Research Article

GNSS/Low-Cost MEMS-INS Integration Using Variational Bayesian Adaptive Cubature Kalman Smoother and Ensemble Regularized ELM

Hassana Maigary Georges, Dong Wang, and Zhu Xiao

College of Computer Science and Electronics Engineering, Hunan University, Changsha 410082, China

Correspondence should be addressed to Zhu Xiao; zhxiao@hnu.edu.cn

Received 9 March 2015; Accepted 12 April 2015

Academic Editor: Carsten Proppe

Copyright © 2015 Hassana Maigary Georges 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.

Among the inertial navigation system (INS) devices used in land vehicle navigation (LVN), low-cost microelectromechanical systems (MEMS) inertial sensors have received more interest because of their price and portability. Kalman filter (KF) based GNSS/INS integration has been widely used to provide a robust solution to the navigation. However, its prediction model cannot give satisfactory results in the presence of colored and variational noise. In order to achieve reliable and accurate positional solution for LVN in urban areas surrounded by skyscrapers or under dense foliage and tunnels, a novel model combining variational Bayesian adaptive Kalman smoother (VB-ACKS) as an alternative of KF and ensemble regularized extreme learning machine (ERELM) for bridging global positioning systems outages is proposed. The ERELM is applied to reduce the fluctuating performance of GNSS during an outage. We show that a well-organized collection of predictors using ensemble learning yields a more accurate positional result when compared with conventional artificial neural network (ANN) predictors. Experimental results show that the performance of VB-ACKS is more robust compared with KF solution, and the prediction of ERELM contains the smallest error compared with other ANN solutions.

1. Introduction

Active safety systems and self-driving cars are a promising solution to reduce the number of traffic accidents [1, 2]. Some advanced driver assistance systems (ADAS) such as adaptive cruise control (ACC), collision warning system (CWS), and emergency braking system (EBS) that already exist in series of land vehicles are able to warn the driver and even to intervene in the state of the vehicle when a hazardous traffic situation is being developed. With the development of various driver assistance systems using vehicle position, the improvement of positioning accuracy is highly desired. Global navigation systems (GNSS), widely used as a positioning system, can provide accuracy of a few meters in suburban areas in order to provide an accurate trajectory or position of vehicles which is crucial for collision warning/avoidance. However, in urban areas or under dense foliage, this accuracy can be degraded to ten meters or more because of the reflection and blocking of global positioning system (GPS) signals by tall buildings. In these environments, signal may be difficult to acquire or the number of satellites available may be not sufficient to provide position information [2]. Thus, a backup or alternative system is required to mitigate the impact of GPS outages.

Hybridization of GNSS with an inertial navigation system (INS) is one of the best solutions for bridging the GNSS gaps. Among the INS devices used in land vehicle navigation (LVN), low-cost microelectromechanical systems (MEMS) inertial sensors have received more interest because of their price and technology enabling compact and portable size [3]. MEMS-INS and GNSS are commonly integrated using Kalman filter (KF) to provide a robust navigation solution, overcoming situations of GPS satellite signals blockage. However, it was shown that long-term GPS outages can be crucial and the INS position accuracy degrades also with time. Additionally, most of the methods proposed in the literature point that the traditional Kalman filter is the main
reason of the failure of the integration of GNSS/INS as its prediction model is not adapted for nonlinear stochastic problems. It is also shown that applying adaptive filters and smoothers techniques to the GNSS/INS integrated navigation system could obtain better estimated performance than by using conventional KF [4, 5]. Vehicular navigation is often characterized with dynamics changes in motion and the use of KF could negatively have an impact on the position accuracy due to its inappropriate stochastic models and its inability to solve nonlinear system problems with variational and colored noise properties [5]. As a consequence of the aforementioned KF drawbacks, different approaches based on Bayesian filtering [6–9] and artificial intelligence (AI) techniques [3, 10–16] have been proposed to improve the effectiveness of the integration methodology in bridging GPS outages.

(1) Research Objectives. In this research, we propose the implementation of ensemble regularized extreme learning machine (ERELM) based GNSS/MEMS-INS integration methodology to improve the accuracy of predictor and achieve better stability through training a set of ELM models and then combining them for final predictions. Unlike the traditional ANN techniques, the extreme learning machine (ELM) offers significant advantages such as fast learning speed and ease of implementation and the ensemble scheme we use is able to avoid the problem of local minima, stopping criteria, and overfitting. Moreover, in ensemble learning, the overfitting can be useful for ELMs. We use the adaptive boost regression and threshold (AdaBoost.RT) scheme which is one of the most popular techniques for generating ensemble ELM models due to its adaptability, its simplicity of implementation, and its ability to handle strong nonlinearity and random variations of dynamic data series. Therefore, to represent the sequential process that describes the input data of GNSS/MEMS-INS integration and deal with variational noise, the combination of a variational Bayesian based adaptive cubature Kalman smoother (VB-ACKS) and ERELM is proposed to estimate INS errors.

