Research Article

An Adaptive UKF Based SLAM Method for Unmanned Underwater Vehicle

Hongjian Wang, Guixia Fu, Juan Li, Zheping Yan, and Xinqian Bian

College of Automation, Harbin Engineering University, Harbin 150001, China

Correspondence should be addressed to Hongjian Wang; cctime99@163.com

Received 24 June 2013; Revised 7 September 2013; Accepted 10 September 2013

Academic Editor: Yuri Vladimirovich Mikhlin

Copyright © 2013 Hongjian Wang 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.

This work proposes an improved unscented Kalman filter (UKF)-based simultaneous localization and mapping (SLAM) algorithm based on an adaptive unscented Kalman filter (AUKF) with a noise statistic estimator. The algorithm solves the issue that conventional UKF-SLAM algorithms have declining accuracy, with divergence occurring when the prior noise statistic is unknown and time-varying. The new SLAM algorithm performs an online estimation of the statistical parameters of unknown system noise by introducing a modified Sage-Husa noise statistic estimator. The algorithm also judges whether the filter is divergent and restrains potential filtering divergence using a covariance matching method. This approach reduces state estimation error, effectively improving navigation accuracy of the SLAM system. A line feature extraction is implemented through a Hough transform based on the ranging sonar model. Test results based on unmanned underwater vehicle (UUV) sea trial data indicate that the proposed AUKF-SLAM algorithm is valid and feasible and provides better accuracy than the standard UKF-SLAM system.

1. Introduction

The simultaneous localization and mapping (SLAM) algorithm [1, 2] was first proposed by Smith, Self, and Cheeseman in 1988 to provide localization and map building for mobile robots and is now widely used in many different mobile robot systems. The SLAM algorithm was first used for unmanned underwater vehicle (UUV) navigation in September 1997 in a collaborative project between the Naval Undersea Warfare Center (NUWC) and Groupe d'Etudes Sous-Marines de l'Atlantique (GESMA). The objective of the trial using SLAM was to get a UUV starting in an unknown location and without previous knowledge of the environment to build a map using its onboard sensors and then use the same map to compute the robot's location.

Given the recent wider use of UUV in the marine environment, it is notable that truly autonomous SLAM-based UUV navigation is still lacking. However, it is challenging to develop SLAM-based UUV navigation due to factors such as system complexity, weak perception, nonstructure, increasing system noise, and unknown statistical characteristics.

SLAM solutions can be divided into two categories: a nonprobabilistic probability estimation method and methods primarily based on probability estimation; Table 1 gives the

summary of SLAM methods. The probability estimation method first developed [3] was the EKF based SLAM algorithm, which suffers difficulty solving data association problems, high computational costs due to the calculation of Jacobian matrices, and inconsistency due to errors introduced during linearization. In trying to reduce storage and computational requirements, Thrun et al. [4] proposed a SLAM algorithm based on sparse extended information filter. However, this method is only applicable for creating feature maps and requires the existence of features that are easy to extract and distinguish in the environment, such as point, line, and face features. More recently, Montemerlo et al. [5] proposed a Rao-Blackwellized particle filter based SLAM method (FastSLAM), where each particle stores its map and robot positioning result. However, this algorithm results in a calculation and storage problem proportional to the number of particles and is unable to avoid the disadvantages of particle degradation and sample dilution.

The unscented Kalman filter (UKF) [6, 7] is a nonlinear filter based on unscented transform (UT). For nonlinear systems, UKFs avoids linearization of the state and measurement equations. Additionally, the UKF principle is simple and easy to implement as it does not require the calculation of Jacobians at each time step, and the UKF is accurate up to second

TABLE 1: Summery of SLAM methods.

Algorithm	Drawbacks
EKF-SLAM	(1) Difficulty to solve data association problems (2) High computational costs (3) Inconsistency due to errors introduced during linearization
EIF-SLAM	(1) Only applicable for creating feature maps (2) Requires the existence of features, such as point, line, and face features.
FastSLAM	(1) Calculation and storage problem proportional to the number of particles (2) Particle degradation and sample dilution

order moments in the probability distribution function propagation whereas the EKF is accurate up to first order moment [8]. However, when an UKF is used in underwater SLAM, it needs to predict mathematical model of the system and a priori knowledge of the noise statistics. In many practical applications, prior statistics of the noise is unknown or not accurate. Even if this information is known, the statistical characteristics easily change due to internal and external uncertainties that reflect strong time-varying characteristics. Thus, a conventional UKF does not have the adaptive ability to respond to changes in the noise statistics, which can lead to large estimation errors and even cause divergence in the case of unknown and time-varying noise statistics [9–11].

In order to solve the above problem, we apply an adaptive UKF (i.e., an adaptive unscented Kalman filter, or AUKF) filtering algorithm to underwater SLAM. By introducing a modified (i.e., suboptimal and unbiased) Sage-Husa maximum a posterior (MAP) noise statistic estimator, the new algorithm provides online estimation of the statistical parameters of unknown system noises and restrains filtering divergence. In addition, the method uses a covariance matching criterion approach to determine the convergence of the filter. When the filter is divergent, the proposed method introduces an adaptive fading factor to correct prediction error covariance, adjust the filter gain matrix K , and suppress filter divergence, thus enhancing the fast tracking capability of the filter. Test results based on sea trial data of UUV indicate that the proposed AUKF-SLAM algorithm provides better navigational accuracy than a conventional UKF-SLAM algorithm.

2. Adaptive UKF Algorithm

2.1. UKF Algorithm. Unscented Kalman filters were first proposed by Julier and Uhlmann [12]. The algorithm's main principle is to select a number of sampling points in the state distribution (sigma points), which can completely capture the true mean and covariance of state distribution. Those sigma points are then substituted into the nonlinear function to obtain the corresponding nonlinear function point set, and it can solve the mean and covariance after transformation by these points.

