Motion-Constrained GNSS/INS Integrated Navigation Method Based on BP Neural Network

Ying Xu, Kun Wang, Changhui Jiang, Zeyu Li, Cheng Yang, Dun Liu, Haiping Zhang

remote sensing

MDPI

Article

Motion-Constrained GNSS/INS Integrated Navigation Method Based on BP Neural Network

Ying Xu [1,2], Kun Wang [1], Changhui Jiang [3], Zeyu Li [1,*], Cheng Yang [4], Dun Liu [5] and Haiping Zhang [6]

[1] College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao 266590, China
[2] Qingdao Key Laboratory of Beidou Navigation and Intelligent Spatial Information Technology Application, Qingdao 266590, China
[3] GEOLOC Laboratory, Université Gustave Eiffel, 77454 Paris, France
[4] Department of Land Surveying and Geoinformatics, China University of Geosciences, Beijing 100083, China
[5] Department of GNSS High Precision Positioning and Ionosphere Correction, No. 22nd Research Institute, CETC, Qingdao 266108, China
[6] Shandong Provincial Institute of Land Surveying and Mapping, Jinan 250013, China
* Correspondence: lizeyu@sdust.edu.cn

Abstract: The global navigation satellite system (GNSS) and inertial navigation system (INS) integrated navigation system have been widely used in Intelligent Transportation Systems (ITSs). However, the positioning error of integrated navigation systems is rapidly divergent when GNSS outages occur. Motion constraint and back propagation (BP) neural networks can provide additional knowledge to solve this issue. However, the predictions of a neural network have outliers and motion constraint is difficult to adapt according to the motion states of vehicles and boats. Therefore, this paper fused a BP neural network with motion constraints, and proposed a motion-constrained GNSS/INS integrated navigation method based on a BP neural network (MC-BP method). The pseudo-measurement of the GNSS was predicted using a fitting model trained by the BP neural network. At the same time, the prediction outliers were detected and corrected using motion constraint. To assess the performance of the proposed method, simulated and real data experiments were conducted with a vehicle on land and a boat offshore. A classical GNSS/INS integration algorithm, a motion-constrained GNSS/INS algorithm, and the proposed method were compared through data processing. Compared with the classical GNSS/INS integration algorithm and the motion-constrained GNSS/INS algorithm, the positioning accuracies of the proposed method were improved by 90% and 64%, respectively, in the vehicle land experiment. Similar performances were found in the offshore boat experiment. Using the proposed MC-BP method, improved meter-level-positioning results can be achieved with the GNSS/INS integration algorithm when GNSS outages occur.

Keywords: motion constraint; BP neural network; integrated navigation system; INS; GNSS outages

Citation: Xu, Y.; Wang, K.; Jiang, C.; Li, Z.; Yang, C.; Liu, D.; Zhang, H. Motion-Constrained GNSS/INS Integrated Navigation Method Based on BP Neural Network. *Remote Sens.* **2023**, *15*, 154. https://doi.org/10.3390/rs15010154

Academic Editor: Yunbin Yuan

Received: 7 November 2022
Revised: 22 December 2022
Accepted: 23 December 2022
Published: 27 December 2022

1. Introduction

Intelligent transportation systems (ITSs) including unmanned driving vehicles, unmanned boats, intelligent logistics, and shared travel put forward high requirements for the accuracy and reliability of vehicle and boat navigation [1]. In order to obtain more precise navigation and positioning results, two (or more) types of integrated navigation and positioning technologies are generally adopted [2]. The fusion of the global navigation satellite system (GNSS) and an inertial navigation system (INS) is a common method [3]. An INS has the advantages of full autonomy, a fast sampling rate, and high accuracy within integration times. Using an INS, 3D positioning and velocity data can be achieved in all types of weather [4]. However, INS positioning errors cumulatively increase over time [5]. In most cases, users can obtain high-precision 3D positioning and velocity data using a GNSS [6]. However, GNSS signals are vulnerable to interference and their sampling rates are low [6]. GNSSs and INSs have complementary functions. Integrated GNSS and INS

navigation systems have been widely used in the fields of air, land, and sea navigation [7]. However, GNSS signals are often blocked by trees, buildings, and viaducts in urban settings and are blocked by large oil-drilling platforms and cross-sea bridges in marine settings, resulting in less visibility for satellites and serious multipath issues. Reliable position information cannot be obtained, the measurement updates anomalies, and then the positioning solutions of integrated navigation systems rapidly become divergent [8]. Therefore, how to keep the GNSS/INS integrated navigation system working normally when GNSS outages occur is an urgent problem [9–11]. Three methods are usually used to overcome the above problems.

One category is the use of external auxiliary sensors including barometers, odometers (ODs), and vision cameras [12]. Godha [13] first obtained a vehicle's forward velocity by incorporating an OD into an integrated navigation system. Yan [14] added ODs to the GNSS/INS integrated navigation system and decimeter-level positioning was achieved. Li and Zhai [5,15] combined vision cameras with a GNSS/INS integrated navigation system, which effectively improved the positioning accuracy in complex urban environments. These methods can effectively improve the positioning accuracy during GNSS outages. However, the methods of using extra sensors increase the cost, and data processing is more complicated.

Another category is the use of motion constraints to overcome divergence errors in positioning. Common motion constraints include elevation constraints, velocity constraints, and attitude constraints. Motion constraint algorithms are derived from a vehicle's inherent motion characteristics. During GNSS outages, the constraint values of the motion constraint algorithm can be introduced to a filter as pseudo measurements to replace missing GNSS measurements [14]. Godha [13] proposed an elevation constraint based on the small change of vehicle elevation over a short time, which effectively reduced the elevation error, but the improvement in the horizontal positioning accuracy was not obvious. Dissanayake [16] proposed the idea of nonholonomic constraint (NHC), and pointed out that it was better to set the lateral velocity and vertical velocity of the vehicle to zero when no jump or sideslip phenomenon occurred. In view of an urban environment, Klein, et al. [17] proposed an attitude angular velocity constraint, and the initial value of the horizontal attitude angle was set to 0. Niu et al. [18] proposed a heading angular velocity constraint that could improve the positioning accuracy of low dynamic vehicles. The above motion constraint information is established by a vehicle in an ideal motion state, which generally yields a constant constraint. However, the actual motion of a vehicle or boat is very complex, which reduces the effectiveness of a constant motion constraint algorithm, especially for the velocity constraint [19]. In order to an achieve an adaptive velocity constraint according to a vehicle's motion, Liu et al. [19] analyzed the relationship between NHC and vehicle's state of motion, and connected vehicle lateral velocity with both forward velocity and heading angular velocity. However, in subsequent vehicle lateral velocity constraint modeling, only a relatively simple linear multiplication adaptive adjustment formula was used.