(2) Related Works. Recently, several techniques based on artificial neural networks (ANNs) have been proposed to replace the function of KF in order to overcome some disadvantages. As a result of their ability to handle the problem of nonlinearity, some approaches such as those based on multilayer perceptron neural networks (MLPNNs), radial basis function neural networks (RBFNNs) [10, 11], and adaptive neural fuzzy inference system (ANFIS) [12] and others based on random forest regression (RFR) [13] and least square support vector machine (LS-SVM) [14] were reported for GNSS/MEMS-INS systems. To overcome the drawbacks associated with the use of GPS and INS in a standalone manner, two recent AI methods were reported in the literature. The first was proposed by Malleswaran et al. combined recurrent neural network (RNN) with two evolutionary algorithms such as particle swarm optimization (PSO) and genetic algorithm (GA) in order to integrate GNSS and INS. Despite the level of prediction accuracy added by the weight optimization techniques, it could be computationally burdensome for navigation systems. Another GNSS/INS technique is based on Bhatt et al. [15] which uses a Dempster-Shafer theory combined with support vector machine (DS-SVM) [16]. The DS based theory is used for the INS and GPS data fusion and the SVM for modeling INS error. Although DS-SVM learning algorithm performs very well for GPS outages prediction, a critical limitation relies on the fact that finding the correct pair of position samples as required by the technique could be time-consuming. All of the above techniques and models were shown to be able to have good performance due to their great power to model the nonlinear relationship. However, in dealing with the GPS outages problem, these AI techniques have not considered some important aspects of ANN such as the accuracy of their architectures, the problem of weight-decay, and local minima problem. Furthermore, most of them suffer from slow convergence rates, thus limiting their applicability and capability for predicting accurately land vehicles navigation. Saadeddin et al. [17] and Chen and Fang [18] proposed an adaptive method based on ANN and a hybrid prediction method based on radial basis function (RBF), respectively, to overcome the limitation of Kalman filter and the authors focused on the improvement of the predictors integrated in the kernel of the ANN. Combining several predictors as demonstrated by ANFIS, RFR are generally more powerful than using single predictor. However, the accuracy of these techniques degrades when applying low-cost MEMS-INS due to their architecture that does not assure generalization performance [19, 20]. The use of ensemble learning models stands as the best solution because of its performance which is much higher in terms of prediction accuracy compared to the global and local individual models. ELM is a freshly emerging learning architecture that provides fast prediction solutions. ELM stands out from other AI methods because of the following particular characteristics: extremely fast training, good generalization, universal approximation capability, and excellent learning accuracy. It has many advantages, and not only avoids many problems encountered by traditional gradient-based neural network learning algorithms such as local minima and various training parameters (training efficiency, stopping criteria, learning epochs, and the hidden layer unit number), but also learns much faster, with higher generalization performance than the established gradient-based learning methods [19]. In ELM, the hidden nodes are randomly generated and this is done without an iterative tuning procedure. Unlike the traditional learning techniques, ELM can generate the hidden node parameters before seeing the training data [20]. ELM requires lesser training time and it is hence remarkably efficient in terms of computational cost. An important aspect of ELM resides in the activation functions that enable us to achieve universal approximation capability. Unlike traditional AI methods, it does not need to calibrate the parameters. ELM enables us to alleviate several challenging issues in AI such as local minima, trivial human intervention, and time consumption in learning. As mentioned earlier, combining a number of learning machines can reduce the risk of overfitting and lead to better generalization performance. Ensemble neural network methods are more and more desirable due to the basic fact that the selection of the weights represents in itself...
an optimization problem with many local minima [21–23]. Ensemble extreme learning machine methods have received much more attention because of their threefold advantages. First, it allows improving the accuracy of predictors. Second, it alleviates the overtraining problem. Third, it enhances the predictive stability of ELM [21].

The remainder of this paper is organized as follows. Section 2 gives a description of the system overview for GNSS/INS integration which includes the INS mechanization process and the computation of the navigation parameters as well as the novel variational Bayesian based adaptive cubature Kalman smoother used for GNSS/INS integration. Section 3 describes the basic RELM and the ERELM based cubature Kalman smoother used for GNSS/INS integration. As depicted in Figure 1, MEMS-INS and GNSS are fused by VB-ACKS and the variational Bayesian based adaptive smoothing algorithm [4, 5].