The mean, estimate variance, and measurement variance obtained from the unscented transform are introduced into a gradually recursive process of Kalman filtering to obtain the UKF. The main steps of a UKF algorithm are as follows.

(1) Initialization

$$\bar{x}_0 = E[x_0], \quad (1)$$

$$P_0 = E[(x_0 - \bar{x}_0)(x_0 - \bar{x}_0)^T].$$

(2) For $k \in \{1, \dots, \infty\}$

(i) calculate sigma points

$$\chi_{k-1} = [\bar{x}_{k-1} \quad \bar{x}_{k-1} \pm \sqrt{(n_x + \lambda) P_{k-1}}]. \quad (2)$$

(ii) UKF prediction

$$\chi_{k|k-1} = f(\chi_{k-1}, u_{k-1}), \quad (3)$$

$$\bar{x}_{k|k-1} = \sum_{i=0}^{2n_x} W_i^{(m)} \chi_{i,k|k-1}, \quad (4)$$

$$P_{k|k-1} = \sum_{i=0}^{2n_x} W_i^{(c)} [\chi_{i,k|k-1} - \bar{x}_{k|k-1}] [\chi_{i,k|k-1} - \bar{x}_{k|k-1}]^T + Q, \quad (5)$$

$$z_{k|k-1} = h(\chi_{k|k-1}, u_{k-1}), \quad (6)$$

$$\bar{z}_{k|k-1} = \sum_{i=0}^{2n_x} W_i^{(m)} z_{i,k|k-1}. \quad (7)$$

(iii) UKF update

$$P_{\bar{z}_k \bar{z}_k} = \sum_{i=0}^{2n_x} W_i^{(c)} [z_{i,k|k-1} - \bar{z}_{k|k-1}] [z_{i,k|k-1} - \bar{z}_{k|k-1}]^T + R, \quad (8)$$

$$P_{x_k z_k} = \sum_{i=0}^{2n_x} W_i^{(c)} [\chi_{i,k|k-1} - \bar{x}_{k|k-1}] [z_{i,k|k-1} - \bar{z}_{k|k-1}]^T, \quad (9)$$

$$K_k = P_{x_k z_k} P_{\bar{z}_k \bar{z}_k}^{-1}, \quad (10)$$

$$\bar{x}_k = \bar{x}_{k|k-1} + K_k (z_k - \bar{z}_{k|k-1}), \quad (11)$$

$$P_k = P_{k|k-1} - K_k P_{\bar{z}_k \bar{z}_k} K_k^T, \quad (12)$$

where Q is the system noise covariance, R is the observation noise covariance, K is the Kalman gain, and W_i is the weight of the mean and covariance.

2.2. Adaptive UKF Algorithm. Adaptive filtering technology has become a focus of research attempting to solve the filter divergence problem caused by the inaccurate statistical properties of the noise and the mathematical model itself. Sage-Husa suboptimal unbiased maximum a posterior (MAP) noise estimators have the advantage that their recursive formula is simple, and the principle is clear and easy to implement. Therefore we chose to use this type of system to estimate unknown system noise.

2.2.1. System Noise Estimation Method. Suppose a class of nonlinear dynamic systems is described as

$$\begin{aligned} x_k &= \mathbf{f}(x_{k-1}, u_{k-1}) + W_{k-1}, \\ z_k &= \mathbf{h}(x_k) + V_k, \end{aligned} \quad (13)$$

where x_k represents the state vector of the system at time k , u_{k-1} represents the control, z_k represents the measurement value of the state at time k , and W_{k-1} and V_k are independent white Gaussian noise with time-varying means of q_k and r_k and covariances of Q_k and R_k , respectively. Note that Q_k is a nonnegative definite symmetric matrix, while R_k is positive definite symmetric matrix.

Emphasis should be placed on recent data when estimating time-varying noise statistics; that is, the algorithm should gradually forget data that is too old. In this paper, we adopt a fading memory weighted exponent method to design a time-varying noise statistics estimator. According to the literature [13], we selected the weighting coefficient to satisfy

$$\beta_i = \beta_{i-1}b, \quad 0 < b < 1, \quad \sum_{i=1}^k \beta_i = 1 \quad (14)$$

which can be written as

$$\begin{aligned} \beta_i &= d_k b^{i-1}, \quad i = 1, \dots, k, \\ d_k &= \frac{(1-b)}{(1-b^k)}. \end{aligned} \quad (15)$$

The recursive formula of a fading memory time-varying noise statistical estimator described by $\hat{q}_k, \hat{Q}_k, \hat{r}_k, \hat{R}_k$ is

$$\begin{aligned} \hat{q}_k &= (1-d_k) \hat{q}_{k-1} + d_k \left[\hat{x}_k - \sum_{i=0}^{2n_x} W_i^{(m)} \mathbf{f}(\chi_{i,k-1}) \right] \\ \hat{Q}_k &= (1-d_k) \hat{Q}_{k-1} \\ &+ d_k \left[\mathbf{K}_k \varepsilon_k \varepsilon_k^T \mathbf{K}_k^T + \mathbf{P}_k \right. \\ &\quad \left. - \sum_{i=0}^{2n_x} W_i^{(c)} (\chi_{i,k|k-1} - \hat{x}_{k|k-1}) (\chi_{i,k|k-1} - \hat{x}_{k|k-1})^T \right], \\ \hat{r}_k &= (1-d_k) \hat{r}_{k-1} + d_k \left[z_k - \sum_{i=0}^{2n_x} W_i^{(m)} \mathbf{h}(\chi_{i,k|k-1}) \right], \\ \hat{R}_k &= (1-d_k) \hat{R}_{k-1} \\ &+ d_k \left[\varepsilon_k \varepsilon_k^T - \sum_{i=0}^{2n_x} W_i^{(c)} \times (z_{i,k|k-1} - \hat{z}_{k|k-1}) \right. \\ &\quad \left. \times (z_{i,k|k-1} - \hat{z}_{k|k-1})^T \right], \end{aligned} \quad (16)$$

where $\varepsilon_k = z_k - h(\hat{x}_{k|k-1})$ is the output residual sequence of the UKF.