Using an artificial neural network (ANN) algorithm is an applicable method to improve the positioning accuracy during GNSS outages. An ANN algorithm is suitable for predicting good results for nonlinear systems. Many researchers have built a variety of neural networks to aid INSs when GNSS outages occur [20–23]. The main idea of an ANN-aided GNSS/INS integrated system algorithm is to mimic the mathematical relationship between the navigation information and the vehicle's dynamical data to maintain high navigation accuracy when GNSS outages occur [21]. Rashad et al. [22] first used a basis function (RBF) neural network to predict the positioning error between an INS and a GNSS. EI-Sheimy et al. [23] used a multi-layer feed-forward network to fit the relationships between time, velocity, yaw and position error, and velocity. Fang et al. [21] used LSTM to predict GNSS position increments during GNSS outages, and also pointed out that an ANN algorithm had the disadvantage of the predicted value inevitably containing outliers. In addition to predicting error correction values (INS errors) or pseudo-measurement (GNSS velocity and position result or increments), an ANN can also be used to solve the adaptive

problem of motion constraint parameters. Zhang et al. [24] used an RBF to predict vehicle lateral and vertical velocity directly. Brossard et al. [25] used the convolutional neural network (CNN) to achieve the optimal estimation of the observed variance in an NHC when processing moving vehicles.

In short, a motion constraint algorithm is very stable when GNSS outages occur. However, the constraint value cannot be adapted according to a vehicle's motion state. The GNSS pseudo-measurement predictions gained by using a neural network algorithm inevitably have outliers. In view of the above problems, this paper proposes a motion-constrained GNSS/INS integrated navigation method based on a BP neural network (MC-BP). In this method, a BP neural network is continuously trained to obtain a fitting model between GNSS positioning increments and integrated navigation system positioning increments when GNSS signals are available. During GNSS outages, the trained fitting model can predict the GNSS positioning increments. At the same time, the outliers of the predicted values are detected and corrected using motion constraint. Then, the pseudo measurement values of the GNSS are obtained. These pseudo measurements can allow the extended Kalman filter (EKF) of the integrated navigation system to work normally.

The remaining sections of this paper are organized as follows: Section 2 introduces the principles of the BP neural network algorithm and motion constraint algorithm, and proposes the MC-BP method. Section 3 assesses the performance of the algorithm proposed in this paper through two sets of experiments: one on land and the other on sea. The conclusions are drawn in Section 4.

2. Methodology

2.1. Classical GNSS/INS Loosely Coupled Integrated Procedure

A classical GNSS/INS loosely coupled integrated procedure can be expressed as follows [8]:

$$x_k = f(x_{k-1}) + \eta_k \tag{1}$$

$$z_k = h(x_k) + v_k \tag{2}$$

where x is the error state vector, f is the system nonlinear state function, η is a noise distribution matrix whose variance matrix is Q_k, Z is the observation vector, h is a system nonlinear state function of the observation equation, and v is the noise distribution matrix whose variance matrix is R_k. The state vector x of the loosely coupled model in the navigation coordinate system is a 15-dimensional vector:

$$x = [\delta\varphi_E \ \delta\varphi_N \ \delta\varphi_U \ \delta v_E \ \delta v_N \ \delta v_U \ \delta p_E \ \delta p_N \ \delta p_U \ \varepsilon_x \ \varepsilon_y \ \varepsilon_z \ \nabla_x \ \nabla_y \ \nabla_z]T \tag{3}$$

where $\delta\varphi_E \ \delta\varphi_N \ \delta\varphi_U$ are the attitude errors, $\delta v_E \ \delta v_N \ \delta v_U$ are the velocity errors, $\delta p_E \delta p_N \ \delta p_U$ are the position errors, ε is the three-axis gyro bias, and ∇ is the three-axis accelerometer bias [8]. The Jacobian matrix is obtained by expanding $f(x_{k-1})$ and $h(x_k)$ at x_{k-1} and x_k, respectively, using the Taylor expansion [21]:

$$F_{k-1} = \frac{\partial f}{\partial x}|_{(x=x_{k-1})} \tag{4}$$

$$H_k = \frac{\partial h}{\partial x}|_{(x=x_k)} \tag{5}$$

where $k-1$ and k represent the state labels, respectively, and F_{k-1} and H_k are the Jacobian matrix. If the 15-dimensional state vector is rewritten in discrete time, the process of the EKF is given as follows [25]:

$$x_{k,k-1} = f(x_{k-1}) \tag{6}$$

$$P_{k,k-1} = F_{k-1}P_{k-1}F_{k-1}^T + Q_{k-1} \tag{7}$$

$$K_k = P_{k,k-1}H_k^T(H_kP_{k,k-1}H_k^T + R_k)^{-1} \tag{8}$$

$$x_k = x_{k,k-1} + K_k(z_k - h(x_{k,k-1}))^{-1} \quad (9)$$

$$P_k = (I - K_k H_k)P_{k,k-1} \quad (10)$$

where P is the covariance matrix of the state, K_k is the gain matrix, and I is the unit matrix.

2.2. Motion-Constrained GNSS/INS Integrated Navigation Method Based on BP Neural Network

2.2.1. Pseudo Measurement Prediction Model Based on BP Neural Network

In practical applications, GNSS measurements can be lost or abnormal due to the occurrence of GNSS outages. As a consequence, Equation (9) cannot proceed normally. ANN-aided navigation systems have shown impressive performances in the pseudo measurement prediction [20–23].

Brossard et al. [25] pointed out that training a neural network is quite difficult and slow. Setting an adapter with a recurrent architecture makes the training even more difficult. Simple CNN-like shallow neural network architecture can effectively overcome the above problems. However, a CNN is generally used for deep learning. Both BP and RBF belong to the category of feed-forward neural networks and have their own advantages. The generalization ability of RBF is better than that of the BP network in many aspects. However, the structure of a BP network is simpler than that of an RBF network when solving problems with the same accuracy requirements. Therefore, a BP neural network was selected for use in this paper. Neural networks can avoid overfitting by using a relatively small number of parameters. The BP neural network is one of the most widely used neural networks, which is usually called artificial neural network that is based on an error back-propagation algorithm. The structure of a BP neural network is as shown in Figure 1. There are three layers in a classical BP neural network structure including an input layer, a hidden layer, and an output layer [26].

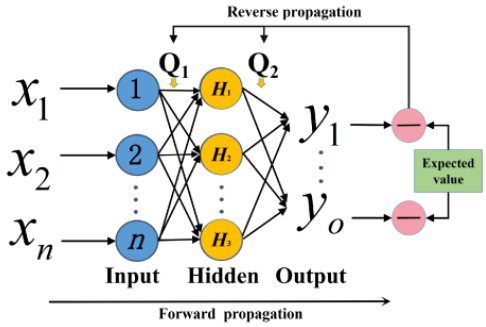


Figure 1. Topology structure and error propagation process of a three-layer BP neural network.

The input layer vector is $x = (x_1, x_2, \ldots x_n)^T$, the hidden layer vector is $H = (H_1, H_2, \ldots H_m)^T$, and the output layer vector is $y = (y_1, y_2, \ldots y_i)^T$. BP neural networks include signal-forward propagation and error-backward propagation. When calculating the error output, it is carried out in the direction from input to output, and the adjustment of weights is carried out in the direction from output to input [26].