2. System Model

The proposed GPS outages bridging technique is realized in low-cost 2D land vehicle tracking system using reduced inertial measurement unit integrated with GPS.

2.1. Computation of Navigation Parameters. Before the pre-processing or postprocessing steps, the GNSS and MEM-INS data have to be converted and mechanized by the conventional methods as described in [3]. The vehicles position, velocity, and attitude that will be later used as input in the AI algorithms can be obtained by the integration of the INS measurements. These measurements are transformed and converted with the well-known coordinate transformation system earth-centered, earth-fixed (ECEF) reference frame. The measured specific force related to the time rate of change of velocity with respect to the ECEF frame is written as

\[ \dot{V}^e = C_e^b F^b - 2\Omega^e V^e - \Omega^2 X^e + G^e, \]  

where \( \dot{V}^e \) is the cosine rotation matrix from body frame to the ECEF frame (e-frame) and \( \Omega^e \) is the skew matrix of the earth's angular velocity. \( X^e \) represents the vehicle's position. \( V^e \) is the vehicle's velocity with respect to the ECEF frame and finally \( G^e \) is the local gravitation vector. \( 2\Omega^e V^e \) is the coriolis acceleration. \( \Omega^2 X^e \) is the centrifugal acceleration. The position of the vehicle with respect to the e-frame can be obtained by integrating the vehicle's velocity as shown in the following equation:

\[ X^e = V^e t. \]  

The attitude with respect to the e-frame is represented in the quaternion vector as

\[ \dot{q}_b^e = [q_1, q_2, q_3, q_4]. \]  

The time rate of change of the attitude of the vehicle is given as follows:

\[ \dot{q}_b^e = \frac{1}{2} \Omega^b q_b^e. \]  

2.2. MEM-INS Measurements Model. The MEMS-INS measurements' model can be represented as in the following equations:

\[ \ddot{f}^b = f^b - b^b_{acc} - w_{acc}, \]

\[ \dot{\omega}^b_{ib} = \omega^b_{ib} - b^b_{gyr} - w_{gyr}, \]  

where \( f^b \) is the measured vehicle specific force in the body frame, \( \ddot{f}^b \) is the measured angular velocity of the vehicle relative to the inertial frame measured in the body frame of the vehicle, \( b_{acc} \) is the bias error in the specific force measurement, \( w_{acc} \) is the bias in the angular velocity measurement, \( b_{gyr} \) is the process noise on the specific force measurement, and \( w_{gyr} \) is the process noise on the angular velocity measurement [8].

3. Proposed Algorithms

3.1. Variational Bayesian Adaptive Cubature Kalman Smoother. After the mechanization process, both MEMS-INS and GNSS parameters are integrated using the traditional KF. However, since the system in land vehicle navigation presents is nonlinear, the use of a smoother is highly required. Furthermore, land vehicle navigation exhibits dynamics changes in motion and the use of KF could negatively impact the position accuracy due to its inappropriate stochastic models and its inability to solve nonlinear filtering problems with variational or colored noise properties. These nonlinear systems can be solved by applying variational Bayesian adaptive cubature Kalman smoother (VB-ACKS) based on the cubature rule which is a variation of Gaussian approximation based smoothing algorithm [4, 5]. As depicted in Figure 1, MEMS-INS and GNSS are fused by VB-ACKS and the
4 Mathematical Problems in Engineering

MEMS-INS
Mechanization
GNSS
VB-ACKS
Position velocity attitude
Final position

Figure 1: Model structure of the VB-ACKS fusion filter for GNSS/INS integration.

The mathematical formulation of the nonlinear discrete-time dynamical system is given as follows:

\[ x_k = f(x_{k-1}) + v_{k-1}, \]

\[ z_k = h(x_k) + w_k, \]  

where \( x_k \) is the state vector of the vehicle, \( z_k \) is the measurement vector, \( f(\cdot) \) and \( h(\cdot) \) are the state and measurement matrices, respectively, of nonlinear system, and \( v_{k-1} \) and \( w_k \) are noise samples from two independent zero-mean Gaussian processes with covariances \( Q_{k-1} \) and \( R_k \), respectively. In order to get the CKS, let us define the posterior density \( p(x_k | D_k) = \int p(x_{k+1} | D_k)p(x_k | x_{k+1}, D_k)dx_{k+1} \) where \( D_k \) is the vector of measurements at time \( k \). Our nonlinear filtering algorithm can be obtained using five steps described as follows.

