta\omega_z$, and the error from the odometer acceleration δa_{od}. The stochastic errors associated with the gyroscope and the odometer-derived acceleration are modeled by Gauss-Markov model where γ_{od} is the inverse of the autocorrelation time for the odometer-derived acceleration noise, σ_{od}^2 is the variance of odometer-derived acceleration noise, β_z is the inverse of the autocorrelation time for the gyroscope noise, and σ_z^2 is the variance of the gyroscope noise. The complete error state

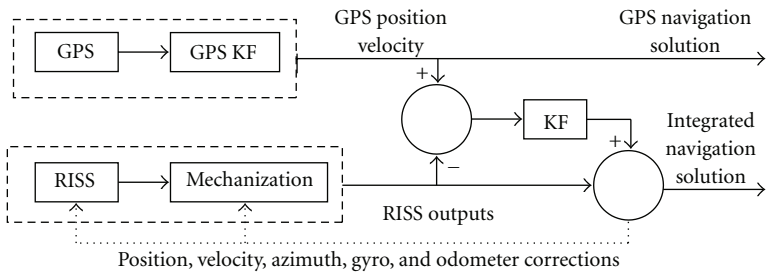


Figure 2: Loosely coupled RISS/GPS KF integration diagram.

system model is expressed as follows with complete detail shown in (12):

$$\delta\dot{\mathbf{x}}_{\mathrm{RISS}} = F_{\mathrm{RISS}}\delta\mathbf{x}_{\mathrm{RISS}} + G_{\mathrm{RISS}}w_{\mathrm{RISS}}, \tag{11}$$

where $\delta\dot{\mathbf{x}}_{\mathrm{RISS}}$ is the state vector, F_{RISS} is the 9×9 dynamic coefficient matrix, G_{RISS} is the 9×1 noise coupling vector, and w_{RISS} is the unit variance white Gaussian noise.

$$\delta\dot{\mathbf{x}}_{\mathrm{RISS}} = \begin{bmatrix} \delta\dot{\varphi} \\ \delta\dot{\lambda} \\ \delta\dot{h} \\ \delta\dot{v}_e \\ \delta\dot{v}_n \\ \delta\dot{v}_u \\ \delta\dot{A} \\ \delta\dot{a}_{\mathrm{od}} \\ \delta\dot{\omega}_z \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & \frac{1}{(R_M+h)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{(R_N+h)\cos\varphi} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_{\mathrm{od}}\cos A\cos p & \sin A\cos p & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -a_{\mathrm{od}}\sin A\cos p & \cos A\cos p & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sin p & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\gamma_{\mathrm{od}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\beta_z \end{bmatrix} \begin{bmatrix} \delta\varphi \\ \delta\lambda \\ \delta h \\ \delta v_e \\ \delta v_n \\ \delta v_u \\ \delta A \\ \delta a_{\mathrm{od}} \\ \delta\omega_z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \sqrt{2\gamma_{\mathrm{od}}\sigma_{\mathrm{od}}^2} \\ \sqrt{2\beta_z\sigma_z^2} \end{bmatrix} w_R. \tag{12}$$

In order to provide optimal estimation of the above error state vector $\delta\dot{\mathbf{x}}_{\mathrm{RISS}}$, observations for the above system can be provided in the following form:

$$\delta\mathbf{z} = H\delta\mathbf{x}_{\mathrm{RISS}} + \gamma, \tag{13}$$

where $\delta\mathbf{z}$ is the observations vector giving the difference between the RISS and GPS positions and velocities, H is the design matrix giving the ideal noiseless relationship between the observations vector and the state vector, and γ is the vector of observations random noise, which is assumed to be white sequence not correlated with the RISS system noise. For the RISS proposed in this study, the parameters of the measurement model are given as follows:

$$\delta\mathbf{z} = \begin{bmatrix} \varphi_{\mathrm{RISS}} - \varphi_{\mathrm{GPS}} \\ \lambda_{\mathrm{RISS}} - \lambda_{\mathrm{GPS}} \\ h_{\mathrm{RISS}} - h_{\mathrm{GPS}} \\ v_{e,\mathrm{RISS}} - v_{e,\mathrm{GPS}} \\ v_{n,\mathrm{RISS}} - v_{n,\mathrm{GPS}} \\ v_{u,\mathrm{RISS}} - v_{u,\mathrm{GPS}} \end{bmatrix}. \tag{14}$$

The measurement design matrix H would be 6×9 for the position and velocity error states, and can be written as follows:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{15}$$

The covariance of the measurement noise matrix R would be a 6×6 matrix consisting of the position and velocity measurement error covariance. Figure 2 shows the loosely coupled RISS/GPS KF integration scheme.

3. Map Matching

Map matching (MM) is the process of utilizing a digital road network map database to improve the predicted position errors during integration. Motivated by the simplicity and drawbacks of KF, this research will focus on reducing the KF integration errors by utilizing MM. The goal of MM is to match the estimated location with the road network map. There have been many different approaches and algorithms

to the MM problem that have been researched [26, 27]. This paper focuses on three main algorithms from [31]. These are point-to-point matching, point-to-curve matching, and curve-to-curve matching.

The point-to-point matching algorithm is basically like a search problem [30]. The algorithm matches the estimated location, P, to the closest node or point in the network. This could take a lot of time to calculate the distance from P to every node in the network. Therefore, the user must identify those nodes that are within a certain distance of P, and only calculate those distances. The distance is dependent on the type of data being use and it is up to the user to determine it. In this research, a distance of 1000 meters from the current prediction solution was used, which will be discussed later in the next section. Point-to-point matching is very simple to implement and fast, but it does have some problems during execution. The algorithm is very sensitive to how the network is digitized.

The point-to-curve matching algorithm tries to identify the curve (arc) that is closest to P, rather than the point. The same problem arises with the amount of time to calculate the distance from P to every arc in the network. Therefore, the user must identify those arcs that are within a certain distance of P, and only calculate those distances. The network uses a piecewise linear solution to estimate the arcs in the roads, hence the algorithm must find the minimum distance from P to each of the line segments and select the smallest. The method used in the research is a combination of point-to-point matching and point-to-curve matching because of the format of the map data used.

The final approach is curve-to-curve matching, which considers the estimated location as a curve, P, consisting of points $P^0, P^1, P^2, \ldots, P^m$. Then it matches it to the closest arc, which requires some measure of distance between curves. There are different ways to measure the distance between two curves. One way is determining the minimum distance and matching it to that curve. Another technique is measuring the average distance between the curves.

As described above, there are many different techniques for MM. The algorithm is heavily dependent on how the data or network is structured. This was a very important challenge to overcome during this research. The method used in this paper is a combination of point-to-point matching and point-to-curve matching, which is dictated by the format of the map data used.

4. Development of the Augmented KF/MM for RISS/GPS Integration for Land Vehicles

The map data used in this research is integrated with inertial sensor measurements through KF for reliable positioning during GPS outages. The map data was provided by the Queen's university, Kingston ON, which was the 2009 street data as a part of the Arc Geographic Information System (ArcGIS) software produced by Environmental Systems Research Institute Incorporated (ESRI). The data was in shape files that consisted of latitude and longitude coordinates and included all types of road ways: highways, rural roads, and urban roads. The coordinates were the start points and end points of line segments of every road. The data used a piecewise linear solution to estimate the arcs in the roads. Whenever there was a turn in the road a new line segment was started and completed. Therefore the length of the line segments varied depending on how straight or curved the road was. Figure 3 gives a representation of the map data. The size of the data was limited to the area of the trajectory that was being experimented, which was the Kingston area. This was very important because it reduced the actual size of the data, which would affect the process time of the MM algorithm. The map data was a large database of street line segments.