In an ANN-aided navigation systems algorithm, the selection of the input layer and output layer of a neural network is very important. Research shows that $O_{INS} - \Delta P_{GNSS}$ mapping models have good performances [21]. In this model, ΔP_{GNSS} is the increment of the GNSS position, and O_{INS} contains the specific force, angular velocity, velocity, and heading angle. The parameters in O_{INS} are also the main factors that affect the values of the increments of INS position ΔP_{INS}. In addition, the position increment accuracy of a GNSS/INS integration algorithm is better than that of INS individually when GNSS signals are available. To reduce the number of input parameters and increase the training speed, only the increments of the integrated navigation system position ΔP_{INT} were used as input, and ΔP_{GNSS} was the output of the pseudo-measurement prediction model based on the BP neural network. The factors of the input layer and output layer are expressed as follows:

$$x_j = \left[\Delta P_{INT}^{n-j}, \ldots \Delta P_{INT}^{n-1}, \Delta P_{INT}^{n}\right] \tag{11}$$

$$y_j = \left[\Delta P_{GNSS}^{n-j}, \ldots \Delta P_{GNSS}^{n-1}, \Delta P_{GNSS}^{n}\right] \tag{12}$$

where x_j is the position increment result of the GNSS/INS integrated navigation system in a training window period, y_j is the position increment result of the GNSS integrated navigation system in a training window period, and j is the epoch of a training window period. In this paper, the position increments of the integrated navigation system and the position increments of GNSS in three directions were trained separately, by subtracting the actual position increment output by the neural network directly and the actual GNSS position increment that was output by the GNSS:

$$e_j = \Delta P_{BP}^{j} - \Delta P_{GNSS}^{j} \tag{13}$$

where, if the error e_j was greater than the expected value, back-propagation of the error was performed, and the weights were adjusted until the e_j was within the ideal range. In addition, the hidden layer of the neural network is set to 3, the maximum iteration times for training were 300, the initial learning rate was 0.01, and the training method was the gradient descent method.

2.2.2. Vehicle Motion Constraints Algorithms

The pseudo-measurement predicted by the BP network-trained model could have outliers. To detect and correct the outliers, motion constraints including elevation constraint, velocity constraint, and attitude constraint were adopted in this paper.

- *Elevation Constraint Algorithm*

When a vehicle moves on a relatively flat area, its elevation does not change greatly in a short time [17]. The elevation of a vehicle in a navigation coordinate system is shown in Figure 2.

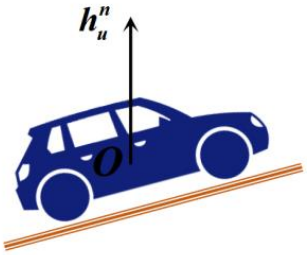


Figure 2. Vehicle elevation in navigation coordinate system.

In the construction of constraint equations for elevation, a constant can be used. However, the observation value of elevation is not constant in an actual observation. Therefore, this paper used the elevation of the last sampling epoch to constrain the elevation of the current epoch. The elevation constraint equation is as follows [17]:

$$h_k^n = h_{k-1}^n + \sigma_h, E(\sigma_h) = 0 \tag{14}$$

where h is the elevation,σ_h is the elevation measurement error of the vehicle in the navigation coordinate system, and E is the expectation value.

- *Velocity Constraint Algorithm*

There are two nonholonomic constraints when a wheeled vehicle moves on the ground. If the vehicle does not slide or jump off the ground, the vertical velocity and lateral velocity of the vehicle should be 0 [16] (Equation (15)). Vertical velocity and lateral velocity in the body coordinate system are shown in Figure 3.

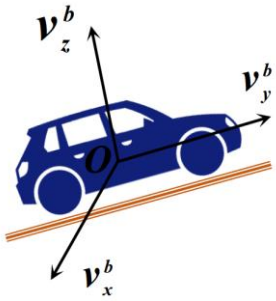


Figure 3. Lateral and vertical velocity in body coordinate system.

$$v_x^b = 0, v_z^b = 0 \tag{15}$$

Although many models can be used to determine whether a vehicle slides or jumps when it is moving, the vehicle state of motion and the tire and ground characteristics should be known prior to using these models. As a consequence, the modeling is very complex [16]. To overcome this issue, this paper took the lateral velocity noise and vertical velocity noise of a vehicle GNSS sensor in the body coordinate system as constraints [17]. The constraint equations are as follows:

$$v_x^b = 0 + \sigma_x^b, E\left(\sigma_x^b\right) = 0 \tag{16}$$

$$v_z^b = 0 + \sigma_z^b, E\left(\sigma_z^b\right) = 0 \tag{17}$$

where v_x^b, v_z^b are the lateral velocity and vertical velocity of a vehicle in the body coordinate system, respectively, and σ_x^b, σ_z^b are the lateral velocity error and vertical velocity errors of the vehicle, respectively. Since the velocity parameters were directly measurable, the modified velocity was used to recalculate the position of the vehicle [27].

- *Attitude Constraint Algorithm*

When a vehicle moves on a relatively flat area, its pitch angle and roll angle change little, and the variation in horizontal attitude angle (pitch angle and roll angle) is about 0 [17]. This paper used the horizontal attitude of the last epoch to constrain the horizontal attitude of the current epoch. The attitude constraint equations are as follows [18]:

$$\theta_{x(k)}^n = \theta_{x(k-1)}^n + \omega_x + \eta_{roll}, E(\eta_{roll}) = 0 \tag{18}$$

$$\theta_{y(k)}^n = \theta_{y(k-1)}^n + \omega_y + \eta_{pitch}, E\left(\eta_{pitch}\right) = 0 \tag{19}$$

where θ_x^n, θ_y^n are the vehicle roll angle and pitch angles, respectively, ω_x, ω_y are the zero biases of the gyroscope, and $\eta_{roll}, \eta_{pitch}$ are the random walk noises of the gyroscope. Since the attitude parameters were directly measurable, the modified attitude was used to correct the position of the vehicle [27].

It should be mentioned that all of the motion constraint settings above were based on vehicle inherent motion characteristics. The available motion constraints for boats are tested in the simulated and real data experiments in Section 3.

- *Outlier Detection and Correction based on Motion Constraints*

As mentioned above, implementing motion constraints in both the system state and measurement models can enhance the EKF filter performance without the use of other sensors. Motion constraints can also be used to obtain vehicle position increments through single filtered calculation. Figure 4 shows the relationship between a space's virtual observation (green ellipse) and real observation (red triangle), as well as measurement updates from motion constraints. As shown in Figure 4a, if the virtual observation space is too small, the motion constraint error ellipse cannot contain the real motion constraint

observation value [24]. In order to overcome this issue, enlarging the observation variance of the virtual observation is necessary, as shown in Figure 4b.

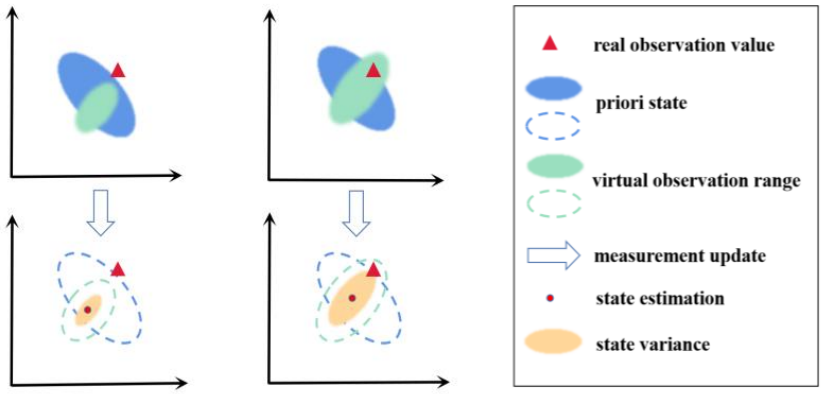


Figure 4. Relationship between a space's virtual observation (green ellipse) and real observation (red triangle), as well as measurement updates from motion constraints; (**a**) little constraint value; (**b**) large constraint value.