2.2.2. Filter Divergence Suppression Method. Since suboptimal Sage-Husa filters often diverge, in this paper we judge whether filtering divergence is occurring according to convergence conditions derived from the covariance matching criterion. If the convergence conditions are satisfied, the Sage-Husa algorithm is applied. If filtering divergence occurs, the proposed method calculates an adaptive weighting coefficient λ_k through a computational fading factor formula and applies this coefficient to correct $\mathbf{P}_{k|k-1}$; thus, the role of the observables is strengthened and the filter divergence is suppressed.

The convergence conditions can be written as

$$v_k^T v_k \leq S \cdot \text{tr} [E(v_k v_k^T)], \quad (17)$$

where $S (S \geq 1)$ is an adjustable coefficient presetting and v_k is the residual sequence, such that $v_k = z_k - \mathbf{h}(\bar{x}_{k|k-1})$.

The correction method of covariance $\mathbf{P}_{k|k-1}$ is

$$\begin{aligned} \mathbf{P}_{k|k-1} &= \lambda_k \cdot \sum_{i=0}^{2n_x} W_i^{(c)} [\chi_{i,k|k-1} - \bar{x}_{k|k-1}] \\ &\quad \times [\chi_{i,k|k-1} - \bar{x}_{k|k-1}]^T + Q_{k-1}. \end{aligned} \quad (18)$$

The adaptive weighting coefficient λ_k is calculated based on the fading factor formula [14, 15]

$$\lambda_k = \begin{cases} \lambda_0 & \lambda_0 \geq 1 \\ 1 & \lambda_0 < 1, \end{cases}$$

$$\lambda_0 = \text{tr} \frac{[N_k]}{\text{tr} [M_k]},$$

$$N_k = \text{tr} (C_{0,k} - R)^T,$$

$$\begin{aligned} M_k &= \text{tr} \left(\sum_{i=0}^{2n_x} W_i^{(c)} [z_{i,k|k-1} - \bar{z}_{k|k-1}] [z_{i,k|k-1} - \bar{z}_{k|k-1}]^T \right), \\ C_{0,k} &= \begin{cases} v_k v_k^T & k = 1 \\ \frac{\rho C_{0,k} + v_k v_k^T}{1 + \rho} & k > 1, \end{cases} \end{aligned} \quad (19)$$

where $\text{tr}(\cdot)$ accounts for matrix trace. Here, $0 < \rho \leq 1$ is a forgetful factor (typically about 0.95) used to increase the filter's tracking ability, with greater values of the factor creating a smaller proportion of information before time k and causing a more prominent residual vector effect. This method has a strong tracking ability for sudden status changes but still keeps tracking for slowly varying state and mutation status changes when the filter reaches a steady state.

2.2.3. Realization of Adaptive UKF Algorithm

(1) *Initialization.* State initialization is done according to (1) and (20)

$$\hat{q}_0 = q_0, \quad \hat{Q}_0 = Q_0. \quad (20)$$

(2) *Time Update.* Given values for the variables $\hat{x}_{k-1}$ and $\mathbf{P}_{k-1}$, we solve $\bar{x}_{k|k-1}$ and $\bar{z}_{k|k-1}$ based on the unscented transform,

that is, using (4) and (7), and covariance $\mathbf{P}_{k|k-1}$ is predicted by (5).

(3) *Convergence Judgment.* At this stage, the method uses (17) to judge whether the filter is converging. If the filter is converging, move to the next step; otherwise correct $\mathbf{P}_{k|k-1}$ using (18)~(19).

(4) *Measurement Update.* Obtain observation covariances $\mathbf{P}_{\tilde{z}_k \tilde{z}_k}$ and $\mathbf{P}_{x_k \tilde{z}_k}$ and filter gain $\mathbf{K}_k$ according to (8)~(10), and then do a measurement update using (11)~(12).

(5) *Recursively Estimate System Statistical Noise Characteristics.* Recursively estimate the system's statistical noise characteristics according to (16).

3. SLAM Algorithm Based on Adaptive UKF

3.1. SLAM System Model

3.1.1. AUV Nonlinear Dynamic Model. As seen from Figure 1, L is the global coordinate system established with the initial location and bow of the UUV, where x , y , and ψ describe the position and heading of the UUV within this system. While V is the UUV vessel frame system and S is the sonar coordinate system. The North-East coordinates are given by E , with its North direction based on magnetic North. Obviously, $z_\psi = \psi + \psi_{L_0}$, where z_ψ is the heading of the UUV as measured by its OCTANS.

Note that there is a distance offset (1.85 m in the X direction, 0.65 m in the Y direction, and a very small deviation in the Z direction that can be ignored) between the mounting positions of three ranging sonar and the UUV's center of gravity. The coordinate systems S and V are shown in more detail in Figure 2. We assume that the positions of the origin of S and V in E are (S_x, S_y) and (V_x, V_y) , respectively, and that three ranging sonar are mounted together (i.e., their mounting positions are the same). This gives a mounting deviation of three ranging sonar in V of $a = 1.85$ m and $b = 0.65$ m, where a is the deviation in the X direction and b is the deviation in the Y direction. The mounting angle of the middle sonar is ϕ , with mounting angles of $\phi \pm 7.5$ for the left and right sonar. According to Figure 2, using the ranging sonar as a reference we get

$$\begin{aligned} S_x &= V_x + \sqrt{a^2 + b^2} * \cos(\phi - z_\psi), \\ S_y &= V_y + \sqrt{a^2 + b^2} * \sin(\phi - z_\psi). \end{aligned} \quad (21)$$

The method uses a simple 4 DOF (degree of freedom) constant velocity kinematics to predict how the state will evolve from time $k-1$ to time k :

$$\begin{bmatrix} x \\ y \\ z \\ \psi \\ u \\ v \\ w \\ r \end{bmatrix}_k = \begin{bmatrix} x + uT \cos(\psi) - vT \sin(\psi) \\ y + uT \sin(\psi) + vT \cos(\psi) \\ z + wT \\ \psi + rT \\ u \\ v \\ w \\ r \end{bmatrix}_{k-1} + \mathbf{n}_k, \quad (22)$$

FIGURE 1: Global coordinate system, the UUV vessel frame system and the sonar coordinate system.