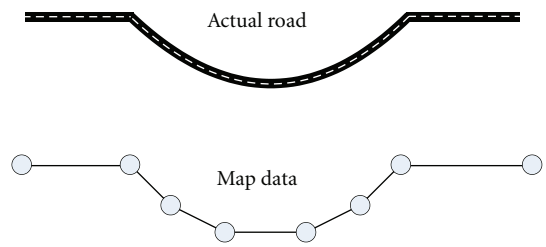


Figure 3: Representation of the map data during a turn.

The initial setup of the map data was a very crucial step in this research. The data was already in latitude and longitude coordinates which was a very good start. It was in a shape file format, which was easily loaded into MATLAB 2009 using the Mapping Toolbox. A shape file is a geospatial vector data format for geographic information systems software. The data was then converted into x and y coordinates in metres. The x and y coordinates are the distances being travelled along the East and North directions. This conversion had to take into account a certain reference point, which was chosen as the start point of the trajectory. The equations below were used for the Easting and Northing calculations into metres:

$$y = \text{North}_{\text{metres}} = (\varphi - \varphi_{\text{initial}})(R_M + h), \tag{16}$$

$$x = \text{East}_{\text{metres}} = (\lambda - \lambda_{\text{initial}})(R_N + h)\cos\varphi, \tag{17}$$

where φ_{initial} and, λ_{initial} are the latitude and longitude of the point chosen to be the origin of the Cartesian coordinates and h is the altitude.

Then the slope (m) and the y-intercept (b) for each line segment were calculated. The slope was calculated using the following equation:

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)}. \tag{18}$$

The y-intercept was calculated using the equation of a straight line, $y = mx + b$, which was rearranged to solve for b as shown below:

$$b = y - (mx). \tag{19}$$

All of these calculations were completed in a simple algorithm with MATLAB, and once completed the results were stored and saved in a database. A representation of how the

Table 1: Map data setup.

Line segment start point		Line segment end ooint		Slope (m)	Y-intercept (b)
X_1	Y_1	X_2	Y_2	m of the line segment	b of the line segment
X_2	Y_2	X_3	Y_3	m of the line segment	b of the line segment
X_3	Y_3	X_4	Y_4	m of the line segment	b of the line segment
X_4	Y_4	X_5	Y_5	m of the line segment	b of the line segment
⋮	⋮	⋮	⋮	⋮	⋮
New road begins					
X_1	Y_1	X_2	Y_2	m of the line segment	b of the line segment

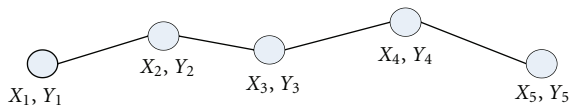


Figure 4: Road using four line segments.

data was stored is shown in Table 1. The first four rows in Table 1 represent a road or street that contains four line segments and five sets of coordinates as shown in Figure 4. The database was quite large and would be used as look-up table as part of the MM algorithm.

The MM algorithm is greatly dependent on the accuracy of the map data. The data acquired was 2009 street data which seems fairly new but there are more things to consider. Highways and roads always have maintenance and construction frequently going on. An example is Highway 401 near Kingston, ON, which is being expanded to accommodate more lanes of traffic. Changes like this will greatly affect the accuracy of the results of MM. However, regularly updating the MM data would mitigate this effect.

Moreover, the size of the data is another limitation of using map data. The size of the data for Kingston, ON, and the surrounding area including Gananoque, ON, is approximately 2.1 megabytes. This does not seem very large but when it is being used as a look-up database, processing time will be increased, especially when including larger areas to cover.

4.1. Map Matching Algorithm. The map matching algorithm developed during this research was the main contribution. During GPS outages, the KF solution still had an error drift due to the inertial sensors errors (including bias drift and scale factor instability). The purpose for the development of the MM algorithm was to improve this position error drift during GPS outages. The method used in the paper is a combination of point-to-point matching and point-to-curve matching because of the setup of the map data used. The results will compare the standalone KF results to the KF/MM results.

Figure 5 is a flowchart of the MM algorithm that was developed. The algorithm consists of five steps. Initially when there is a GPS outage, the KF will go into the prediction stage, and it will predict the position errors, velocity errors, and the azimuth errors based on the dynamic error model.

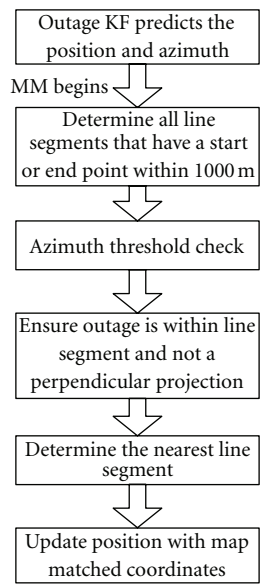


Figure 5: Map matching algorithm flowchart.

The position, velocity, and azimuth components are then obtained after removing these errors. The MM algorithm will be then called as shown in Figure 5.

The first step is to determine all the line segments that have a start or end point within 1000 metres from the GPS outage, and store these line segments. The second step is the azimuth threshold check, which stores all the remaining line segments that pass this check. The third step is to ensure that the GPS outage is within the line segment and does not perpendicularly intersect the line outside of the line segment. The fourth step is determining the nearest line segment from the GPS outage. This step could contain many line segments or only a few, depending on how many segments made it through the first three steps. Final step is the map matching step where position, latitude, and longitude are updated or matched with the coordinates on the nearest line segment. These five steps will be discussed in detail in the next five sub-sections.

4.1.1. Finding All Line Segments within a Certain Distance. The first step in the MM algorithm is to determine all the line segments that have a start point or end point within 1000

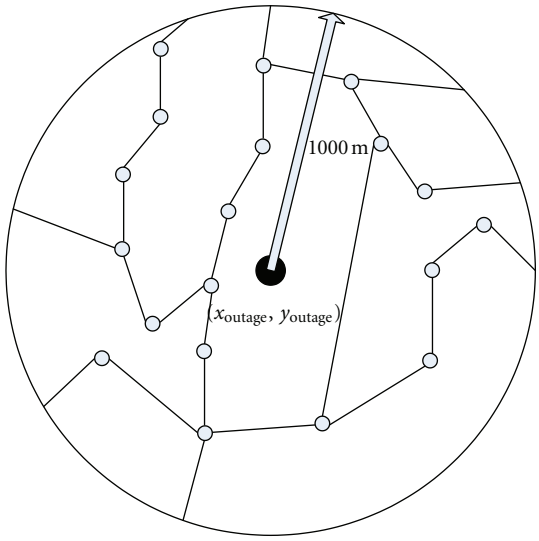


Figure 6: Map matching representation after step one.