1) **Prediction.** Cholesky Factorization

\[ P_{k|k} = S_{k|k}S_{k|k}^T. \]  

Generate the cubature points \((i = 1, 2, \ldots, 2n)\)

\[ x_{k+1|i} = S_{k+1|i} \tilde{z}_i + \tilde{x}_{k+1|i}. \]  

Compute the propagated cubature point

\[ x_{k+1|i} = f(x_{k|i}). \]  

Compute the predicted state as follows:

\[ \tilde{x}_{k+1|i} = \frac{1}{2n} \sum_{i=1}^{2n} x_{k+1|i}. \]  

Compute the predicted state error covariance \( P_{k+1|i} \) and the cross-covariance \( P_{k+1|i} \) as follows:

\[ P_{k+1|i} = \frac{1}{2n} \sum_{i=1}^{2n} x_{k+1|i} x_{k+1|i}^T - \tilde{x}_{k+1|i} \tilde{x}_{k+1|i}^T + Q_k. \]  

\[ P_{k+1|i} = \frac{1}{2n} \sum_{i=1}^{2n} x_{k+1|i} x_{k+1|i}^T - \tilde{x}_{k+1|i} \tilde{x}_{k+1|i}^T. \]  

(2) **Measurement Update.** Equations (10) to (14) at time \( k + 1 \) are updated as follows:

\[ P_{k+1|k} = S_{k+1|k}S_{k+1|k}^T. \]  

Generate the cubature point

\[ x_{k+1|k} = S_{k+1|k} \tilde{z}_k + \tilde{x}_{k+1|k}. \]  

Compute the propagated cubature points \((i = 1, 2, \ldots, 2n)\)

\[ z_{k+1|i} = h(x_{k+1|i}). \]  

Computing the predicted measurement \( \tilde{z}_{k+1|i} \), the innovation covariance \( P_{zz,k+1|k} \) and the cross-covariance \( P_{xz,k+1|k} \) are as follows:

\[ \tilde{z}_{k+1|i} = \frac{1}{2n} \sum_{i=1}^{2n} z_{i,k+1|k} \]  

\[ P_{zz,k+1|k} = \frac{1}{2n} \sum_{i=1}^{2n} z_{i,k+1|k} z_{i,k+1|k}^T - \tilde{z}_{k+1|i} \tilde{z}_{k+1|i}^T + R_k. \]  

Using (19), compute the filter gain \( G_{k+1} \), the filtered state \( \tilde{x}_{k+1|k+1} \), and the filtered state error covariance \( P_{k+1|k+1} \) which are as follows:

\[ G_{k+1} = P_{xz,k+1|k} P_{zz,k+1|k}^{-1}, \]  

\[ \tilde{x}_{k+1|k+1} = \tilde{x}_{k+1|i} + G_{k+1} (z_{k+1|i} - \tilde{z}_{k+1|i}) \]  

\[ P_{k+1|k+1} = P_{k+1|i} - G_{k+1} P_{zz,k+1|k} G_{k+1}^T. \]  

Given the posterior density \( p(x_k | D_k) \) at time \( k \), compute the predictive density defined by

\[ p \left( x_k | D_k \right) = \mathcal{N} \left( \tilde{x}_{k+1|k}, P_{k+1|k} \right). \]  

(3) **Smoothing.** Let us assume that the joint posterior probability density functions (PDF) of \( x_{k+1|k+1} \) and \( \tilde{x}_k \) conditioned on \( D_k \) are Gaussian distributed. According to [5], the smoothed estimate and covariances can be obtained using the Gaussian approximation of the one-step smoothing PDF \( p(x_k | D_{k+1}) \) as follows:

\[ \tilde{x}_{k+1|k+1} = \tilde{x}_{k+1|i} + G_{k}^s (z_{k+1|i} - \tilde{z}_{k+1|i}) + G_{k}^s T, \]  

\[ P_{k+1|k+1} = P_{k+1|i} - G_{k}^s P_{zz,k+1|i} G_{k}^s T, \]  

where \( G_{k}^s = P_{xz,k+1|i} P_{zz,k+1|i}^{-1} \) is the smoothing gain matrix and \( P_{k+1|k+1} \) is the covariance matrix of the smoothed states. The superscript “s” represents smoother.

In order to make our filter adaptive to colored or variational noise, we use the heuristic as in [4], defined by a dynamic model and obtained using an approximation of the product of a Gaussian term and inverse Wishart (IW) term.

We consider the dynamic model that can be expressed as follows:

\[ \begin{align*}
    p \left( x_{k+1|k} | R_{k+1} \right) &= \mathcal{N} \left( x_{k+1|k} | \tilde{x}_{k+1|i}, P_{k+1|i} \right) \\
    &\times \text{IW} \left( R_{k+1} | \delta a_{k-1}, A_{k-1} \right),
\end{align*} \]
where $\rho$ is a parameter which controls the dynamics $0 < \rho \leq 1$ and $B$ is a matrix, $0 < |B| \leq 1$. $\rho = 1$ corresponds to stationary covariance and lower values allow for higher time-fluctuation. Assume that the state and the covariance estimate at time $k$ are obtained in advance using Gaussian distributions.