FIGURE 2: Installation position and angle deviation of the ranging sonar.

where $[x, y, z, \psi]$ is the position and heading of the UUV in L ; $[u, v, w, r]$ gives the line velocity and angle velocity of the UUV in V , and k is the sample time. In this equation, $\mathbf{n}(k)$ is the portion of the system noise with a time-varying mean and covariance, with the covariance of vector $\mathbf{n}$ given by system noise $\mathbf{Q}$

$$\begin{aligned} E[\mathbf{n}_k] &= \mathbf{q}_k, \\ E[(\mathbf{n}_k - \mathbf{q}_k)(\mathbf{n}_j - \mathbf{q}_j)^T] &= \delta_{kj} \mathbf{Q}_k, \end{aligned} \quad (23)$$

where δ_{kj} is a delta function.

3.1.2. Feature Model. The feature data used in this paper is derived from measurements of the structured port environment, so the algorithm itself is what chooses for line features with which to build a feature map. The line feature model used in the proposed method is

$$\mathbf{X}_{f_i,k} = \mathbf{X}_{f_i,k-1} \quad i = 1, \dots, n. \quad (24)$$

3.1.3. Observation Model. The UUV uses a Doppler velocity log (DVL), compass, and pressure sensor to provide direct measurements of the vehicle's velocity, heading, and depth, respectively. Thus, the observation model is linear, and so the common model becomes

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_{k|k-1} + \mathbf{m}_k, \quad (25)$$

where $\mathbf{z}$ is the observation vector, $\mathbf{m}$ is a white Gaussian with zero mean, and $\mathbf{H}$ varies with changes in the measurement:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \quad (26)$$

3.1.4. Ranging Sonar Model. The transmission beam of a ranging sonar creates a conical surface, which is a fan in a two-dimensional plane. David Ribas et al. [16] proposed the underwater mechanical scanning imaging sonar model based on the terrestrial single beam ranging sonar model [17]. In this paper, we determine the location of line features in the environment using the measurement data from the ranging sonar.

In Figure 3, α represents the horizontal beam width, β represents the incidence angle, and ρ^S is the range at which the bin was measured from the sensor. Reference frame S defines the position of the transducer head at the moment the sensor obtains a particular bin, where $[x_S^B, y_S^B, \theta_S^B]$ is the transformation defining the position of S with respect to B to the chosen base reference. Both $[x_S^B, y_S^B, \theta_S^B]$ and ρ^S are obtained from information in the data buffer.

To emulate the effect of the horizontal beam width, the model uses a set of i values at a given resolution within an aperture of $\pm\alpha/2$ around the direction in which the transducer is oriented. This set of values is also referred to as θ_S^B , where

$$\theta_S^B - \frac{\alpha}{2} \leq \theta_i^B \leq \theta_S^B + \frac{\alpha}{2}. \quad (27)$$

Each value θ_i^B represents a bearing parameter for a line tangent of the arc that models the horizontal beam width. As stated earlier, not only are all lines tangent to the arc candidates for line features, but the ones inside the maximum incidence angle limits of $\pm\beta$ are also considered candidates. For this reason, the algorithm takes each k value at a given resolution for each value of θ_i^B and within an aperture of $\pm\beta$; that is,

$$\theta_i^B - \beta \leq \theta_{i,k}^B \leq \theta_i^B + \beta. \quad (28)$$

The results of this operation are $i \times k$ different values of θ_i^B for the given aperture. These are the bearings for a set of lines representing all the possible candidates compatible with the bin. Given the geometry of the problem, the range parameter $\rho_{i,k}^B$ corresponding to each one of the $\theta_{i,k}^B$ bearings is

$$\rho_{i,k}^B = x_S^B \cos(\theta_{i,k}^B) + y_S^B \sin(\theta_{i,k}^B) + \rho^S \cos(\theta_{i,k}^B). \quad (29)$$

FIGURE 3: Ranging sonar model to distinguish line features.

3.2. Feature Extraction. In the sea trial, the application environment of the SLAM algorithm is a cross-section of the ports, dams, and other environment structures. Note that the scanning surface of the sonar and the vertical wall or other vertical extensions of the surface will create line features in the resulting acoustic image. However, the parameters of such static line features will not change as the sonar position changes.

The most popular line feature extraction methods include the split-and-merge method [18] proposed by Pavlidis, the RANSAC method [19] proposed by Fischler and Bolles, and the Hough transform (HT) [20] proposed by Illingworth and Kittler. Between these choices, HT is the most popular line feature extraction method, and a number of others have developed many methods to improve HT line feature extraction [21–23]. In this paper, we use the HT method to extract line features from the ranged sonar data. The HT is a voting scheme where the distance values of each ranged sonar image accumulates evidence for the presence of a line feature. Units that get the most votes in the HT space correspond to line features in the actual environment.