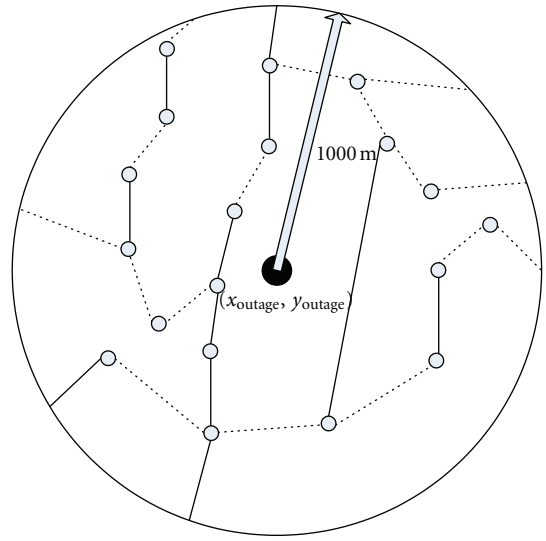


Figure 7: Map matching representation after step two.

metres from the GPS outage. When a line segment meets these criteria, all the line segment information, start and end point coordinates, slope, and y-intercept are stored in a new database. The distance equations used are given as follows:

$$\begin{aligned} \text{Distance}_{\text{start}} &= \sqrt{\left(x_{\text{outage}} - x_1\right)^2 + \left(y_{\text{outage}} - y_1\right)^2}, \\ \text{Distance}_{\text{end}} &= \sqrt{\left(x_{\text{outage}} - x_2\right)^2 + \left(y_{\text{outage}} - y_2\right)^2}, \end{aligned} \tag{20}$$

where the GPS outage latitude and longitude coordinates are converted to x_{outage} and y_{outage} using (16) and (17), respectively.

Figure 6 is the start of a map matching example that will be used to demonstrate the five steps of the MM algorithm. After step one, the line segments are reduced to the segments that have a start or end point within 1000 metres from the GPS outage location.

As displayed in Figure 6, there is a possibility to have many line segments that meet the criteria in step one of the algorithm which are stored and carried over to step two.

4.1.2. Azimuth Threshold Check. The second step of the MM algorithm is the azimuth angle threshold check. Azimuth or heading is defined as the horizontal angle measured clockwise from any fixed reference plane.

During this step the azimuth angle of each line segment, carried over from step one, is calculated. The azimuth angle is calculated as follows:

$$\begin{aligned} dx &= x_2 - x_1, \\ dy &= y_2 - y_1, \\ Az_{\text{line segment}} &= \tan^{-1}\left(\frac{dx}{dy}\right). \end{aligned} \tag{21}$$

Both directions of the line segment are compared to predicted KF azimuth angle. These two directions have heading of $Az_{\text{line segment}}$ and $Az_{\text{line segment}} + 180^\circ$. The threshold of azimuth verification could be changed but typically a threshold of 40° was used. Only the line segments below this threshold are kept and the rest are rejected. Figure 7 indicates the line segments with dashes that are removed by the azimuth threshold check.

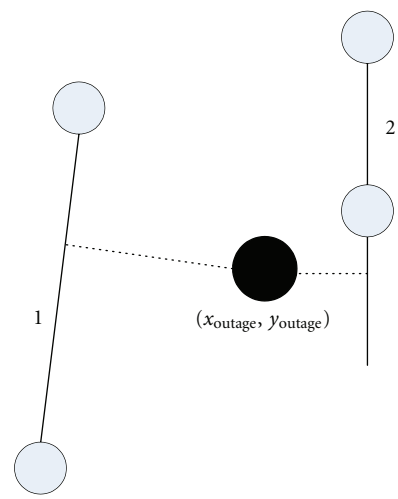


Figure 8: Perpendicular line verification.

4.1.3. GPS Outage Position Check. The third step of the algorithm is to find between which line segments the GPS outage actually lies. Figure 8 gives an excellent example of the perpendicular line verification. The outage is closer to line segment 2 but it is not in between the two points of the segment. This step will remove the line segments that the GPS outage does not fall within. However, this is done with a tolerance so that line segments which are much closer could also be considered.

This verification is completed by calculating the coordinates where a perpendicular line from the GPS outage would

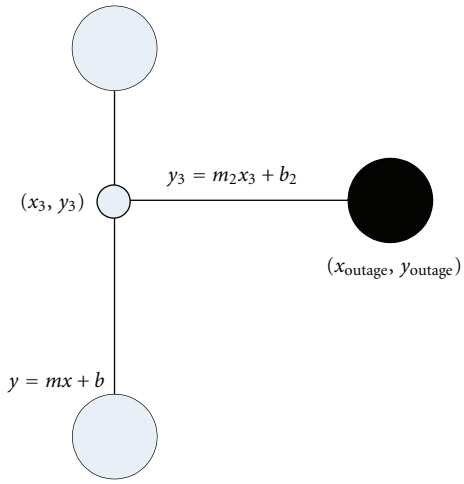


Figure 9: Representation of perpendicular intersection.

intersect the line segment. If these coordinates fall within the line segment, that line segment is stored and carried over to the fourth step. The following equations are used to complete this step. We first start with calculating the normal slope of the perpendicular line as follows:

$$m_2 = -\left(\frac{1}{m_{\text{line segment}}}\right). \tag{22}$$

This is then followed by calculating the y-intercept of the perpendicular line with respect to the GPS outage,

$$b_2 = y_{\text{outage}} - \left(m_2 x_{\text{outage}}\right). \tag{23}$$

Consequently, if the two straight lines are made equal, x_3 can be solved for,

$$\begin{aligned} mx_3 + b &= m_2x_3 + b_2, \\ x_3 &= \frac{(b_2 - b)}{(m - m_2)}. \end{aligned} \tag{24}$$

To solve for y_3, just use the equation of a straight line,

$$y_3 = mx_3 + b. \tag{25}$$

In the above equation, (x_3, y_3) are the coordinates where a perpendicular line from the outage would intersect the line segment. Figure 9 gives an illustration of the above procedure.

Then to verify if (x_3, y_3) are on the line segment, the length of the line segment is compared to the distance of the start point to (x_3, y_3). The same comparison is done with the end point of the line segment. If the length of the line segment is greater than both, then the line segment is stored and carried over to step four.

Figure 10 displays the line segments that are removed (dashed line segments) by the outage position check.

4.1.4. Determine the Nearest Line Segment. The fourth step of the algorithm is just a basic calculation to determine the closest line segment to the GPS outage. The perpendicular distance from the outage to the line segment is calculated as shown:

$$d_{\text{perpendicular}} = \left| \frac{-mx_{\text{outage}} + y_{\text{outage}} - b}{\sqrt{m^2 + 1}} \right|. \tag{26}$$

This perpendicular distance is calculated for all the remaining line segments and then the line segment with the smallest perpendicular distance is selected as the match. Figure 11 displays the line segments that are removed (dashed line segments). The solid line segment is selected for the map matching.

4.1.5. Update the Position. The fifth step is the final step of the proposed algorithm and provides the actual map matching. Here the position, latitude, and longitude are updated or matched with the perpendicular coordinates, (x_3, y_3), on the nearest line segment. This is shown in Figure 12.

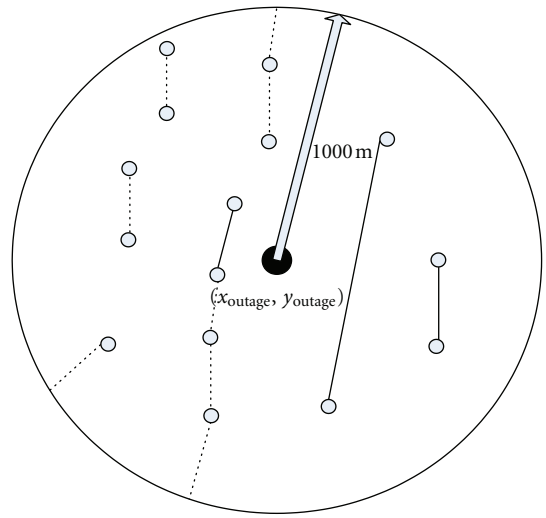


Figure 10: Map matching representation after step three.