(4) Update. Set $\bar{x}_k = \bar{x}_{k|k+1}, P_{k|k+1}^{(0)} = R_{k+1}, a_k = 1 + a_k^*, \text{and } A_k^{(0)} = A_k^*$. We can compute the following parameters:

$$
\mu_k = \int h(x_k)N(x_k, \bar{x}_{k|k+1}, P_{k|k+1}) \, dx_k,
$$

$$
\alpha_k = \int h(x_k - \mu_k) h(x_k - \mu_k)^T \times N(x_k, \bar{x}_{k|k+1}, P_{k|k+1}) \, dx_k,
$$

$$
\gamma_k = \int (x_k - \bar{x}_{k|k+1}) (x_k - \bar{x}_{k|k+1})^T \times N(x_k, \bar{x}_{k|k+1}, P_{k|k+1}) \, dx_k,
$$

(24)

(5) Iteration. ($j = 1, 2, \ldots, N$)

$$
M_{k|j+1}^{(j)} = \alpha_k + (a_k - n - 1)^{-1} A_k^{(j)},
$$

$$
G_k^{(j+1)} = \gamma_k \left[ M_k^{(j+1)} \right]^{-1},
$$

$$
\bar{x}_k^{(j+1)} = \bar{x}_{k|k+1} + G_k^{(j+1)} (x_k - \mu_k),
$$

$$
P_{k|k+1}^{(j+1)} = P_{k|k+1} - G_k^{(j+1)} M_k^{(j+1)} \left[ G_k^{(j+1)} \right]^T.
$$

Set the final state estimate and its covariance as follows: $\bar{x}_k = \bar{x}_k^{(N)}, P_{k|k+1}^{(N)} = P_{k|k+1}^{(N)}, A_k = A_k^{(N)}$.

3.2. Extreme Learning Machine. As a learning algorithm for single-hidden layer feed-forward neural networks (SFLNs), ELM randomly selects weights and biases for hidden nodes and analytically determines the output weights by finding least square solution. The main concept behind ELM lies in the random initialization of the SLFN weights and biases. Under the condition that the transfer functions in the hidden layer are infinitely differentiable, the optimal output weights for a given training set can be determined analytically. The obtained output weights minimize the square training error. The basic structure of SFLNs is shown in Figure 2. ELM tends to reach not only the smallest training error but also the smallest norm of output weights [19, 20]. Given a training set $\{(x_j, y_j) \mid x_j \in \mathbb{R}^n, y_j \in \mathbb{R}^m, j = 1, \ldots, N\}$ which contains $N$ distinct examples, the SFLNs with $\bar{N}$ hidden nodes and activation function $g(x)$ can be formulated as

$$
\sum_{i=1}^{\bar{N}} \beta_i g(w_i \cdot x_j + b_i) = o_j, \quad j = 1, \ldots, N,
$$

where $w_i$ is the weight vector connecting the input nodes and the $i$th hidden node, $b_i$ is the bias of the $i$th hidden node, $\beta_i$ is the weight vector connecting the $i$th hidden node and the output nodes, $x_j$ is the $j$th training example, $o_j$ is the corresponding output of $x_j$, and $\cdot$ denotes the dot product of two vectors. An approximation of the SFLNs can be made. One has $\exists \omega_i, b_i, \beta_i$, such that

$$
\sum_{i=1}^{\bar{N}} \beta_i g(w_i \cdot x_j + b_i) = y_j, \quad j = 1, \ldots, N,
$$

(27) can be written in a simple form of matrix

$$
H^T \beta = Y,
$$

(28)

where

$$
H(w_j, b_j, x_j)
$$

$$
= \begin{bmatrix}
g(w_1 \cdot x_1 + b_1) & \cdots & g(w_{\bar{N}} \cdot x_1 + b_{\bar{N}}) \\
\vdots & \ddots & \vdots \\
g(w_1 \cdot x_N + b_1) & \cdots & g(w_{\bar{N}} \cdot x_N + b_{\bar{N}})
\end{bmatrix}_{N \times \bar{N}},
$$

(29)

$$
\beta_j = \begin{bmatrix}
\beta_1^T \\
\vdots \\
\beta_{\bar{N}}^T
\end{bmatrix},
$$

(30)

$$
Y_j = \begin{bmatrix}
y_1^T \\
\vdots \\
y_{N}^T
\end{bmatrix}.
$$

$H$ is often called the hidden layer output matrix.