A constraint value that is too large or too little will affect the performance of the algorithm proposed in this paper. After experimental verification, the tripling motion constraint standard deviation is regarded as a reasonable value. As a consequence, triple motion constraint standard deviation was used to calculate the position increments in this paper. These position increments were used to detect and correct the outliers of the predictions of the BP neural network-trained model. The relative equations are as follows:

$$\Delta P_{pre} = \Delta P_{pre-GNSS},\ \Delta P_{pre-GNSS} \leq \Delta P_{mc(3\tau)} \tag{20}$$

$$\Delta P_{pre} = \Delta P_{mc(\tau)},\ \Delta P_{pre-GNSS} > \Delta P_{mc(3\tau)} \tag{21}$$

where τ is the standard deviation of the motion constraints including the lateral velocity constraint, the vertical velocity constraint, the elevation constraint, and the horizontal attitude constraint. $\Delta P_{pre-GNSS}$ is smaller than $\Delta P_{mc(3\tau)}$ and the prediction value $\Delta P_{pre-GNSS}$ is reasonable [28], which can be brought into the EKF for measurement updating as the ΔP_{pre}. Otherwise, the prediction value $\Delta P_{pre-GNSS}$ is recognized as an outlier, and the position is directly calculated by the $\Delta P_{mc(\tau)}$.

2.2.3. Steps of the Proposed MC-BP Method

The algorithm flowchart proposed in this paper is shown in Figure 5, where, λ, l, h are the points of latitude, longitude, and height output by the integrated navigation system, respectively. The system principle is shown in Figure 5.

By combining the advantages of a BP neural network and motion constraints, a new GNSS/INS integrated navigation method (MC-BP method) was proposed. The flowchart of the algorithm proposed in this paper is shown in Figure 5. There were two system working states in this method. The green dotted box shows the data processing flowchart when the GNSS signals are available, while the red dotted box shows the data processing flowchart when GNSS outages occur.

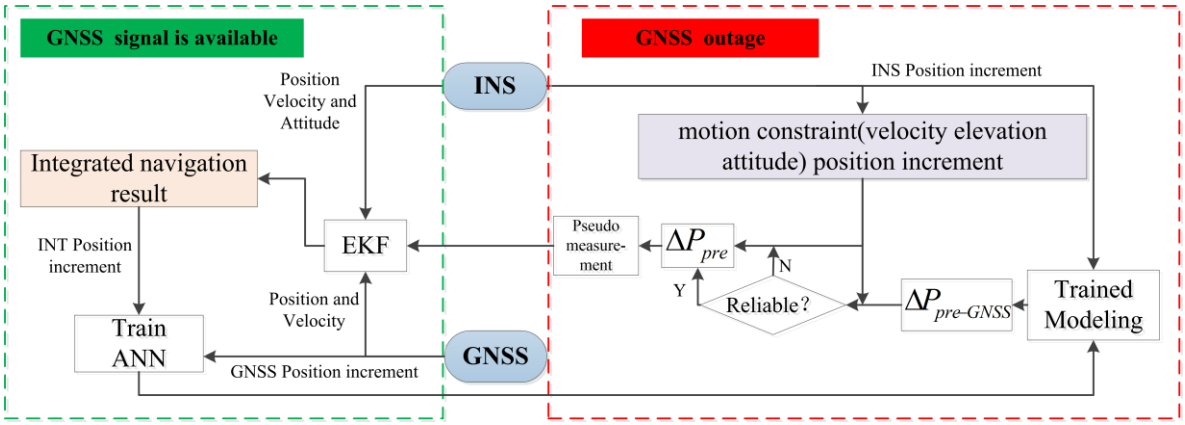


Figure 5. Flowchart of the proposed method.

(1) GNSS signals are available:

- The position increments $\Delta P_{INT} = (\Delta P^1_{INT}, \Delta P^2_{INT}, \ldots \Delta P^j_{INT})^T$ gained from the integrated navigation system are used as the input for the BP neural network;
- The position increments $\Delta P_{GNSS} = (\Delta P^1_{GNSS}, \Delta P^2_{GNSS}, \ldots \Delta P^j_{GNSS})^T$ from the GNSS are used as output for the BP neural network;
- The training process is continued between the input layer and the output layer of the BP neural network, and a fitting model is obtained. At the same time, the position increments $\Delta P_{mc(3\tau)}$ of the vehicle for the next epoch are obtained by using the motion constraint algorithm.

(2) GNSS outages:

- Assuming a GNSS outage occurs at the epoch of $m+1$, and the GNSS cannot be obtained. At this time, the position increments $\Delta P^{m+1}_{INS} = (\Delta\lambda^{m+1}_{INS}, \Delta\phi^{m+1}_{INS}, \Delta h^{m+1}_{INS})^T$ output by the INS are brought into the trained model to obtain the predicted values of the position increments $\Delta P^{m+1}_{pre-GNSS} = (\Delta\lambda^{m+1}_{pre-GNSS}, \Delta\phi^{m+1}_{pre-GNSS}, \Delta h^{m+1}_{pre-GNSS})^T$ for the GNSS;
- The position outliers are detected and corrected using Equations (20) and (21), and the ΔP_{pre} is obtained.
- The pseudo-measurement is introduced, which is obtained by ΔP_{pre} using the EKF and the final GNSS/INS integration navigation results are obtained.

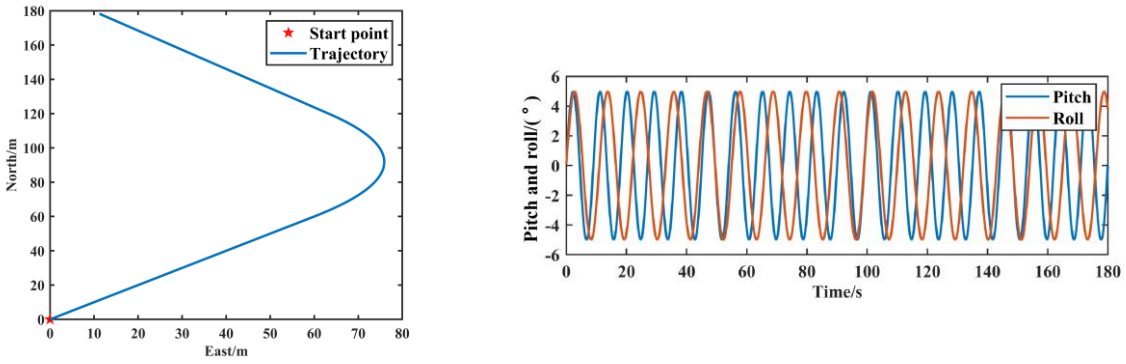
3. Experiments and Analysis

Since boats are affected by various kinds of environmental factors in the ocean such as waves, wind, and biology, they present axial movements along the x, y, and z directions. In actual sea conditions, waves also affect the lateral velocity of boats. In addition, boats also have a sliding phenomenon when turning. Niu et al. [29] proposed that vehicles in the form of bicycles swing during cycling, so an NHC would no longer be applicable in this situation. Theoretically, an NHC is no longer applicable to boats moving in complex offshore conditions [30]. In order to assess the availability of motion constraints for boats and to choose applicable constraints for boats, this paper first designed a set of simulation experiments. To evaluate the performance of the proposed MC-BP method, real data experiments were conducted on land and offshore with vehicles and boats, respectively.