3.2.1. Data Processing of Ranging Sonar Data. A data buffer helps to separate and manage the stream of measurements produced by the continuous arrival of the sonar beams. The buffer stores variables such as the range and bearing for each bin used in the voting, the position and heading in the North-East coordinate system E , and the transmit angle of the beams so that ρ - θ parameter pairs used to present line features are extracted based on HT.

The steps to process the data set from each ranged sonar scan [24, 25] are given below, using the left sonar as an example. First, the data buffer is set, the data is loaded with the range values, and a 0-1 matrix is built where the units without range values are set to 0 and units with range values are set to 1. In the second step, the transmit angle of the sonar in S , the time, the position and heading of the UUV in E , and the position and transmit angle of the sonar in E are stored into the data buffer. The third step defines the base frame B , where B is the current position of the sonar head when the voting is performed. Finally, the position and heading of the sonar

instrument in B at every moment is acquired and stored in the data buffer.

3.2.2. Hough Transform. There are three steps to extract line features from the sonar image data. First, the data from all three sonar instruments are loaded, and distance resolution, angle resolution, and threshold value are defined. Secondly, the accumulator is defined, and the index values of the nonzero elements of the accumulator are found. Finally, we use (27)~(29) to vote, and the ρ - θ parameter pairs that get the most votes are used to represent the detected line features in B .

3.3. Data Association. Once the model has extracted line features in the environment based on the HT algorithm, it needs to create an environment map and improve the state estimation of the UUV by fusing the detected line features. The next step is then to perform data association [26] to determine if a measured line Z_i^V corresponds to any of the features F_j , $j = 1, \dots, n$ already existing in the map and so used to update the system or if it is a new line and has to be incorporated into the map. To make this distinction, the most popular individual compatibility nearest neighbor data association algorithm is used to select the best candidate.

Given the transformation of B and V , the position of the i th line measurement in V can be represented by

$$Z_i^V = [\rho_i^V \ \theta_i^V]^T. \quad (30)$$

If we assume the position of the line feature j already exists in the map in L ; that is,

$$X_j^L = \begin{bmatrix} \rho_j \\ \theta_j \end{bmatrix}. \quad (31)$$

Then the position of line feature j in V is

$$X_j^V = \begin{bmatrix} \rho_j - x \cos \theta_j - y \sin \theta_j \\ \theta_j - \psi \end{bmatrix}. \quad (32)$$

And line feature j corresponds to the i th line measurement, where

$$\begin{aligned} z_{i,k}^V &= h_j(x_k, \mathbf{s}_i), \\ h_j(x_k, \mathbf{s}_i) &= \begin{bmatrix} \rho_j - x \cos \theta_j - y \sin \theta_j \\ \theta_j - \psi \end{bmatrix} + \mathbf{s}_i. \end{aligned} \quad (33)$$

Here, ρ_i^V and θ_i^V are the parameters of the line features in the V frame of reference, ρ_i and θ_i are the parameters of the line features presented in the L frame of reference, and $\mathbf{s}_i$ is the noise with a zero-mean white noise with covariance $\mathbf{R}$ affecting the line feature observation.

The proposed method uses an innovation term ν_{ij} to calculate the discrepancy between the measurement and its prediction, with its associate covariance matrix $\mathbf{S}_{ij}$ obtained by

$$\begin{aligned} \nu_{ij,k} &= z_{i,k}^V - z_{j,k|k-1}, \\ z_{j,k|k-1} &= h_j(\bar{x}_{k|k-1}, \mathbf{s}_i), \\ \mathbf{S}_{ij,k} &= \mathbf{P}_{\bar{z}_k \bar{z}_k} + \mathbf{R}_k. \end{aligned} \quad (34)$$

To determine if the correspondence is valid, an individual compatibility (IC) test [27] using the Mahalanobis distance is carried out

$$\mathbf{D}_{ij}^2 = \nu_{ij,k}^T \mathbf{S}_{ij,k}^{-1} \nu_{ij,k} < \chi_{d,\alpha}^2, \quad (35)$$

where $d = \dim(h_j)$ and α is the desired confidence level.

Data association is only performed if and when a line feature is detected based on HT. If the data association is successful, that is, the line feature exists in the map, then the model updates the state estimate. Otherwise, state augmentation is carried out where the new measurement is added to the current state vector as a new feature. However, the algorithm cannot do this augmentation directly because the new feature is represented in V . Thus, the algorithm must first perform a change of reference using the following:

$$\widehat{\mathbf{X}}(k) = \begin{bmatrix} \widehat{\mathbf{X}}_V(k) \\ \widehat{\mathbf{X}}_{f_1}(k) \\ \vdots \\ \widehat{\mathbf{X}}_{f_n}(k) \end{bmatrix} \Rightarrow \widehat{\mathbf{X}}(k)^+ = \begin{bmatrix} \widehat{\mathbf{X}}_V(k) \\ \widehat{\mathbf{X}}_{f_1}(k) \\ \vdots \\ \widehat{\mathbf{X}}_{f_n}(k) \\ \widehat{\mathbf{X}}_V(k) \oplus z_i^V(k) \end{bmatrix}. \quad (36)$$

4. AUKF-SLAM Algorithm Verification Based on Sea Trial Data

4.1. Test Conditions. The data used in the test conducted in this work comes from a sea trial completed in October 2010 near Dalian Xiaoping Island using a UUV developed by the authors. The navigation route of the UUV was from point A to point B in Figure 4. As configured for the trails, the UUV possessed a number of different sensors including DVL, OCTANS, depth sensor, and three ranging sonar which mounted in the horizontal frame as a whole set to observe the environment.

The initial position of the UUV (point A) was longitude 121.5231°, north latitude 38.8271°, and the trial ended at longitude 121.5083°, North latitude 38.8328° (position B). The initial heading of the UUV was -70.70°. During the trail, the UUV stayed near the surface of the water so that GPS was available throughout the trial. The total navigation time was 17 minutes and 6 seconds.