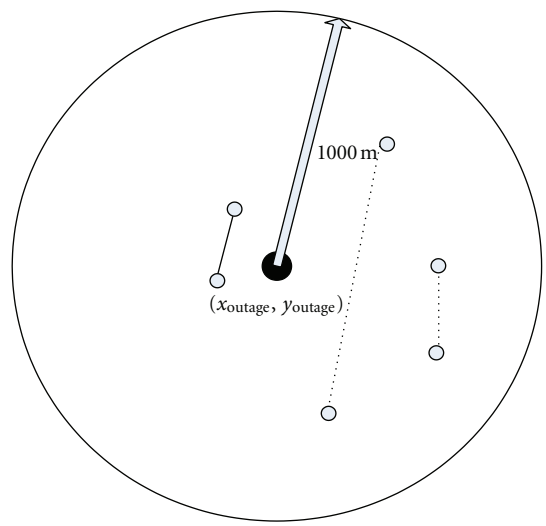


Figure 11: Map matching representation after step four.

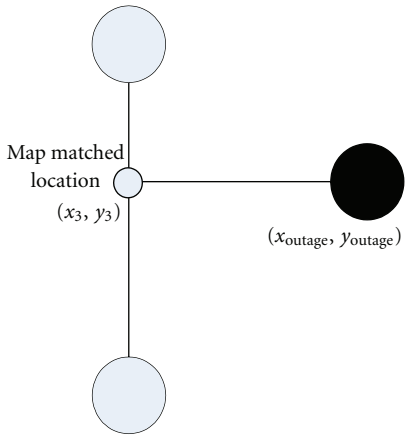


Figure 12: Map matching step.

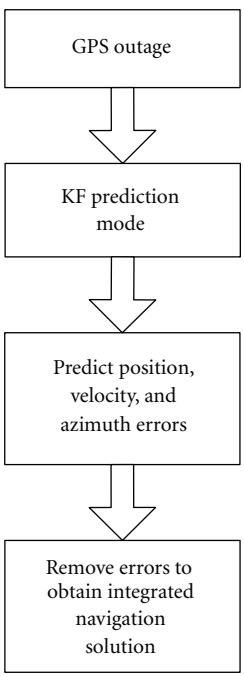


Figure 13: KF prediction stage block diagram.

The (x_3, y_3) coordinates must be converted to latitude and longitude. For the latitude, this is done by rearranging (16) as follows:

$$\varphi = \left(\frac{\text{North}_{\text{metres}}}{R_M + h}\right) + \varphi_{\text{initial}}. \tag{27}$$

For the longitude, the conversion is done by rearranging (17) as follows:

$$\lambda = \left(\frac{\text{East}_{\text{metres}}}{(R_N + h)\cos\varphi}\right) + \lambda_{\text{initial}}, \tag{28}$$

where $\text{North}_{\text{metres}}$ is y_3 and $\text{East}_{\text{metres}}$ is x_3.

4.2. Integration of MM Algorithm with KF. The above MM algorithm is integrated with the KF-based method of RISS/GPS integration. While at least 4 GPS satellites are visible, GPS will provide update to the KF. Initially when there is a GPS outage, the KF will go into the prediction stage and it will predict the position errors, velocity errors, and the azimuth errors based on the dynamic error model. The integrated navigated solution (position, velocity, and azimuth components) is then obtained after removing these errors. Thle above procedure is shown by the block diagram in Figure 13.

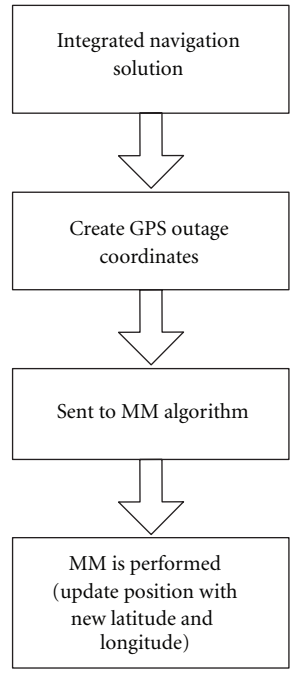


Figure 14: Integration of MM algorithm with KF-based solution.

Figure 15: Data collection equipment mounted inside the road test vehicle.

The latitude and longitude from the integrated navigation solution are sent to the MM algorithm as the GPS outage coordinates. Then the MM algorithm commences and the integrated navigation solution is updated by the map matched position. This is shown by the block diagram in Figure 14.

The map matching algorithm discussed above is very intuitive which was implemented with the KF algorithm that was already developed by our research group. As will be

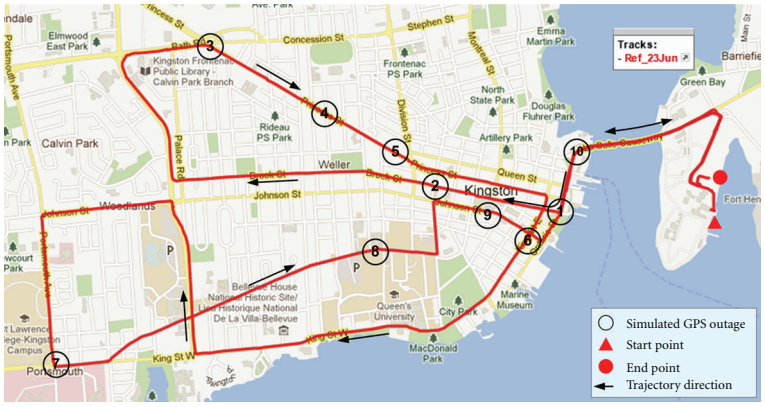


Figure 16: Kingston downtown trajectory, ten outages are shown by circles.

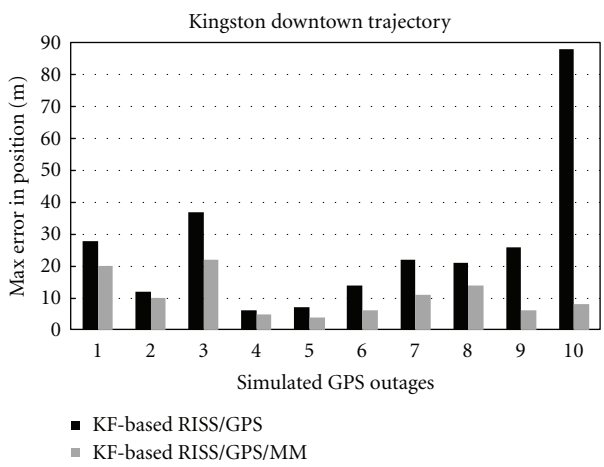


Figure 17: Maximum error in position (m) for Kingston downtown trajectory.

shown in the next section, the KF results could not compensate all the errors caused by the inertial sensors whereas MM algorithm mitigated most of the errors and improved the navigational solution to a great extent which was limited mostly by the accuracy of the map data used.

5. Experimental Results

This section introduces the equipment used and describes the road tests performed to assess the efficacy of the MM algorithm. The results will be shown with the different trajectories that were examined. The focus will be on Kingston area trajectories, including downtown, rural, and highway driving. The results of the proposed method, augmented KF/MM integration, will be discussed in detail and compared to the results of the traditional method of KF-based RISS/GPS integration for land vehicles. The developed method was examined through real road test trajectories by introducing GPS outage at various places encompassing the scenarios of a typical road trajectory.

Crossbow IMU300CC MEMS-based inertial sensors were used for the experiments [41]. The IMU is a six-degree-of-freedom inertial system that uses solid-state devices to measure the angular rate and linear acceleration. This IMU was utilized in RISS architecture, and the performance was examined on real road data collected over various trajectories.