3.1. Boat Motion Constraint Simulation Experiment

In this simulation experiment, the boat first moved with a constant velocity of 1 m/s for 60 s, made a uniformly accelerated turn with an acceleration of 0.1 m/s^2 for 60 s, and finally performed a uniform linear motion for 60 s. The starting point of the simulation experiments was 36.0264° N, 120.3045° E. The motion trajectory of the simulation experiment for the

boat is shown in Figure 6a. The pitch angles and roll angles are shown in Figure 6b. The navigation information in Figure 6a was used as a reference result for the simulation experiments. The simulated GNSS measurement data were generated by adding a pseudo-random error to the simulated reference value of the trajectory. The average value of the pseudo-random error was approximately ±0.05 m, which was the positioning accuracy of the GNSS sensor used in the real test.



(**a**) (**b**)

Figure 6. (**a**) Motion trajectory of the simulation experiment boat; (**b**) pitch and roll.

In order to reduce the abnormal filtering results of the positioning caused by noise constraints that are too loose or tight, three constraint schemes were designed in this simulation experiment. The velocity constraint, attitude constraint, and elevation constraint were adopted using Equations (14) and (17)–(19). Considering that the sea level changes little in a short time, the elevation constraints of the three schemes were uniform. The constraint values of the constraints for scheme 1 were the same as the positioning accuracy of the sensor used in the simulation experiment [13]. The inertial navigation equipment *inertial+* was produced by OXTS. The constraint values of schemes 2 and 3 were larger than that of scheme 1. The velocity and attitude constraints of the three schemes are shown in Table 1. To simulate a GNSS outage occurrence, only the inertial navigation solution was used. The positioning errors using pure INS and INS with the three motion constraint schemes of the simulated experiments are shown in Figure 7.

Table 1. Motion constraint schemes.

Constraints	Scheme 1	Scheme 2	Scheme 3
vertical velocity	0.01 m/s	0.05 m/s	0.1 m/s
pitch	0.2°	0.5°	1°
roll	0.2°	0.5°	1°
heave	0.1 m	0.1 m	

It can be seen from Figure 7 that the errors diverged quickly when the pure INS mode was used, and the maximum positioning errors in the north, east, and up directions were 402 m, 471 m, and 15.9 m, respectively. When the constraint schemes were introduced, the positioning error reduced significantly. Compared with schemes 1 and 3, the performance of scheme 2 was better. Using scheme 2, the maximum positioning errors in the north, east, and up directions were 23.2 m, 19.5 m, and 1.3 m, respectively. In short, the motion constraints were able to be used for boats, and the positioning performance was affected by motion constraints that were too loose or too tight. The usage of an appropriate constraint value was necessary, and motion constraint scheme 2 was used in the real offshore experiments described in the following section.

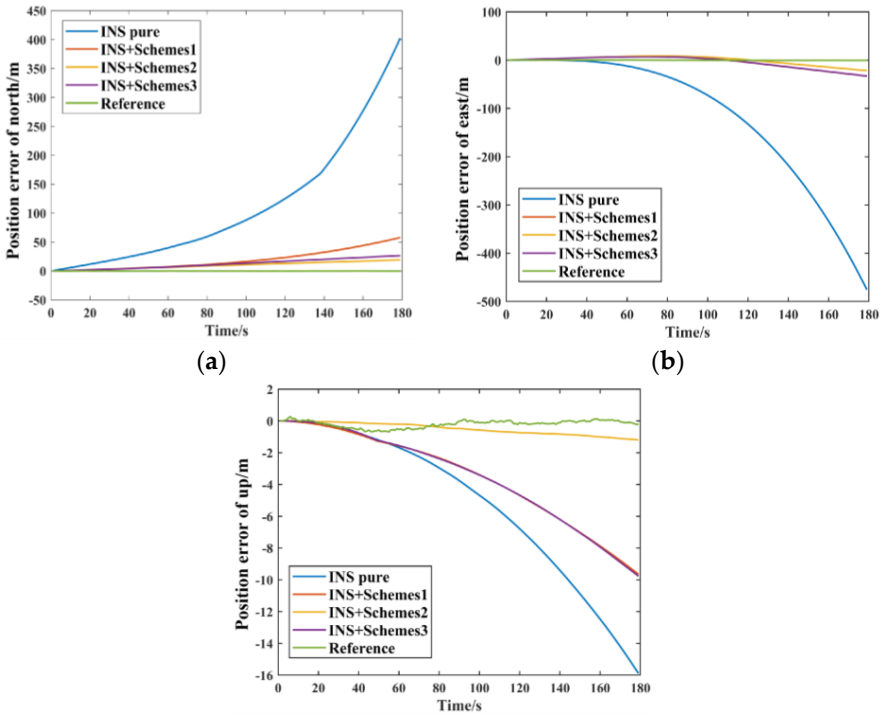


Figure 7. Positioning errors using different motion constraint schemes: (**a**) north direction; (**b**) east direction; (**c**) up direction.

3.2. Real Data Experiments

3.2.1. Experimental Data and Data Processing Strategy

In this paper, the real data experiments were divided into two experimental scenes: land and offshore. The land experiment was conducted in the new west coast area of Qingdao on 28 June 2021, from 07:46:00 to 08:00:00 (GPST). The offshore experiment was conducted in Houchawan, Qingdao, on 2 August 2021, from 08:20:00 to 08:35:00 (GPST). In order to assess the performance of the motion-constrained GNSS/INS integrated navigation method during GNSS outages, the GNSS signal was interrupted artificially. The real data for both land and offshore were interrupted five times, and the interruption last 60 s each time. As shown in Figure 8, the interruption sequence and corresponding trajectory are labeled as ①~⑤, respectively. The motion states of the vehicle and boat during the interruptions included acceleration, deceleration, turning, and traveling straight.

Figure 8. Trajectories of the experiments: (**a**) on land; (**b**) offshore.

The same experimental equipment was used for the land and offshore experiments. The inertial navigation system used was *inertial+* MEMS inertial navigation equipment produced by OXTS. The sampling frequency of the INS was 100 Hz and the sampling

frequency of the GNSS was 1 Hz. The sensors' main performance specifications are shown in Table 2, and the carriers are shown in Figure 9. Figure 9a depicts the vehicle used in the land experiment and Figure 9b shows the boat used in the sea experiment. The GNSS positioning method was real-time kinematic (RTK).