DVL can measure a UUV's current velocity, bottom tracking speed, and so on. However, in the sea trial DVL was only used to measure the bottom track speed while OCTANS was used to measure the UUV's heading in real-time, that is, the angle between the front of the UUV and magnetic North. The pressure sensor provided depth data by measuring the water pressure, and three ranging sonar provided online environment perception and measurement. Three ranging sonar mounted in the horizontal frame is shown as Figure 5.

A general description of the AUKF-SLAM algorithm is given in Figure 6.

4.2. Acquisition of Embankment Measurement Points Using Ranging Sonar. Given a true trajectory as provided by GPS, we can obtain embankment measurement features by fusing this GPS information with the ranging sonar data and then

FIGURE 4: Satellite image of the sea trial test area.

FIGURE 5: The ranging sonar mounted in the horizontal frame.

performing a coordinate transformation for the measurement data in a two-dimensional plane. Assume the position of point P in S is $X_P^S = [x_S \ y_S]^T$. Then for each ranging sonar, we have

$$X_P^S = \begin{bmatrix} x_S \\ y_S \end{bmatrix} = \begin{bmatrix} \rho_i \times \sin\left(\frac{\theta_i \times \pi}{180}\right) \\ \rho_i \times \cos\left(\frac{\theta_i \times \pi}{180}\right) \end{bmatrix}, \quad (37)$$

where ρ_i and θ_i represent distance values and mounting angles of the three sonar instruments. For the sea trial, the mounting angles were $\theta_1 = 7.5^\circ$, $\theta_2 = 0^\circ$, and $\theta_3 = -7.5^\circ$, corresponding to the left ($i = 1$), middle ($i = 2$), and right ($i = 3$) sonar.

The relationship between V and S is shown in Figure 2, with the position of point P in V given by

$$X_P^V = \begin{bmatrix} x_V \\ y_V \end{bmatrix} = \begin{bmatrix} x_S + a \\ y_S + b \end{bmatrix}. \quad (38)$$

FIGURE 6: Basic principle of the AUKF-SLAM algorithm.

FIGURE 7: Feature measurement of the embankment.

The position of V relative to L is $X_V^L = [x_L \ y_L \ \varphi_L]^T$, so we can obtain the position of point P in L using the following synthesis operator

$$X_P^L = X_V^L \oplus X_P^V = \begin{bmatrix} x_L + x_V \cos \varphi_L - y_V \sin \varphi_L \\ y_L + x_V \sin \varphi_L + y_V \cos \varphi_L \end{bmatrix}. \quad (39)$$

Figure 7 gives the embankment measurement, where the green, red, and blue points were acquired by the left, middle, and right sonar, respectively.

4.3. Line Feature Extraction of the Embankment Based on Ranging Sonar Model. The proposed model can extract line features using an HT form ranging sonar model based on the following assumptions:

- (a) angle resolution of the HT space is 5° and the distance resolution is 1 m;
- (b) the largest number of votes is selected as the threshold;
- (c) a 10° arc is used to model the horizontal beam width of the ranging sonar; that is, $\alpha = 10^\circ$;
- (d) the model does not consider uncertainty in the vertical beam width;
- (e) assume that there is only an echo signal when transmit beam of the ranging sonar is parallel to the vertical extension surface; that is, $\beta = 0^\circ$.

In the sea trial the algorithm extracted six line features L1~L6. The ρ - θ parameter, time, and position of the UUV when the feature was detected are given in Table 2 for each line feature.

4.4. Test Validation Based on Sea Trial Data. To verify the performance of the AUKF-SLAM algorithm, we compared the results of tests using AUKF-SLAM, UKF-SLAM, and EKF-SLAM algorithms against trial data with the UUV and assuming that the statistical properties of the system noise was unknown in all cases.

4.4.1. Test Conditions. The actual statistical characteristics of the system noise during the field trial was unknown, so for this work we assumed that actual system noise behavior was based on two laws, one time-varying and one constant. Using these laws, we conducted both a time-varying noise test and a constant noise test. Table 3 gives the results of the system noise tests and the resulting laws, where k is the step number. The observation noise is

$$R = \text{diag}([0.04^2 \ 0.04^2 \ 0.04^2 \ 0.04^2 \ 0.04^2]). \quad (40)$$

4.4.2. Test Results. Figures 8~15 are the test results given for both time-varying and constant noise conditions. Figures 8 and 12 compare the trajectory estimations from the different algorithms against the UUV's GPS trajectory with time-varying noise and constant noise, respectively. Figures 9 and 13 compare position errors in the East direction between the different algorithms for the different types of noise, while Figures 10 and 14 provide a comparison of the position errors in the North direction. Figures 11 and 15 show the adaptive weighting coefficient of AUKF-SLAM algorithm in time-varying and constant noise test, respectively.

4.4.3. Analysis of Test Results. The heading z_ψ measured by the OCTANS is relative to the magnetic north of E coordination system shown in Figure 1, and it should be transformed to the heading ψ in the L coordinate system for state updating. The heading ψ is shown as Figure 16.

(1) Performance Analysis of the AUKF-SLAM Algorithm. We calculate the estimation error of the algorithm by

$$\delta_k = \begin{bmatrix} \delta_{e,k} \\ \delta_{n,k} \end{bmatrix} = \begin{bmatrix} \hat{x}_k - x_k \\ \hat{y}_k - y_k \end{bmatrix}, \quad (41)$$

FIGURE 8: Comparison of the trajectory estimations with time-varying noise.