3.3. Regularized ELM. In the traditional feed-forward network training methods, such as the back-propagation (BP) or MLP algorithms, all the parameters, including the weight vectors and bias values, need to be tuned iteratively, thus the
training speed may be very slow. As mentioned previously, what is more, they also suffer from other problems, such as local minima and overfitting. The ELM algorithm is summarized around three important steps which consist of generating random weights and bias for each hidden node, calculating the hidden output matrix of hidden layer $H$, and calculating the output weight $\beta$. It was shown that RELM can produce much more consistent results than the ELM. In order to get an optimized solution of the output weight, the ELM is often regularized using the residual error variable $\epsilon$ and the Lagrangian for (31) can be written as follows:

\[
\min \frac{1}{2}\|\beta\|^2 + \frac{1}{2} \omega \|D\epsilon\|^2
\]

\[\text{st} \sum_{i=1}^{L} \beta_i g \left( (w_i \cdot x_i + b_i) - y_j \right) - y_j = \epsilon_j, \quad j = 1, 2, \ldots, N, \tag{31}\]

where $D = \text{diag}(v_1, v_2, \ldots, v_N)$ is a matrix that can weight error variable $\epsilon_j = [\epsilon_1, \epsilon_2, \ldots, \epsilon_N]$ using weighting factors $v_i$ to obtain the best generalization performance by the optimal tradeoff between these two risks, and the Lagrangian for (31) can be written as follows:

\[
L(\beta, \epsilon, \alpha) = \frac{\omega}{2} \|D\epsilon\|^2 + \frac{1}{2} \|\beta\|^2 - \sum_{j=1}^{N} \alpha_j \sum_{i=1}^{L} \beta_i g \left( (w_i \cdot x_i + b_i) - y_j - \epsilon_j \right) \tag{32}\]

\[
= \frac{\omega}{2} \|D\epsilon\|^2 + \frac{1}{2} \|\beta\|^2 - \alpha (H\beta - Y - \epsilon) \tag{32}\]

The optimality conditions can be obtained; the partial difference equation group set equals zero as follows:

\[
\frac{\partial L}{\partial \beta} = 0 \rightarrow \beta^T = \alpha H, \tag{33}\]

\[
\frac{\partial L}{\partial \epsilon} = 0 \rightarrow \omega e^TD^3 + \alpha = 0, \tag{33}\]

\[
\frac{\partial L}{\partial \alpha} = 0 \rightarrow H\beta - Y - \epsilon = 0. \tag{33}\]

We can get a simplified expression of $\alpha$ by substituting (33) into (32), as follows:

\[
\alpha = -\omega (H\beta - Y)^T, \tag{34}\]

\[
\beta = \left( \frac{1}{\omega} + H^T D^3 H \right)^+ H^T D^3 Y. \tag{34}\]

RELM is summarized in Algorithm 1.

### 3.4. Ensemble RELM Based AdaBoost-R.T.

Many researchers have proposed various ensemble methods that integrate a set of ELMs into a combined network structures, and it was shown that they could perform better than when using individual ELM. In this paper, more recent boosting algorithms for regression problems called modified-AdaBoost regression and threshold (modified-AdaBoost.RT) [21, 22] are used to improve the performance of single RELM as an ensemble method. The original sample is divided into many samples and fed to every learner of the ensemble model. The weak learners are generated to evaluate which of them has the best prediction accuracy. The entire process of learning is running using a certain number of iterations $T$. At each step of the iteration, a distribution is assigned to every RELM weak learner that is called to provide it with the distribution, and the associated errors are calculated. A regression model is built within the process to minimize the errors of the output provided by the weak learners. In this work, we chose to apply the SW-RR which enables the using of the past inputs in order to reduce errors on the current outputs [24, 25]. Each learner calculates the absolute relative error for each training and updates the distribution. The structure of an ensemble RELM model is shown in Figure 3. Based on the description above, we derive RELM algorithm for GNSS/MEMS-INS integration as shown in Algorithm 2.