Table 2. Navigation and positioning sensor parameters.

Sensor Type	Error Term	Value
gyroscope	zero bias	$2^\circ/\sqrt{h}$
	random walk	$0.2^\circ/\sqrt{h}$
accelerometer	zero bias	5 mm·s^{-2}
	random walk	20 mg
GNSS	horizontal positioning accuracy	0.02 m(RTK)
	vertical positioning accuracy	0.05 m(RTK)
	velocity measurement accuracy	0.05 m/s(RTK)
lever arm error	x	0.014 m
	y	0.342 m
	z	0.568 m

(**a**) (**b**)

Figure 9. Experimental instrument and carriers: (**a**) vehicle; (**b**) boat.

In order to assess the performance of the method proposed in this paper, a classical GNSS/INS integration algorithm, a motion-constrained GNSS/INS algorithm, and the proposed MC-BP method were used in the data processing. The tight combined positioning results post-processed by the GNSS RTK/INS were used as the reference. The motion constraint schemes of the experiments on land and offshore are shown in Table 3. The values of the constraints for the land experiment were the same as the positioning accuracy of the sensor used in this paper. The offshore experiment used constraint scheme 2 of the simulation experiment. The pitch and roll angle constraints of the land experiment were smaller than those of the offshore experiment.

Table 3. Motion constraint schemes.

Constraint Content	Land Constraint Scheme	Offshore Constraint Scheme
vertical velocity	0.05 m/s	0.05 m/s
lateral velocity	0.05 m/s	\
pitch	0.2°	0.5°
roll	0.2°	0.5°
heave	0.1 m	0.1 m

3.2.2. Analysis Results of the Vehicle Land Experiment

The positioning errors of the classical GNSS/INS integration algorithm without constraints are shown in Figure 10. Figure 11 shows the positioning errors of the motion-

constrained GNSS/INS algorithm and the proposed MC-BP method. From Figure 10, we can find that the positioning errors for the classical GNSS/INS integration algorithm in all three directions diverge rapidly when GNSS outages occur. The maximum positioning errors in the north, east, and up directions were 296.21 m, 336.70 m, and 24.61 m, respectively. Compared with the classical GNSS/INS integration algorithm, the constrained GNSS/INS algorithm can effectively reduce the positioning errors of the integrated navigation system when GNSS outages occur. However, the maximum position errors could still reach tens of meters (Figure 11). The north and east positioning errors could be reduced to less than 5 m and that of the up direction can be reduced to less than 1 m by using the MC-BP algorithm proposed in this paper. In detail, the maximum positioning errors in the north, east, and up directions were 4.94 m, 4.68 m, and 0.30 m, respectively, for the MC-BP algorithm.

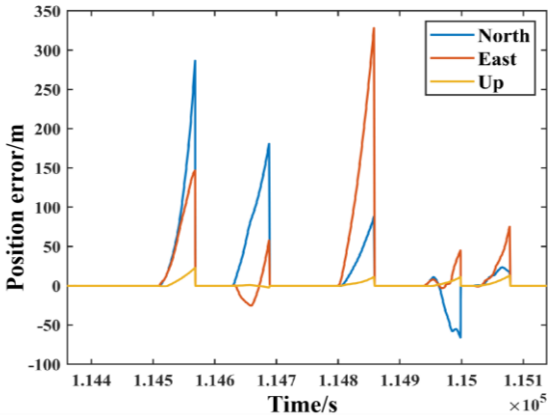


Figure 10. Positioning errors of classical GNSS/INS integration algorithm during GNSS outages in land experiment.

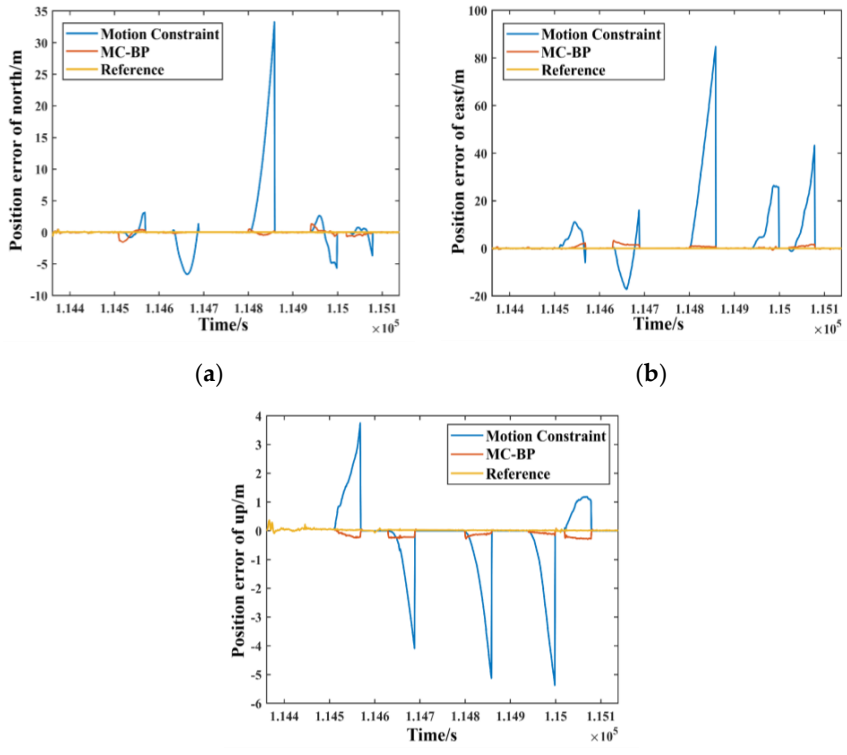


Figure 11. Vehicle land experiment's positioning errors of motion constraint algorithm (Motion Constraint) and MC-BP algorithm: (**a**) north direction; (**b**) east direction; (**c**) up direction.

The RMS values of the positioning errors for the vehicle land experiment during the period when GNSS was interrupted are displayed in Table 4. It can be seen from Table 4 that the RMS values of the classical GNSS/INS integration algorithm were the largest. When the motion-constrained GNSS/INS algorithm was added, the positioning accuracy was improved by 73%, 64%, and 67% in north, east, and up directions, respectively. When the proposed MC-BP algorithm was added, the positioning accuracy improved by 97%, 96%, and 93%, respectively.

Table 4. RMS values of positioning errors for vehicle land experiment (m).

Direction	Experimental Algorithm		
	Classical GNSS/INS Integration	Motion Constraint	MC-BP
North	42.750	11.305	0.955
East	48.362	17.457	1.554
Up	3.580	1.173	0.220

3.2.3. Analysis Results for the Marine Boat Experiment

The positioning errors of the classical GNSS/INS integration algorithm without constraints for the offshore experiment are shown in Figure 12. It indicates that the maximum positioning errors for the east and north direction were 179.52 m and 226.64 m, respectively. Similar to the vehicle land experiment, the maximum error occurred when the carrier was turned. Figure 13 shows the positioning errors of the motion-constrained GNSS/INS algorithm and the proposed MC-BP method. The maximum positioning errors using the motion-constrained GNSS/INS algorithm in the north, east, and up directions were 8.32 m, 27.64 m, and 0.91 m, respectively. It can be seen that the proposed method has the best performance. The maximum positioning errors of this method in the north, east, and up directions were 1.82 m, 7.64 m, and 0.21 m, respectively.