The reference solution used to evaluate the proposed method is based on the Honeywell HG1700 AG11 high-end tactical-grade IMU. This IMU was integrated with the NovAtel GPS receiver using an off-the-shelf assembly, the G2 Pro-Pack SPAN unit, also developed by NovAtel [42]. This integrated system provides a tightly coupled RISS/GPS navigation solution, which was used as the reference for comparisons of the proposed methods. The forward speed (odometer data) was gathered from the vehicle's built in sensors and collected by the On-Board Diagnostics version II (OBD II) interface using a device called CarChip [43]. The setup inside the road test vehicle is shown in Figure 15. It may be noted that GPS used for the system is of higher quality; however, the focus of the paper was not to see the performance of the algorithm during inaccurate readings of GPS but the ability of the algorithm to bridge the complete GPS outages. Since the outages were simulated, the quality of GPS is not a main factor to consider here, especially when the outages simulate total blockage of the GPS signals.

It may be noted that the trajectory figures were created using GPS Visualizer [44] which uses Google maps and suffers from small errors due to which even the reference trajectories sometimes seems off road, especially when zoomed in. However, for calculation purposes, reference trajectory is considered as the best solution. Another point worth noting is that the map data does not match perfectly with the reference solution. One of the most obvious reasons for this is that the map data uses a piecewise linear solution to estimate the arcs in the roads, whereas the reference solution was taken from the integrated RISS/GPS solution provided by the NovAtel SPAN unit which is mounted inside the vehicle and produces data at 100 Hz producing which is virtually continuous. Also, the map data used one single line segment (down the centre line) for urban and rural streets. It did not have a line segment for both directions. Therefore, there is

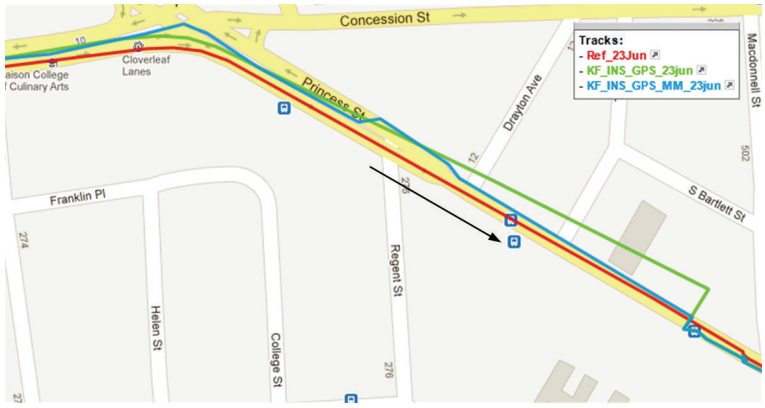


Figure 18: GPS outage three during the Kingston downtown trajectory.

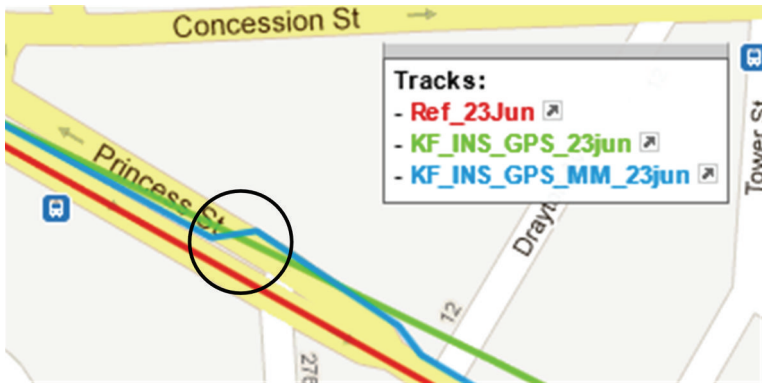


Figure 19: Zoomed-in portion of MM error during GPS outage three.

already a small margin of error between the map data and the reference solution.

5.1. Kingston Downtown Trajectory. The first trajectory examined pertained to downtown Kingston, ON. The majority of this trajectory is urban driving. There were ten intentionally introduced GPS outages of 60 seconds to examine the performance during the outages, focusing on areas like sharp turns and curves in the road which represent the most demanding scenarios for the proposed MM approach. Another challenging feature of this trajectory was its variable speed with frequent stops and sudden accelerations. The velocity of the vehicle was constantly changing due to traffic lights, pedestrians, and sharp turns. Figure 16 displays the downtown Kingston trajectory with GPS outages depicted with circles during which the results of the proposed KF/MM method were compared to the standalone KF and reference solutions.

The maximum position error (meters) for all the ten outages of the downtown Kingston trajectory are shown by the bar graph in Figure 17. Both solutions, the KF based and KF/MM based, were compared against the reference solution, which is shown in Figure 16. It may be noticed that the proposed KF/MM method showed an improvement over the standalone KF method in all the ten GPS outages. During outage 10, the proposed KF/MM showed the most improvement (90%) which happened to be during a turn. The KF solution had a maximum position error of 88 m whereas the KF/MM solution differed only 8 m from the reference trajectory. For GPS outage seven, we observed a maximum position error of 22 m for the KF-based solution and only 11 m for the KF/MM solution which is an improvement of 50%. We will now take a closer look at three of the GPS outages and discuss how MM greatly improved the results.

GPS outage three is a good example of an outage occurring during a turn at a higher speed which is depicted in Figure 18. The trend of KF solution is similar to GPS outage one where it constantly drifts away from the reference whereas KF/MM solution limits the error growth.

It should be observed that the KF/MM solution matches the wrong road about half way through the GPS outage. This portion of Figure 18 is zoomed in and shown in Figure 19. This is a unique situation because there is a small separation between two streets until they meet to make one street. The two streets have the same azimuth angle or heading, therefore they would both pass the azimuth threshold check during the MM algorithm. When the first match occurred to the wrong street, displayed by the circle, the KF-based solution is closer to the wrong street until the two streets connect together. Therefore, the MM algorithm is actually operating correctly but in this unique situation the algorithm is matching to the

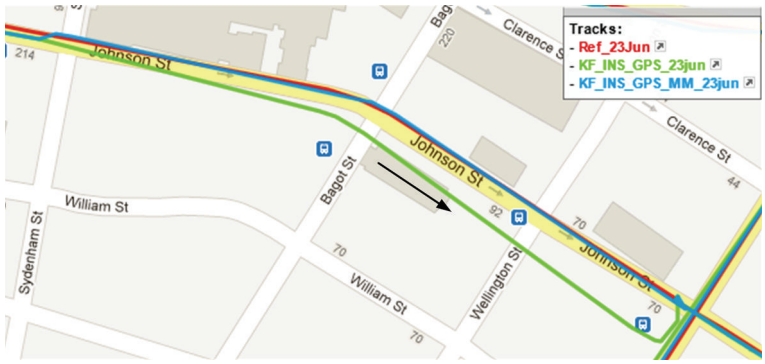


Figure 20: GPS outage nine during the Kingston downtown trajectory.