### 4. Methodology for GNSS/INS Integration and GPS Outages Prediction

#### 4.1. GNSS/INS Integration Scheme

In this section, the ERELM/VB-ACKS architecture for GNSS/INS integration is described. In order to achieve the integration of INS and GNSS data, we adopted a loosely coupled integration strategy [8]. GPS measurements are processed and fused with INS for its error computations [10]. The output of the mechanization is depicted in Figure 4; that is, the INS data is fused with the raw GNSS data using the VB-ACKS filter. The output of this fusion can provide accurately the position of a land vehicle in case of nonsignal blockages of GNSS. However, when the GNSS encounters signals blockages, the mechanized output that provides the INS data is trained by the ERELM method. The INS and GNSS data fused by VB-ACKS are selected as input for predicting the position of the vehicle using the ensemble ELM model.
4.2. Prediction Process. For simplicity of implementation, we denote by \( x_{in} = \delta P_{cat} \) the input sample and \( y_{out} \) the desired output of the prediction model. According to Algorithm 2, the input and output samples can be divided into \( m \) samples as follows: \((x_{in}^{1},y_{out}^{1})\cdots(x_{in}^{m},y_{out}^{m})\), where \((x_{in}^{m},y_{out}^{m})\) represents the \( m \)th input and output samples position, velocity, and attitude of the vehicle. The prediction starts with the training ERELM input samples (position, velocity, and attitude) provided by the past VB-ACKS outputs. These output samples are chosen as the inputs of ERELM while the current VB-ACKS output is considered as the desired output of ensemble ERELM model. When GPS outages occur, the trained ERELM is employed as the replacement of VB-ACKS to predict INS errors. The predicting process is dynamic because each ERELM output at current time will be provided for predicting the output of next moment. The position of a land vehicle is predicted by the ERELM as follows: Given a training set consisting of position error samples at time \( t \) denoted \( x_{in} \) (e.g., \( \delta P_{cat} \)) which represents the differences of position, velocity, and attitude between INS and GNSS. The training set is partitioned into \( m \) samples \((x_{in}^{1},y_{out}^{1})\cdots(x_{in}^{m},y_{out}^{m})\), where \((x_{in}^{m},y_{out}^{m})\) is used as input to the ensemble model as depicted in Figure 3. Each sample is given as input to the initialized weak learners and the number of iterations \( T \) is chosen. After initializing the weak learner and the error rate for each learner and setting the distribution for each weak learner, the RELMs are called to train the position, velocity, and attitude of samples. Within the same step, a regression model is built and the absolute relative error of the training is calculated. The error rate of each regression and the distributions are updated using equations \((***)\) and \((***\)) in Algorithm 2, respectively. The contribution of each training is evaluated and the final hypothesis is calculated. When GPS outages occur, the ensemble model chooses the learner that has the least error rate. The EREM model that combined the AdaBoost.RT is established to predict the position of the vehicle at time. The prediction process is shown in Figure 5. Based on the analytical study above, we derive ERELM algorithm for GNSS/MEMS-INS integration as described in Algorithm 3.
5. Experimental Results

5.1. Parameters of the GPS/INS Integrated Navigation Model. To evaluate the performance of the proposed scheme, the dataset collection was made using OBD-II coupled with a GPS u-blox LEA-6T and G-force data logger based on arduino MEGA2560, which serves as a MEMS-INS device. The inertial measurement unit is sampled at 50 Hz and consists of three accelerometers and three gyroscopes to get vehicle’s three axes angular velocity rate, three axes gyro rate, and three axes acceleration rate. The tests were conducted in land vehicle environments. The equipment used for data collection is shown in Figure 6. The trajectory of a vehicle as depicted in Figure 7 includes all the real-world scenarios, that is, U-turn, S-turn, or straight line and dynamics encountered in a typical LVN environment.

5.2. Results and Discussion. The assessment of the performance of our proposed algorithms for GNSS/MEMS-INS integration was made in two manners. Firstly, we tested the performance of the VB-ACKS and compare it with the cubature Rauch Tung Striebel Smoother (CRTS) [5] and the traditional KF. Figures 8, 9, and 10 show the estimated errors of the position, velocity, and attitude filtered by KF, CRTS, and VB-ACKS. As it can be seen in these figures, the VB-ACKS approach demonstrates its superiority compared with the KF and the CRTS, due to the fact that the VB-ACKS is able to deal with the nonlinear formulation and the variational noise, while the linear model used by the KF does not reflect the actual dynamic behavior when the vehicle is maneuvering. Although the CRTS outperforms the tradition KF, the same figures demonstrate that the VB-ACKS is much better. From Figures 8 and 9, we can see that the KF and CRTS solutions have large drift error even though the CRTS reduces the error of the KF. The VB-ACKS results are in general satisfactory in terms of errors compared to the CRTS and the conventional KF.

We trained the ERELM model for a fixed number of ELMs, which are 20. The system accuracy depends on the number of ELMs, but for the sake of computational cost the maximum number of ELMs used was 50; we choose to test our algorithm using 300-sample time. This is motivated by the fact that sampling the point will enable a good evaluation of the GPS accuracy and will benefit the low-cost MEMS-type device since it has a lower sampling rate compared to the expensive MEMS devices.