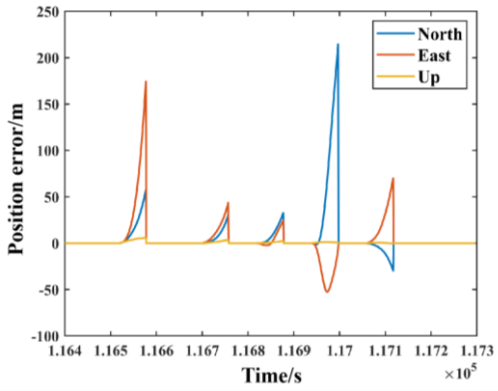


Figure 12. Positioning errors of classical GNSS/INS integration algorithm during GNSS outages in offshore experiment.

The statistical results of the positioning errors for the offshore boat experiment during the period when the GNSS signals were interrupted are displayed in Table 5. It can be seen from Table 5 that the RMS values for the classical GNSS/INS integration are 26.443 m, 21.935 m, and 1.170 m in the north, east, and up directions, respectively. When the motion-constrained GNSS/INS algorithm was introduced, the positioning accuracy for these three directions could be improved by 64%, 26%, and 49% in the three directions, respectively. When the proposed MC-BP algorithm is added, the positioning accuracy can be improved by 96%, 91%, and 53%, respectively.

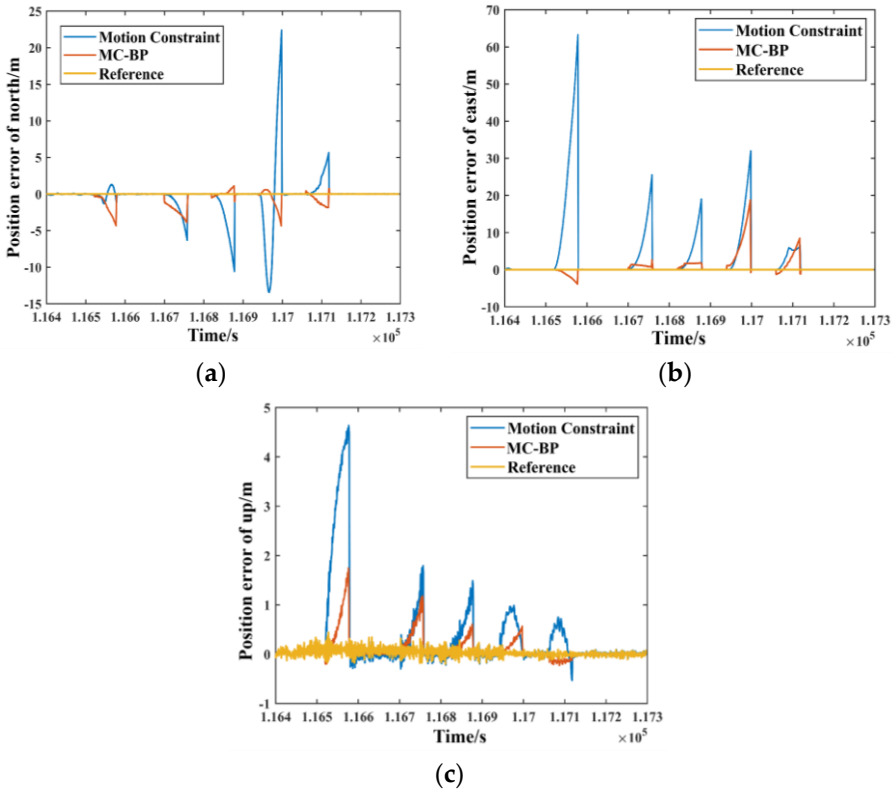


Figure 13. Marine boat experiment's positioning errors of motion constraint algorithm (Motion Constraint) and MC-BP algorithm: (**a**) north direction; (**b**) east direction; (**c**) up direction.

Table 5. RMS values of positioning errors for offshore boat experiment (m).

Direction	Experimental Algorithm		
	Classical GNSS/INS Integration	Motion Constraint	MC-BP
North	26.443	9.485	0.954
East	21.935	16.090	1.979
Up	1.170	0.593	0.545

4. Conclusions

During GNSS outages, the motion constraints for GNSS/INS integration algorithms have adaptability problems due to the complex state of vehicle motion. At the same time, GNSS pseudo-measurement predictions from neural network algorithms inevitably have outliers. With the aiming of solving these problems, a motion-constrained GNSS/INS integrated navigation method based on a BP neural network was proposed in this paper. In order to evaluate the performance of this method, simulated and real data experiments were conducted on land and offshore. The positioning errors of the integrated navigation system using the three solution strategies of a classical GNSS/INS integration algorithm, a motion-constrained GNSS/INS algorithm, and the proposed MC-BP method were compared. The experimental results show the following:

1. Motion constraints were not only applicable for vehicles on land, but also for boats when based on an appropriate constraint value.
2. For the vehicle land experiment, compared with the classical GNSS/INS integration algorithm, the positioning accuracy values using the MC-BP method were improved by over 90% in the north, east, and up directions. The improvement percentages were larger than 64% compared with the motion-constrained GNSS/INS algorithm. Similar performances were achieved by the offshore boat experiments.

3. The RMS of the positioning errors with the MC-BP method in north, east, and up directions are 0.955 m, 1.554 m, and 0.220 m when GNSS outages occur, and were 0.954 m, 1.979 m, and 0.545 m for the offshore boat experiment. By using the proposed MC-BP method, meter-level-positioning results could be provided by the GNSS/INS integration algorithm during GNSS outages.

In this paper, the velocity prediction was ignored, and a related model was not established using the BP neural network to consider the vehicle state of motion. In the future, a BP neural network based on a GNSS/INS adaptive integrated navigation method considering velocity predictions should be studied further.