FIGURE 9: Comparison of position error in the East direction with time-varying noise.

where $\delta_{e,k}$ and $\delta_{n,k}$ are the estimation error in the East and North directions at time k , respectively, while (x_k, y_k) and $(\hat{x}_k, \hat{y}_k)$ are the true position and estimated position of the UUV at time k . Note that $|\delta_{e,k}|$ and $|\delta_{n,k}|$ represent absolute values of the error in the East and North directions at time k , respectively. Obviously, small values of $|\delta_e|$ and $|\delta_n|$ indicate higher accuracy in the filtering algorithm. Figures 8~10 and Figures 11~14 show that $|\delta_e|$ and $|\delta_n|$ for the proposed AUKF-SLAM algorithm are lower than the estimation errors given by the other algorithms.

From Figures 11 and 15, it can be seen that the adaptive weighting coefficient is greater than one at some certain

TABLE 2: Extracted line feature parameters.

Line feature	ρ	θ	Time (step)	Position of UUV in E(m)
L1	13	-0.2618	260~310	(-786.3968, -40.6424)
L2	33	0	320~370	(-949.0549, -47.3198)
L3	39	0	430~500	(-1279.0335, 47.42)
L4	15	-0.0873	570~610	(-1504.7369, 206.5273)
L5	31	-0.0873	640~720	(1449.9067, 483.5776)
L6	35	-0.2618	730~820	(-1278.9245, 649.0867)

TABLE 3: Change law of system noise.

Test	Value of Q
Time-varying noise test	$\text{diag}([9e-6 \ 1e-6 \ 8e-5 \ 3e-5 \ 9e-6 \ 1e-6 \ 8e-6 \ 3e-6]) \quad 1 < k < 320$
	$Q_k = \text{diag}([9e-5 \ 1e-5 \ 8e-3 \ 3e-3 \ 9e-4 \ 1e-4 \ 8e-5 \ 3e-5]) \quad 320 \leq k < 640$
	$\text{diag}([9e-5 \ 1e-5 \ 8e-4 \ 3e-4 \ 9e-4 \ 1e-3 \ 8e-4 \ 3e-4]) \quad 640 \leq k < 820$
Constant noise test	$Q_k = \text{diag}([5e-6 \ 5e-6 \ 5e-4 \ 9e-3 \ 2e-4 \ 2e-4 \ 4e-4 \ 4e-4])$

FIGURE 10: Comparison of position error in the North direction with time-varying noise.

time. The results indicate that once the Sage-Husa UKF-SLAM tends to diverge according to convergence conditions, the adaptive weighting coefficient is introduced to correct covariance and ensure the tracking ability.

(2) *Root-Mean Square Error.* We use the root-mean square error (RMSE) of the position to compare the performance of the various nonlinear filters

$$\text{RMSE}_{\text{pos}} = \sqrt{\frac{1}{T} \sum_{k=1}^T ((x_k - \hat{x}_k)^2 + (y_k - \hat{y}_k)^2)}, \quad (42)$$

where T is the total running step and (x_k, y_k) and $(\hat{x}_k, \hat{y}_k)$ are the true position and estimated position, respectively, of the

FIGURE 11: Adaptive weighting coefficient of AUKF-SLAM algorithm in time-varying noise test.

TABLE 4: Comparison of RMSE.

Algorithm	Time-varying noise test (m)	Constant noise test (m)
AUKF-SLAM	26.1487	28.5021
UKF-SLAM	28.0639	29.4876
EKF-SLAM	38.5579	39.8530

UUV at time k . Table 4 gives the RMSE for each of the tested algorithms. From the table, we can see that the RMSE of the AUKF-SLAM algorithm is the smallest for both time-varying system noise and constant system noise. The RMSE in the case of time-varying noise for the AUKF-SLAM algorithm is smaller than the RMSE found in the constant noise scenario by 2.3534 m. In the time-varying noise test, the RMSE of

FIGURE 12: Comparison of the trajectory estimations with constant noise.

FIGURE 14: Comparison of position error in the North direction with constant noise.

FIGURE 13: Comparison of position error in the East direction with constant noise.

FIGURE 15: Adaptive weighting coefficient of AUKF-SLAM algorithm in constant noise test.

the AUKF-SLAM algorithm is smaller than the RMSE found using the UKF-SLAM algorithm by 1.9152 m; in the constant noise test the AUKF-SLAM algorithm RMSE is smaller by 0.9855 m.

From the above analysis, we can see that the AUKF-SLAM algorithm has good tracking ability and produces a RMSE that is smaller than what the other algorithms are capable of achieving for both time-varying and constant system noise. Also, the AUKF-SLAM algorithm produces a smaller RMSE in the presence of time-varying system noise as compared to when operating in a system with constant noise.

Figure 17 shows a comparison of the line features extracted during the environment and feature measurement of the

embankment, while Figure 18 is a simultaneous localization and mapping based on AUKF. During UUV navigation, the algorithm uses the three ranging sonar to perceive the environment, continuously extracting ρ - θ parameters based on the Hough transform. The model accomplishes navigation based on the AUKF-SLAM algorithm by repeatedly observing available line features to continuously correct the UUV's position and update its existing line features map. The AUKF-SLAM algorithm avoids the error acceleration caused by dead reckoning and ensures that a UUV no longer has to periodically float to the surface for GPS correction; one of the most important conditions for UUV's tasked with long-term missions where they must remain hidden.

FIGURE 16: Heading of the UUV during the field trial.

FIGURE 17: Comparison of line features extracted from the environment and the feature measurement of the embankment.

5. Conclusion

The proposed AUKF-SLAM algorithm adopts an improved Sage-Hausa suboptimal unbiased maximum posterior noise statistical estimator to estimate unknown system noise. The algorithm estimates and corrects the statistical character of noise in real time and decreases estimated error. At the same time, the algorithm judges whether the filter is converging and introduces an adaptive forgetting factor to correct the predicted covariance adjust the Kalman gain, and restrain the divergence of the filter when it diverges, therefore increasing the algorithm's fast tracking ability. The AUKF-SLAM algorithm provides a new method for simultaneous underwater localization and mapping in an unknown environment. Assistant navigation based on the proposed AUKF-SLAM algorithm can help UUV's fulfil missions requiring

FIGURE 18: SLAM diagram based on the AUKF.

marine environment monitoring, marine terrain inspection, and long-term underwater tasks.