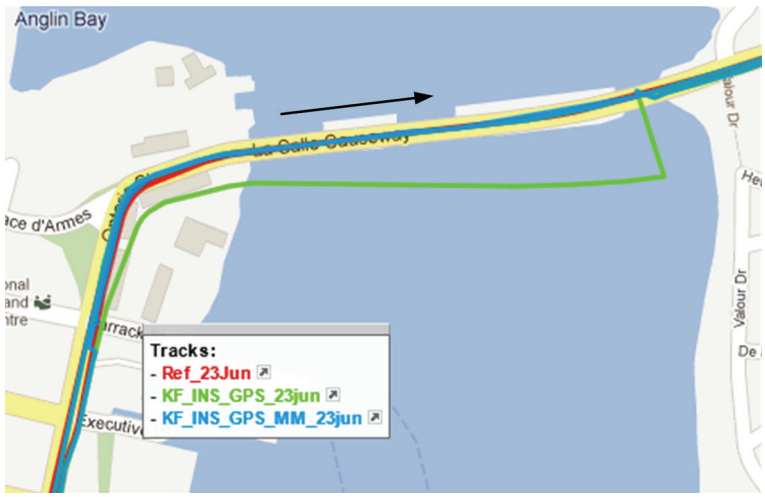


Figure 21: GPS outage ten during the Kingston downtown trajectory.

wrong street for approximately 2 seconds. A possible way to correct this is to use the previous map matched position and the velocity. If the distance from the new map-matched position to the previous map matched position is too large for the velocity being travelled, the algorithm would choose the second nearest line segment.

Figure 20 shows a closer look at GPS outage nine. The KF-based solution has a maximum position error of 26 m and the KF/MM-based solution has a maximum error of 6 m, which is an improvement of 77%. It is easily recognizable that the KF-based solution (green) has a constant error drift from the road or reference solution. This is due to the MEMS sensors which have a constant error drift over time. The developed method of KF/MM (blue) limited the error drift, by constantly matching the position back to the actual road. The MM is performed at every iteration during the outage.

GPS outage ten is the best example of how the MM algorithm improved the overall accuracy of the position information. Figure 21 shows the performance during GPS outage ten, which starts during a sharp turn before the LaSalle causeway. The KF-based solution has a maximum position error of 88 m and the KF/MM solution has a maximum position error of only 8 m. This is an improvement of 90%. The KF-based solution (green) is constantly drifting from the reference and the KF/MM-based solution is limiting the error growth by restricting it to the actual road.

The KF/MM has shown promising results for the Kingston downtown trajectory where the majority of the GPS outages occurred on urban road ways. The next trajectory, which took place in Kingston suburbs, is chosen to assess the performance of the proposed algorithm in mostly rural areas with some urban portions.

5.2. Kingston Suburbs Trajectory. The second trajectory that was examined was a trajectory of the suburbs of Kingston, ON. The majority of this trajectory is rural with some urban driving. There were ten intentionally introduced GPS outages of 60 seconds to examine the performance during the outages, focusing on areas like sharp turns and curves on the road ways. Another great feature about this trajectory is that some of the GPS outages could be simulated at much higher speeds reaching 80 km/hr. Figure 22 displays the Kingston suburbs trajectory with the ten circles marking the location of the simulated GPS outages during which the positional error of KF and KF/MM is compared. The

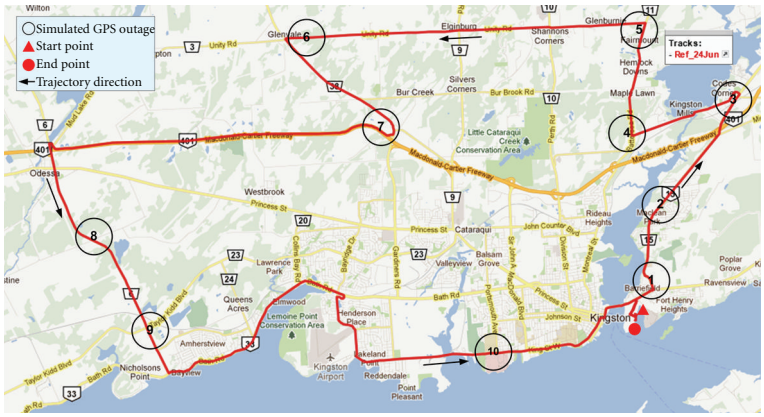


Figure 22: Kingston suburbs trajectory, ten outages are shown by circles.

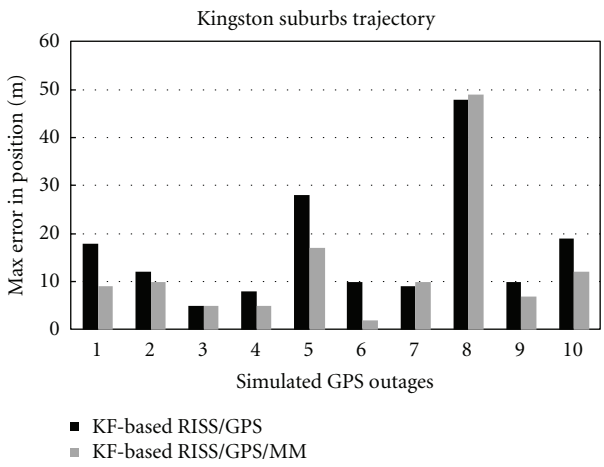


Figure 23: Maximum error in position (m) for the Kingston suburbs trajectory.

maximum position error (meters) for the Kingston suburbs trajectory is shown in the bar graph of Figure 23.

Both solutions, the KF based and KF/MM based, were compared against the reference solution, which is shown in Figure 22. The proposed KF/MM method showed an improvement over the KF-based solution in all GPS outages except GPS outage 8 which will be examined in detail later. It can be seen that GPS outage five, which was simulated during a sharp turn (greater than 90°), has a maximum position error of 28 m for the KF-based solution and 16 m for the KF/MM based solution. This is an improvement of 43% or 12 m in positional error. GPS outage six has a maximum position error of 10 m for the KF-based solution and only 2 m for the KF/MM-based solution, which is an improvement of 80%.

In GPS outage eight, the KF-based solution had a maximum position error of 48 m and the KF/MM-based solution also had a large maximum position error of 49 m. As shown in Figure 24, the KF-based solution (green) has a familiar drift which takes the solution away from the road or the reference solution. The developed KF/MM method does correct this error drift throughout the outage except for one iteration as shown in Figure 25. For this one iteration the MM algorithm selects the wrong road; matched to the intersecting road (Caton Road). This is the reason why both solutions have a large maximum position error. During the next iteration, the algorithm corrected itself and matched back to the correct road. It may be visualized that KF/MM is still better then KF because it stays close to the reference except for a short time whereas KF starts to go away from the reference right from the onset of the outage. This is evident from the RMS error of the outage which is only 19 m for KF/MM-based solution as compared to 31 m for KF-based solution.

GPS outage nine is another example of how MM is restricting the positioning solution of the developed KF/MM method to the actual road. The KF-based solution has a maximum position error of 10 m and the KF/MM-based solution has a maximum error of 7 m, which is an improvement of 30%. This GPS outage occurred during a straight stretch with an average velocity of 50 km/hr.

5.3. Gananoque Trajectory. The third trajectory was conducted mostly in rural areas with slow to medium speeds in straight as well as winding patches of the road. This trajectory is called Gananoque trajectory as it passes through this town and is shown in Figure 26.

There were ten intentionally introduced GPS outages of 60 seconds to examine and compare the performance of the algorithms. The insertion of the outages was carefully planned such that they include straight portions, sharp turns, and curves on the road ways. This trajectory also included areas where outages could be simulated at higher speeds reaching up to 80 km/hr. Figure 26 displays the Gananoque trajectory with the ten GPS outages which were included for performance analysis.

As shown by the bar graph in Figure 27, the developed KF/MM method greatly improved the accuracy of the results. RISS/GPS integration for land vehicles using the developed method of KF/MM had an average maximum position error of 13.5 m and the KF-based solution had an average

Figure 24: GPS outage eight during the Kingston suburbs trajectory.

