Underwater Target Bearings-Only Trajectory Tracking Based on Long Short-Term Memory Neural Network

Zhe Chen[1,2], Shenqi Zhao[1], and Mingsong Chen[1](✉)

[1] School of Information and Communication, Guilin University of Electronic Technology, Guilin 541004, China
cms@guet.edu.cn

[2] Cognitive Radio and Information Processing Key Laboratory Authorized By China's Ministry of Education Foundation, Guilin University of Electronic Technology, Guilin 541004, China

Abstract. Marine acoustic passive target tracking has received much attention from researchers as an important part of ocean exploration. This issue becomes challenging in complex marine environments when the target motion model changes abruptly. The primary objective of underwater passive target tracking is to acquire the bearing parameters of the target from the observation station. In this paper, we apply Long Short-Term Memory (LSTM) neural network and dual observation stations system to the problem of underwater target bearings-only trajectory tracking. Results of computer simulation prove the stability of the proposed algorithm in the problem of bearings-only target tracking when only angles are measured, and effectively reduce the root-mean-square error. Therefore, this study provides a new technical solution for underwater bearings-only maneuvering target tracking.

Keywords: Target Tracking · Bearings-only · Long Short-Term Memory Neural Network · Dual Observation Stations

1 Introduction

Being one of the vital objects of marine study, underwater target tracking is a common concern of researchers in military and civilian fields. In sonar applications, the tracking system can be categorized into two types: active and passive [1]. The passive sonar sensors, which provide bearing measurements, require minimal power and can remain hidden without risking being detected. However, due to the range unobservability and system nonlinearity, there are limitations to bearings-only target tracking applications. To overcome the unobservability problem, a dual stationary observation stations system can be deployed, which enables the acquisition of two bearings of the target simultaneously [2].

Extensive research has been conducted over the past few decades on underwater target tracking using bearing measurements, resulting in the proposal and application

J. Wang (Ed.): MobiMedia 2023, LNICST 576, pp. 76–85, 2025.
https://doi.org/10.1007/978-3-031-60347-1_8

of various algorithms in engineering practice. The Kalman Filter (KF), which belongs to the Bayesian filtering domain, was one of the earliest solutions and was proposed by Kalman R. E. et al. [3]. In the KF algorithm, there is a prediction phase and an update phase, which enables refinement of the state estimate by combining previous information and prediction information [4]. However, research has consistently shown that in complex underwater environments, KF may produce a divergence phenomenon due to the extremely uncertain and nonlinear behavior of the measurement model [5]. To cope with the nonlinearity of the observed model, the Extended Kalman Filter (EKF) algorithm was proposed based upon the KF algorithm [6]. In the EKF, the first Taylor expansion is used to revert the nonlinear models to linear [7]. Cundong et al. applied the EKF to the research of image path estimation and demonstrated that the EKF has better tracking accuracy and stability [8]. Whereas, a major problem with this kind of algorithm is that in strongly nonlinear systems, it may fail to converge due to the omission of higher-order terms in the measurement function [9]. To address the limitations of the EKF, Julier et al. proposed the Unscented Kalman Filter (UKF) algorithm, which utilizes the Unscented Transform (UT) rule in the conventional model of the KF algorithm [10]. Previous studies have shown that the UKF algorithm effectively reduces errors and is useful in solving nonlinear filter problems [11]. Despite this, the UKF is known to have negative definite covariance in high-dimensional systems, which leads to divergent results [12]. For the problem of high-dimensional systems, Arasaratnam I et al. designed a more accurate nonlinear filter called the Cubature Kalman Filter (CKF) algorithm [13], which employs the cubature rule to estimate the mean and variance of the approximation function at the sampling points, thereby reducing linearization errors. At the same time, the CKF algorithm requires a strict target model, and it performs poorly when the motion model is uncertain [14].