Author Contributions: Data acquisition, Y.X. and K.W.; methodology, Y.X., K.W., C.J. and Z.L.; data processing and experimental analysis, C.Y., D.L., H.Z. and K.W.; article writing and typesetting, K.W., Y.X. and Z.L.; financial support, Y.X. and Z.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (42174035), Talent Introduction Plan for Youth Innovation Team in Universities of Shandong Province (innovation team of satellite position and navigation) and Shandong Provincial Natural Science Foundation, China (Grant No. ZR2021QD148).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ning, Z.; Zhang, K.; Wang, X.; Obaidat, M.S.; Guo, L.; Hu, X.; Hu, B.; Guo, Y.; Sadoun, B.; Kwok, R.Y. Joint computing and caching in 5G-Envisioned internet of vehicles: A deep reinforcement learning-based traffic control system. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 5201–5212. [CrossRef]
2. Chiang, K.W.; Duong, T.T.; Liao, J.K. The performance analysis of a real-time integrated INS/GPS vehicle navigation system with abnormal GPS measurement elimination. *Sensors* **2013**, *13*, 10599–10622. [CrossRef]
3. Kim, Y.; An, J.; Lee, J. Robust Navigational System for a Transporter Using GPS/INS Fusion. *IEEE Trans. Ind. Electron.* **2018**, *65*, 3346–3354. [CrossRef]
4. Noureldin, A.; Karamat, T.B.; Georgy, J. *Fundamentals of Inertial Navigation, Satellite-Based Positioning and Their Integration*, 2nd ed.; Springer: Berlin, Heidelberg, 2013.
5. Li, T.; Zhang, H.P.; Gao, Z.Z.; Niu, X.J.; El-Sheimy, N. Tight Fusion of a Monocular Camera, MEMS-IMU, and Single-Frequency Multi-GNSS RTK for Precise Navigation in GNSS-Challenged Environments. *Remote Sens.* **2019**, *11*, 610. [CrossRef]
6. Yang, Y.X.; Xu, T.H.; Song, L.J. Robust estimation of variance components with application in Global Positioning System network adjustment. *J. Surv. Eng.* **2005**, *131*, 107–112. [CrossRef]
7. Niu, X.J.; Nasser, S.; Goodall, C.; El-Sheimy, N. A Universal Approach for Processing any MEMS Inertial Sensor Configuration for Land-Vehicle Navigation. *J. Navig.* **2007**, *60*, 233–245. [CrossRef]
8. Xiao, Y.M.; Luo, H.Y.; Zhao, F.; Wu, F.; Gao, X.L.; Wang, Q.; Cui, L.Z. Residual Attention Network-Based Confidence Estimation Algorithm for Non-Holonomic Constraint in GNSS/INS Integrated Navigation System. *IEEE Trans. Veh. Technol.* **2021**, *70*, 11404–11418. [CrossRef]
9. Zhang, G.H.; Hsu, L.T. Intelligent GNSS/INS integrated navigation system for a commercial UAV flight control system. *Aerosp. Sci. Technol.* **2018**, *80*, 368–380. [CrossRef]
10. Meng, Y.; Gao, S.S.; Zhong, Y.M.; Hu, G.G.; Subic, A. Covariance matching based adaptive unscented Kalman filter for direct filtering in INS/GNSS integration. *Acta Astronaut.* **2016**, *120*, 171–181. [CrossRef]
11. Jiang, H.T.; Shi, C.; Li, T.; Dong, Y.T.; Li, Y.H.; Jing, G.F. Low-cost GPS/INS Integration with Accurate Measurement Modeling Using an Extended State Observer. *GPS Solut.* **2021**, *25*, 17. [CrossRef]
12. Li, T.; Zhang, H.P.; Gao, Z.Z.; Chen, Q.J.; Niu, X.J. High-accuracy positioning in urban environments using single-frequency multi-GNSS RTK/MEMS IMU integration. *Remote Sens.* **2018**, *10*, 205. [CrossRef]
13. Godha, S.; Cannon, M.E. GPS/MEMS INS Integrated System for Navigation in Urban Areas. *GPS Solut.* **2007**, *11*, 193–203. [CrossRef]
14. Yan, P.H.; Jiang, J.G.; Tang, Y.N.; Zhang, F.N.; Xie, D.P.; Wu, J.J.; Liu, J.H.; Liu, J.N. Dynamic adaptive low power adjustment scheme for Single-Frequency GNSS/MEMS-IMU/Odometer integrated navigation in the complex urban environment. *Remote Sens.* **2021**, *13*, 3236. [CrossRef]
15. Zhai, R.; Yuan, Y.B. A Method of Vision Aided GNSS Positioning Using Semantic Information in Complex Urban Environment. *Remote Sens.* **2022**, *14*, 869. [CrossRef]

16. Dissanayake, G.; Sukkarieh, S.; Nebot, E.; Durrant-Whyte, H. The Aiding of A low-cost Strapdown Inertial Measurement Unit Using Vehicle Model Constraints for Land Vehicle Applications. *IEEE Trans. Robot. Autom.* **2001**, *17*, 731–747. [CrossRef]
17. Klein, I.; Filin, S.; Toledo, T. Vehicle Constraints Enhancement for Supporting INS Navigation in Urban Environments. *Navigation-US.* **2011**, *58*, 7–15. [CrossRef]
18. Niu, X.J.; Zhang, H.P.; Chiang, K.W.; El-Sheimy, N. Using Land-Vehicle Steering Constraint to improve the Heading Estimation of MEMS GPS/INS Georeferencing Systems. In Proceedings of the Geomatics Conference and Symposium of Commission I, SPRS Convergence in Geomatics Shaping Canada's Competitive Landscape, Calgary, AB, Canada, 15–18 June 2010.
19. Liu, W.K.; Nong, Q.; Tao, X.L.; Zhu, F.; Hu, H.J. OD/SINS adaptive integrated navigation method with non-holonomic constraints. *Acta Geod. et Cartogr. Sinica.* **2022**, *51*, 9–17.
20. Jaradat, M.A.K.; Abdel-Hafez, M.F. Non-linear auto regressive delay-dependent INS/GNSS navigation system using neural networks. *IEEE Sens. J.* **2016**, *17*, 1105–1115. [CrossRef]
21. Fang, W.; Jiang, J.G.; Lu, S.Q.; Gong, Y.L.; Tao, Y.F.; Tang, Y.N.; Yan, P.H.; Luo, H.Y.; Liu, J.N. A LSTM algorithm estimating pseudo measurements for aiding INS during GNSS signal outages. *Remote Sens.* **2020**, *12*, 256. [CrossRef]
22. Sharaf, R.; Noureldin, A. Sensor integration for satellite-based vehicular navigation using neural networks. *IEEE Trans. Neural Netw.* **2007**, *18*, 589–594. [CrossRef]
23. El-Sheimy, N.; Chiang, K.-W.; Noureldin, A. The utilization of artificial neural networks for multi sensor system integration in navigation and positioning instruments. *IEEE Trans. Instrum. Meas.* **2006**, *55*, 1606–1615. [CrossRef]
24. Zhang, X.H.; Zhou, Y.H.; Zhu, F.; Hu, H.J. A new vehicle motion constraint model with parameter autonomous learning and analysis on inertial drift error suppression. *Acta Geod. Cartogr. Sinica.* **2022**, *51*, 1249–1258.
25. Brossard, M.; Barrau, A.; Bonnabel, S. AI-IMU dead reckoning. *IEEE Trans. Intell. Transp. Syst.* **2020**, *5*, 585–595. [CrossRef]
26. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [CrossRef]
27. Wu, F.M.; Yang, Y.X. GPS/INS/Odometer integrated navigation algorithm with prior velocity in land vehicle system. *J. Astro.* **2010**, *31*, 2314–2320.
28. Han, S.L.; Wang, J.L. Integrated GPS-INS navigation system with Dual-rate Kalman filter. *GPS Solut.* **2012**, *16*, 389–404. [CrossRef]
29. Niu, X.J.; Ding, L.Y.; Kuang, J.; Wu, Y.B. A MENS IMU and motion constraint based positioning algorithm for shared bicycles. *J. Chin. Inert. Technol.* **2021**, *29*, 300–306.
30. Nie, T.; He, B.; Bi, G.L.; Zhang, Y.; Wang, W.S. A Method of Ship Detection under Complex Background. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 159–177. [CrossRef]