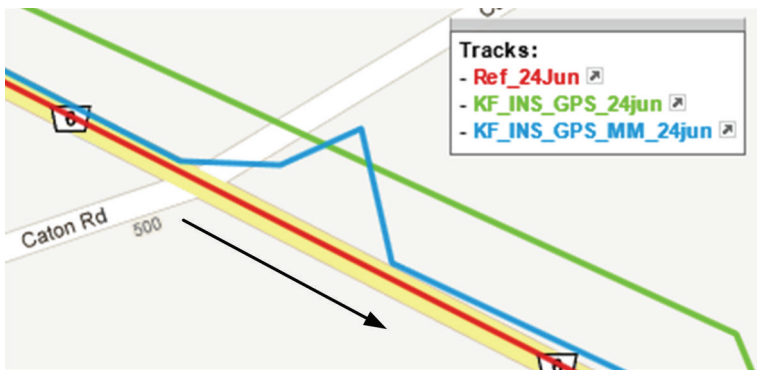


Figure 25: MM error during GPS outage eight.

maximum position error of 25.8 m. This is an overall average improvement of 46%.

We will now take a closer look at the GPS outage which was simulated on highway 2 during a slight turn with an average velocity of 80 km/hr. During this outage, the KF-based solution has a maximum position error of 22 m and the KF/MM-based solution has a maximum error of 11 m which is an improvement of 50%. Although there are errors in the KF/MM solution due to piecewise approximation of the road inherent in the map data, the output trajectory is mostly within the road boundaries.

GPS outage ten had the most significant improvement. The KF-based solution has a maximum position error of 52 m while the developed KF/MM based solution has a maximum position error of 12 m. This is a large improvement of 77% or 40 m in accuracy.

6. Conclusion

This paper focused on reducing the KF-based RISS/GPS integration errors by augmenting it with MM. MM limited the error growth during GPS outages by restricting the position solution to the road network. This was accomplished by using digital maps of the road networks as a constraint in the integration process. To reduce the errors contributed by the low-cost inertial sensors, a reduced inertial sensor system was used where only one gyroscope and two accelerometers were used along with built-in odometer of the vehicle, which constitute the 3D RISS. Owing to its simplicity and comparable accuracy during good satellite visibility, loosely coupled integration was used for integration of inertial sensors and odometer with GPS. The proposed method, augmented KF/MM for 3D RISS/GPS integration, was tested on three disparate trajectories by simulating ten GPS outages in each trajectory at various locations including straight portions, slight turn as well as sharp bends. It was also ensured to include different dynamics by choosing low and high speeds, stops, and sudden accelerations. The results of the proposed method were analyzed in detail and compared with the traditional method of KF-based 3D RISS/GPS integration for land vehicles. To elucidate the comparison and clarify the exceptions to the performance of the proposed algorithm, individual outages were discussed. It was found that the proposed method outperformed the KF solution in all the three trajectories with a clear margin despite being dependent on the accuracy of the map data. For the first trajectory, the average improvement in maximum position error of the KF/MM method over KF-only method was 59%. For the second trajectory and third trajectory, it was 30% and 46%, respectively. By this account, overall, the average

Figure 26: Gananoque trajectory, ten outages are shown by circles.

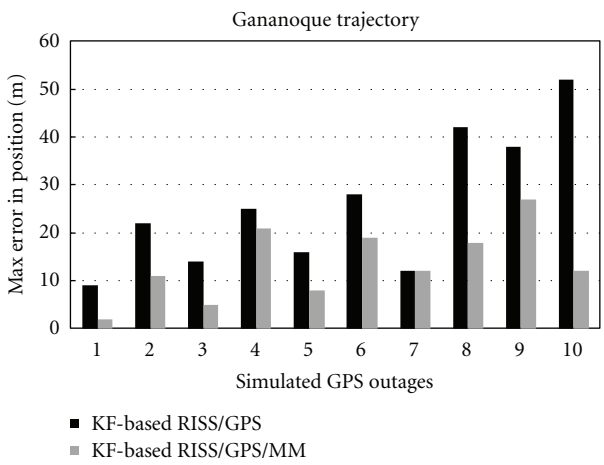


Figure 27: Maximum error in position (m) for the Gananoque trajectory.

improvement in maximum position error of the proposed method over traditional KF-only method was about 45%. There were few instances where the apparent performance of the proposed algorithm was poorer. These were highlighted and probable reasons and possible solutions were furnished where required.

References

[1] P. D. Groves, *Principles of GNSS, Inertial, and Multi-Sensor Integrated Navigation Systems*, Artech-House, Boston, Mass, USA, 2008.

[2] J. A. Farrell, *Aided Navigation : GPS with High Rate Sensors*, McGraw-Hill, New York, NY, USA, 2008.

[3] M. S. Grewal, L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2007.

[4] K. R. Britting, *Inertial Navigation Systems Analysis*, John Wiley & Sons, New York, NY, USA, 1971.

[5] C. Jekeli, *Inertial Navigation Systems with Geodetic Applications*, Walter De Gruyter, Berlin, Germany, 2001.

[6] A. Noureldin, T. B. Karamat, M. D. Eberts, and A. El-Shafie, "Performance enhancement of MEMS-based INS/GPS integration for low-cost navigation applications," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 3, pp. 1077–1096, 2009.

[7] J. Bernstein, "An overview of MEMS inertial sensing technology," *Sensors*, vol. 20, no. 2, pp. 14–21, 2003.

[8] C. Goodall, "Intelligent integration of a MEMS IMU with GPS using a reliable weighting scheme," in *Proceedings of the 18th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS '06)*, pp. 1661–1670, Fort Worth, Tex, USA, September 2006.

[9] D. H. Titterton and J. Weston, *Strapdown Inertial Navigation Technology*, American Institute of Aeronautics and Astronautics, New York, NY, USA, 2nd edition, 2005.

[10] A. Lawrence, *Modern Inertial Technology: Navigation, Guidance, and Control*, Springer, New York, NY, USA, 2nd edition, 1998.

[11] P. Aggarwal, Z. Syed, A. Noureldin, and N. El-Sheimy, *MEMS-Based Integrated Navigation*, Artech House, Norwood, Mass, USA, 2010.

[12] J. Georgy, A. Noureldin, M. J. Korenberg, and M. M. Bayoumi, "Low-cost three-dimensional navigation solution for RISS/GPS integration using mixture particle filter," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 2, pp. 599–615, 2010.

[13] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, vol. 1, Academic Press, New York, NY, USA, 1979.

[14] G. Minkler and J. Minkler, *Theory and Application of Kalman Filtering*, Magellan Book, Palm Bay, Fla, USA, 1993.

[15] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer, New York, NY, USA, 2001.

[16] B. Ristic and S. Arulampalam, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech-House, Boston, Mass, USA, 2004.

[17] J. Georgy, A. Noureldin, Z. Syed, and C. Goodall, *Nonlinear Filtering for Tightly Coupled RISS/GPS Integration*, Indian Wells, Ca, USA, 2010.

[18] J. Georgy, T. Karamat, U. Iqbal, and A. Noureldin, "Enhanced MEMS-IMU/Odometer/GPS Integration Using Mixture Particle Filter," *GPS Solutions*, vol. 15, no. 3, pp. 239–252, 2011.

[19] N. El-Sheimy, K. W. Chiang, and A. Noureldin, "The utilization of artificial neural networks for multisensor system integration in navigation and positioning instruments," *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 5, pp. 1606–1615, 2006.

[20] R. Sharaf and A. Noureldin, "Sensor integration for satellite-based vehicular navigation using neural networks," *IEEE Transactions on Neural Networks*, vol. 18, no. 2, pp. 589–594, 2007.