In this paper, we propose a bearings-only target trajectory tracking method based on Long Short-Term Memory (LSTM) neural network and dual observation stations system in the context of abrupt changes of underwater target motion model and poor accuracy of conventional filtering algorithms. Firstly, a trajectory segment containing constant velocity (CV) model and coordinate turn (CT) model is input to the network for training. Then, the network predicts the target motion parameters at the next moment based on its memory. The current motion model of the target is determined by combining the sensor measurement data from the observation stations with the LSTM neural network prediction data. Finally, the target trajectory tracking is performed recursively. This work using the root mean square error (RMSE) [15] as the evaluation criterion. The simulation results demonstrate that the LSTM neural network target trajectory prediction accuracy is significantly better than the traditional Interacting Multiple Model (IMM) filtering algorithm [16].

2 Dual Observation Stations Bearings-Only Tracking Model

2.1 Target Tracking Model

The geometric relationship between the target and the observation stations in a two-dimensional Cartesian coordinate system is shown in Fig. 1. Observation station0 is at the origin of the coordinates and observation station1 is $(L,0)$. The initial vector of the target vector is $\left[x_0\ v_x\ y_0\ v_y\right]^T$, so the target starts moving at the initial position (x_0, y_0) along heading K with initial velocity (v_x, v_y). At the initial position A, the azimuth of the two stations can be expressed as (β_{0A}, β_{1A}), and after time k the target moves to point B, where the azimuth of the two stations is (β_{0B}, β_{1B}).

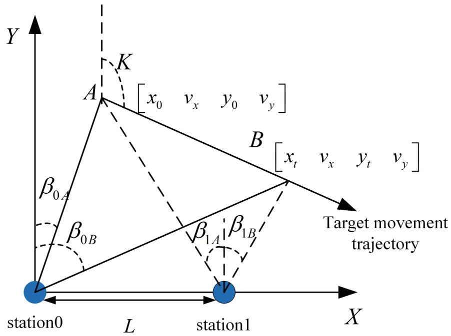


Fig. 1. Target bearings-only trajectory tracking model for dual observation stations.

2.2 State Equation

The state equation of the target is determined by the motion mode of the target, and its expression is:

$$X(k+1) = F \cdot X(k) + W(k) \tag{1}$$

where $X(k) = \left[x_k\ v_k\ y_k\ v_k\right]^T$ is the position and velocity vector of the target at time k, F is the state transition matrix, $W(k)$ denotes the target motion process noise which is a Gaussian white noise with zero mean and covariance Q. In the state transition matrix, the current motion model is determined by internal parameters. The state equations of the CV model and the CT model are shown in formula (2) and formula (3), respectively:

$$X(k+1) = F_{CV} \cdot X(k) + W(k) \tag{2}$$

$$X(k+1) = F_{CT} \cdot X(k) + W(k) \tag{3}$$

where $F_{CV} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$, $F_{CT} = \begin{bmatrix} 1 & \frac{\sin(\omega T)}{\omega} & 0 & \frac{\cos(\omega T)-1}{\omega} \\ 0 & \cos(\omega T) & 0 & -\sin(\omega T) \\ 0 & \frac{1-\cos(\omega T)}{\omega} & 1 & \frac{\sin(\omega T)}{\omega} \\ 0 & \sin(\omega T) & 0 & \cos(\omega T) \end{bmatrix}$,T is the sampling interval and ω is the angular velocity of target.

2.3 Measurement Equation

The measurement equation of the observation stations is described as:

$$Z(k) = h(k) + V(k) \tag{4}$$

where $Z(k)$ is the measurement information of the target at time k, $h(k)$ is the measurement function, and $V(k)$ is the measurement noise, which follows a Gaussian white noise with zero mean and covariance R. The dual observation stations system measurement function can be written as:

$$h(k) = \begin{bmatrix} \arctan\left(\frac{x_k}{y_k}\right) \\ \arctan\left(\frac{x_k - L}{y_k}\right) \end{bmatrix} \tag{5}$$

where (x_k, y_k) is the position of the target at time k and L is the distance between two observation stations. Here we assume that $W(k)$ and $V(k)$ are independent of each other.