Figure 11 shows the East and North position errors estimated by the ERELM, LS-SVR [14], and the VB-ACKS. It can
be clearly observed that our proposed prediction method outperforms the LS-SVR and the VB-ACKS. Identically, Figures 12 and 13 show that our proposed ensemble RELM algorithm for predicting GPS outages outperforms the other prediction algorithm (i.e., LS-SVR) in estimating the North and East velocities errors. The proposed ERELM-VB-ACKS scheme improves as well the errors of roll, pitch, and yaw angles estimated by the conventional KF by 87%. We considered four different outages periods of 10, 30, 70, and 100 s each to evaluate the effectiveness of the proposed hybrid prediction algorithm, to mimic both short- and medium-term GPS outages conditions. As it is observed from Tables 1 and 2, the proposed ERELM model produces less positional error compared to VB-ACKS used in a standalone way and the LS-SVR. Our proposed GPS outage prediction methodology is capable of delivering stable output for the case where vehicle experiences sudden change in its trajectory as demonstrated through outage 2 in Figure 8. Table 2 shows the results obtained by our proposed algorithm during outage 1 and
outage 2 and is compared with LS-SVR algorithm which has shown good performance so far in GNSS/MEMS-INS integration. As it can be seen, our algorithm outperforms the VB-ACKS and LS-SVR. Table 2 confirms as well the performance of ERELM/VB-ACKS which is 30% better than the LS-SVR/VB-ACKS and LS-SVR/KF. From the same tables, we can see that the ERELM/VB-ACKS algorithm improves the position and velocity accuracy of GNSS/INS integration systems. In addition, corresponding RMSEs in these tables demonstrate that the ERELM prediction algorithm is able to provide remarkable position accuracy during the outages. LVN is often characterized by challenging trajectories that comprise curves. In order to test the performance of our algorithm, one outage was introduced during the “S” turn trajectory as depicted in Figure 8.

An illustration of our algorithm performance is shown in Figure 14 where two consecutive outages were introduced between time slots 200 and 250 on the East position during the training of ERELM and LS-SVR algorithms. This Figure shows that our algorithm outperforms the two other solutions, that is, VB-ACKS and LS-SVR. Compared with the other two solutions, ERELM prediction solution shows the outstanding performance during all the four GPS outages that ascribes to its ability to improve prediction accuracy through the combination of several predictors using AdaBoost.RT and the RELM algorithm.

Finally, the performance of ERELM during the four GPS outages introduced in the trajectory of the vehicle is demonstrated during the testing process. From Figure 14, we can clearly see that our algorithm shows a remarkable performance when the four GPS outages are introduced at the same time. As it can be observed, VB-ACKS gives the best results compared to the other two filters (CRTS/KF and conventional KF) solution used. Ten seconds GPS outages can be observed in Figure 14 between time slots 50 and 100. One
can notice that the novel filter VB-ACKS is able to reduce the error compared to the CRTS and conventional KF. The error of KF reduced from the highest 12.04 m to 0.08 m. This is due to the ability of VB-ACKS in handling variational noise. From Figure 14 and Table 1, ERELM/VB-ACKS shows good results as well compared to the LS-SVR. Other GPS outages were introduced between time slots 200 and 250 as depicted in Figure 14. We can see that ERELM still outperforms the other algorithms. The error of KF reduced from 48.12 m to 0.57 m by the ERELM/VB-ACKS. Because ERELM uses the past output of VB-ACKS results, the output of our algorithm gives less positional error in general compared to the other LS-SVR based prediction algorithm.

6. Conclusion

This paper has proposed an integration scheme for low-cost MEMS-INS sensor using a nonlinear AI model based on ensemble learning model. This model is derived by employing the well-known AdaBoost ensemble ELM. The ELM was regularized in advance. The variational Bayesian Kalman smoother (VB-ACKS) is used in case of an absence of GPS outages to overcome the flaws of the KF in the GNSS/INS integration navigation. VB-ACKS is able to provide a better position error in the presence of dynamic variation of the vehicle and the INS sensor error variation. The results indicate that the proposed smoothing algorithm outperforms the conventional KF. In case of the presence of GPS outages, the ERELM is used as an error compensation solution. The prediction of the ERELM provides a better accuracy than the prediction of the AI algorithm LS-SVR.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.
Acknowledgments

This work was partly supported by National Natural Science Foundations of China (no. 61301148 and no. 61272061), the Fundamental Research Funds for the Central Universities of China (nos. 531107040263, 531107040276), the Research Funds for the Doctoral Program of Higher Education of China (no. 20130161100002 and no. 20120161120019), Hunan Natural Science Foundation of China (no. 14JJ7025).

References


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