[21] W. Abdel-Hamid, A. Noureldin, and N. El-Sheimy, "Adaptive fuzzy prediction of low-cost inertial-based positioning errors," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 3, pp. 519–529, 2007.

[22] L. Semeniuk, *Bridging global positioning system outages using neural network forward prediction of inertial navigation position and velocity errors*, M.S. thesis, Department of Electrical and Computer Engineering, Royal Military College of Canada, Kingston, Canada, 2006.

[23] K. W. Chiang, A. Noureldin, and N. El-Sheimy, "A new weight updating method for INS/GPS integration architectures based

on neural networks," *Measurement Science and Technology*, vol. 15, no. 10, pp. 2053–2061, 2004.

[24] N. El-Sheimy, W. Abdel-Hamid, and G. Lachapelle, "An adaptive neuro-fuzzy model for bridging GPS outages in MEMS-IMU/GPS land vehicle navigation," in *Proceedings of the 17th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS '04)*, pp. 1088–1095, Long Beach, Ca, USA, 2004.

[25] W. B. Zavoli and S. K. Honey, "Map matching augmented dead reckoning," in *Proceedings of the 36th IEEE Vehicular Technology Conference.*, pp. 359–362, Dallas, Tex, USA.

[26] C. Fouque, P. Bonnifait, and D. Betaille, *Enhancement of Global Vehicle Localization Using Navigable Road Maps and Dead-Reckoning*, Monterey, Ca, USA, 2008.

[27] Y. Cui and S. S. Ge, "Autonomous vehicle positioning with GPS in urban canyon environments," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 1, pp. 15–25, 2003.

[28] P. Davidson, M. A. Vazquez, and R. Piche, *Uninterrupted Portable Car Navigation System Using GPS, Map and Inertial Sensors Data*, Piscataway, NJ, USA, 2009.

[29] O. Pink and B. Hummel, "A statistical approach to map matching using road network geometry, topology and vehicular motion constraints," in *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems (ITSC '08)*, pp. 862–867, Piscataway, NJ, USA, December 2008.

[30] C. E. White, D. Bernstein, and A. L. Kornhauser, "Some map matching algorithms for personal navigation assistants," *Transportation Research Part C*, vol. 8, no. 1–6, pp. 91–108, 2000.

[31] D. Bernstein and A. Kornhauser, "An introduction to map matching for personal navigation assistants," Tech. Rep., New Jersey Institute of Technology, New Jersey, NJ, USA, 1996.

[32] J. Georgy, U. Iqbal, and A. Noureldin, "Quantitative comparison between Kalman filter and particle filter for low cost INS/GPS integration," in *Proceedings of the 6th International Symposium on Mechatronics and its Applications (ISMA '09)*, pp. 1–7, Sharjah, UAE, 2009.

[33] R. G. Brown and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering: with MATLAB Exercises and Solutions*, John Wiley & Sons, New York, NY, USA, 3rd edition, 1997.

[34] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice Using MATLAB*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2001.

[35] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, vol. 2, Academic Press, New York, NY, USA, 1982.

[36] A. Gelb, Ed., *Applied Optimal Estimation*, MIT Press, Cambridge, Mass, USA, 1974.

[37] T. Karamat, J. Georgy, U. Iqbal, and A. Noureldin, "A tightly-coupled reduced multi-sensor system for urban navigation," in *Proceedings of the 22nd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS '09)*, pp. 177–187, Savannah, Ga, USA, September 2009.

[38] U. Iqbal, A. F. Okou, and A. Noureldin, "An integrated reduced inertial sensor system—RISS / GPS for land vehicle," in *Proceedings of the IEEE/ION Position, Location and Navigation Symposium (PLANS '08)*, pp. 1014–1021, Monterey, Ca, USA, May 2008.

[39] J. Georgy, U. Iqbal, M. Bayoumi, and A. Noureldin, *Reduced Inertial Sensor System (RISS)/GPS Integration Using Particle Filtering for Land Vehicles*, Savannah, Ga, USA, 2008.

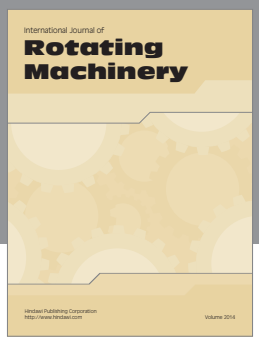
[40] U. Iqbal, T. B. Karamat, A. F. Okou, and A. Noureldin, "Experimental results on an integrated GPS and multisensor system for land vehicle positioning," *International Journal of Navigation and Observation*, vol. 2009, Article ID 765010, 18 pages, 2009.

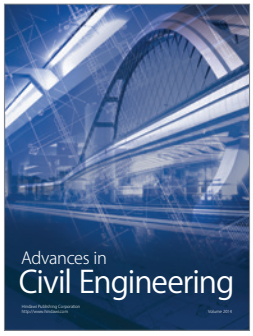
[41] IMU300—6DOF Inertial Measurement Unit: Crossbow Technology Inc., 2011, http://www.davisnet.com/product_documents/drive/spec_sheets/8211-21-25_carchip_specsB.pdf.

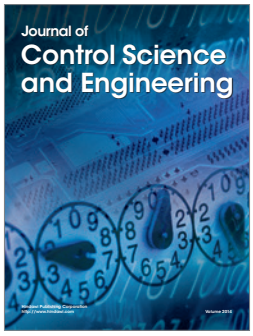
[42] SPAN Technology System User Manual OM-20000062: NovAtel Inc., 2011, http://www.novatel.com/Documents/Manuals/om-20000062.pdf.

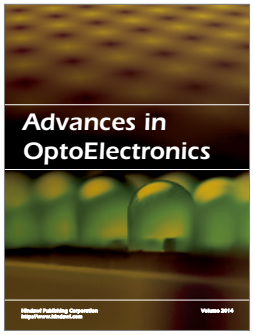
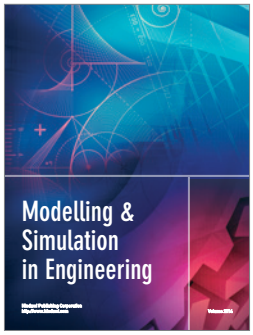
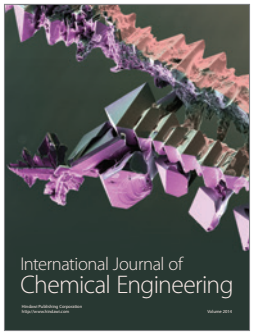
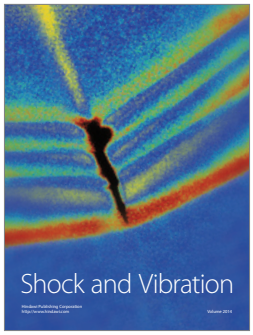
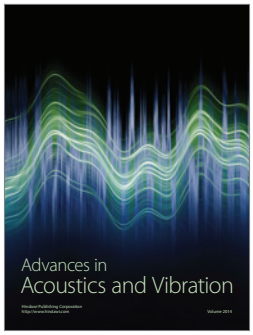
[43] CarChip OBDII-Based Vehicle Data Logger and Software: Davis Instruments, 2011, http://www.davisnet.com/product_documents/drive/spec_sheets/8211-21-25_carchip_specsB.pdf.

[44] A. Schneider, Draw a Google Map from a GPS file, 2008, http://www.gpsvisualizer.com/map_input?form=google.







Hindawi

Submit your manuscripts at
http://www.hindawi.com