Acknowledgments

This research work is supported by the National Natural Science Foundation of China (Grant no. E091002/50979017), the Ph.D. Programs Foundation of Ministry of Education of China and Basic Technology (Grant no. 20092304110008), the Research Operation Item Foundation of Central University (Grant no. HEUCFZ 1026), the Harbin Science and Technology Innovation Talents of Special Fund Project (Outstanding Subject Leaders) (no. 2012RFXXG083), and the New Century Excellent Talents Foundation of Education of China (no. NCET-10-0053).

References

- [1] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics and Automation Magazine*, vol. 13, no. 2, pp. 99–108, 2006.
- [2] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): part II," *IEEE Robotics and Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006.
- [3] R. C. Smith and P. Cheeseman, "On the representation and estimation of spatial uncertainty," *International Journal of Robotics Research*, vol. 5, no. 4, pp. 56–68, 1986.
- [4] S. Thrun, Y. Liu, D. Koller, A. Y. Ng, Z. Ghahramani, and H. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," *International Journal of Robotics Research*, vol. 23, no. 7-8, pp. 693–716, 2004.
- [5] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "Fast-SLAM: a factored solution to the simultaneous localization and mapping problem," in *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI-02), 14th Innovative Applications of Artificial Intelligence Conference (IAAI '02)*, pp. 593–598, August 2002.

- [6] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.
- [7] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.
- [8] R. Van der Merwe, *Sigma-Point Kalman Filters for probabilistic inference in dynamic state-space models [Ph.D. thesis]*, Oregon Health and Science University, 2004.
- [9] K. Xiong, H. Y. Zhang, and C. W. Chen, "Performance evaluation of UKF-based nonlinear filtering," *Automatica*, vol. 42, no. 2, pp. 261–270, 2006.
- [10] Y. Wu, D. Hu, and X. Hu, "Comments on 'Performance evaluation of UKF-based nonlinear filtering,'" *Automatica*, vol. 43, no. 3, pp. 567–568, 2007.
- [11] K. Xiong, H. Y. Zhang, and C. W. Chan, "Author's reply to 'Comments on 'Performance evaluation of UKF-based nonlinear filtering,'" *Automatica*, vol. 43, no. 3, pp. 569–570, 2007.
- [12] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Proceedings of the Signal Processing, Sensor Fusion, and Target Recognition VI*, pp. 182–193, April 1997.
- [13] L. Zhao, X.-X. Wang, M. Sun, J.-C. Ding, and C. Yan, "Adaptive UKF filtering algorithm based on maximum a posterior estimation and exponential weighting," *Acta Automatica Sinica*, vol. 36, no. 7, pp. 1007–1019, 2010.
- [14] D. H. Zhou, Y. G. Xi, and Z. J. Zhang, "Suboptimal fading extended kalman filter for nonlinear systems," *Control and Decision*, no. 5, pp. 1–6, 1990.
- [15] D. H. Zhou, Y. G. Xi, and Z. J. Zhang, "A suboptimal multiple fading extended kalman filter," *Acta Automatica Sinica*, vol. 17, no. 6, pp. 689–695, 1991.
- [16] D. Ribas, P. Ridao, J. D. Tardós, and J. Neira, "Underwater SLAM in man-made structured environments," *Journal of Field Robotics*, vol. 25, no. 11-12, pp. 898–921, 2008.
- [17] J. D. Tardós, J. Neira, P. M. Newman, and J. J. Leonard, "Robust mapping and localization in indoor environments using sonar data," *International Journal of Robotics Research*, vol. 21, no. 4, pp. 311–330, 2002.
- [18] T. Pavlidis, *Algorithms for Graphics and Image Processing*, vol. 877 of *Lecture Notes in Mathematics*, Computer Science Press, Rockville, Md, USA, 1982.
- [19] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [20] J. Illingworth and J. Kittler, "A survey of the hough transform," *Computer Vision, Graphics and Image Processing*, vol. 44, no. 1, pp. 87–116, 1988.
- [21] J. Ji, G. Chen, and L. Sun, "A novel Hough transform method for line detection by enhancing accumulator array," *Pattern Recognition Letters*, vol. 32, no. 11, pp. 1503–1510, 2011.
- [22] H. J. WANG, J. WANG, Q. U. L. P, and L. I. U. Z. Y, "Sea environment feature extraction based on fuzzy adaptive Hough transform," *Chinese Journal of Scientific Instrument*, vol. 34, no. 1, pp. 32–37, 2013.
- [23] J. Cha, R. H. Cofer, and S. P. Kozaitis, "Extended Hough transform for linear feature detection," *Pattern Recognition*, vol. 39, no. 6, pp. 1034–1043, 2006.
- [24] X. Zhang, H. J. Wang, J. J. Zhou, X. Q. Bian, and L. Xiong, "A method study on SFEKF-SLAM for a UUV's structural environment," *Journal of Harbin Engineering University*, vol. 33, no. 8, pp. 1016–1021, 2011.
- [25] L. Xiong, *Research and design on the underwater SLAM in man-made structures environments based on multi single-beam sonars [M.S. thesis]*, Harbin Engineering University, 2011.
- [26] T. Bailey, *Mobile robot localisation and mapping in extensive outdoor environments [Ph.D. thesis]*, Sydney University, Sydney, Australia, 2002.
- [27] J. Neira and J. Tardos, "Robust and feasible data association for simultaneous localization and map building," in *Proceedings of the IEEE conference on Robotic and Automation Workshop W4*, San Francisco, Calif, USA, 2000.