3 LSTM Neural Network

3.1 LSTM Neural Network Model

The LSTM neural network is a temporal recurrent network in which the input information and the memory information of the network model are weighted during the computation, and the weight of each part is obtained by training the network. Figure 2 shows the model of the LSTM neural network, whose core idea is to process the input information through a structure known as a ‘gate’. Common ‘gates’ include forgetting gate, input gate, and output gate, which are selective and determine which information can pass through. The function of the forgetting gate is to remove unimportant information from the past; the function of the input gate is to feed the current moment value into the network and select it; and the function of the output gate is to output the result of the network calculation, which can be used as the output of the network at the current moment or as the input to the network at the next moment.

3.2 LSTM Neural Network Structure

In Fig. 2, the first ’gate’ (blue dashed part) is the forgetting gate, which takes as input the network state $h(t-1)$ at the previous time and the input information $x(t)$ at the current time. The Sigmoid function outputs the forgetting gate coefficients $f(t)$, which C_{t-1} represent the network memory of the entire process. The expression $f(t)$ is given by Eq. (6), where $f(t)$ is the forgetting gate weight parameter, obtained through internal learning, and b_f is the bias term.

$$f(t) = \sigma\left(W_f.\left[h(t-1)\ x(t)\right] + b_f\right) \tag{6}$$

The second ’gate’ is the input gate (black dashed part), which contains two parts, the first part is similar to the forgetting gate, but utilizes different internal parameters, it

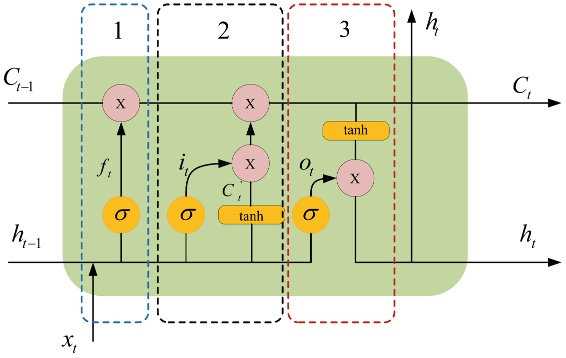


Fig. 2. LSTM neural network model.

is characterized by a weight parameter denoted as W_i, a bias term as b_i, and an output as $i(t)$. Meanwhile, the second part involves an activation function known as the tanh function, which is associated with a weight parameter W_c and a bias term b_c. This part of the input gate store short-term memory of the current time step, represented as $C_t\prime$. The current moment part information that stored in the permanent memory is rep-resented by $i(t) \cdot C_t\prime$. The $i(t)$, $C'(t)$ expressions are:

$$i(t) = \sigma\big(W_i\big[h(t-1)\ x(t)\big] + b_i\big) \tag{7}$$

$$C\prime(t) = \tanh\big(W_C \cdot \big[h(t-1)\ x(t)\big] + b_c\big) \tag{8}$$

The long-term memory update of an LSTM neural network depends on the relevant information to be retained at the current time step and the contents of the memory from the previous time step. The expression for the long-term memory is denoted by $C(t)$.

$$C(t) = f(t) \cdot C(t-1) + i(t) \cdot C\prime(t) \tag{9}$$

The third 'gate' is the output gate (red dashed part), which filter the information by the internal weight parameter W_o and the bias term b_o, generating an output function denoted as $o(t)$. The output function is then multiplied by the long-term memory $C(t)$ processed by the activation function tanh to produce the current moment output $h(t)$. The expressions for $o(t)$ and $h(t)$ are:

$$o(t) = \sigma\big(W_o \cdot \big[h(t-1)\ x(t)\big] + b_o\big) \tag{10}$$

$$h(t) = o(t) \cdot \tanh(C(t)) \tag{11}$$

The three components discussed above represent the minimum unit of the LSTM neural network. Depending on the nature of the problem and the characteristics of the input sequence, the network can be expanded horizontally and vertically. The weight parameters of the network are learned internally during the training process. By adjusting the neural network parameters, a complete neural network model can be constructed, which is capable of retaining information over long periods, adapting to changing inputs, and making accurate predictions after sufficient training.

4 LSTM Neural Network Training

The motion models of underwater maneuvering targets are subject to uncertainties, and their models can be summarized as CV models and CT models. In this paper, we use Matlab software to generate a simulated trajectory that serves as the input for training the LSTM neural network. The training process consists of the following steps:

Step 1: Generate motion trajectory in a two-dimensional coordinate system with the starting point of the target motion trajectory is (500,500), 0-60S is constant velocity motion, 61-150S is right turning motion, 151-220S is constant velocity motion, 221-310S is left turning motion, 311-390S is constant velocity motion. The motion parameters for each phase were set according to Table 1, and the resulting trajectory is illustrated in Fig. 3.

Table 1. Target trajectory parameters.

time/s	motion model	motion parameter
0 ~ 60	CV	v_x =2m/s,$v_y = 0m/s$
61 ~ 150	CT	$\omega = -\pi/60$
151 ~ 220	CV	$v_x = 0.1185m/s$, $v_y = 2.8560m/s$
221 ~ 310	CT	$\omega = \pi/60$
311 ~ 390	CV	$v_x = 2.2471m/s$, $v_y = -0.0199m/s$

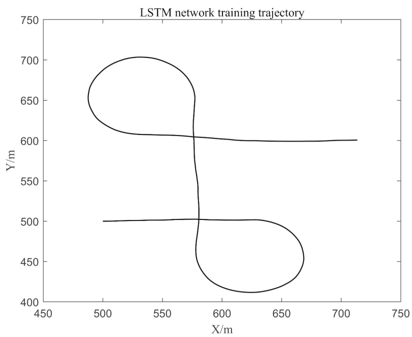


Fig. 3. Trajectory for LSTM neural network training.

Step 2: Train the LSTM neural network.

The LSTM neural network consists of input layer, LSTM layer, dropout layer, fully connected layer and regression prediction layer. The input data in the input layer contains two features, which are X-axis and Y-axis coordinates. The LSTM unit network is expanded horizontally to use the data from the first 20 moments to predict the 21st point, as shown in Fig. 4 is LSTM training model. In the LSTM layer, set the number of hidden units to 128 and the output mode to last. To prevent overfitting, we set the dropout parameter to 0.2 in the dropout layer. The output features of the fully connected layer include the predicted X-axis and Y-axis data, and then analyze the network and export. Other parameters for defining the network is trained using 100 epochs, selected adam optimizer with an initial learning rate of 0.005.

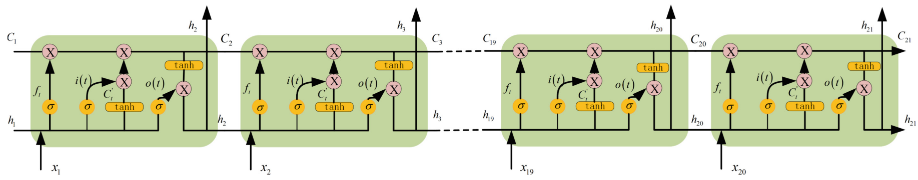


Fig. 4. LSTM neural network training model.

Step 3: Predict and update.

To obtain stable training parameters, the network is trained multiple times using the actual motion trajectory of the target as input.

Step 4: Performance evaluation criteria.

The root mean square position error between the true and predicted trajectory of the target motion at each moment is used as the performance evaluation criterion of the LSTM neural network. The expression for RMSE is:

$$\mathrm{RMSE}(X_0, X_i) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(X_i - X_0)^2} \quad (12)$$

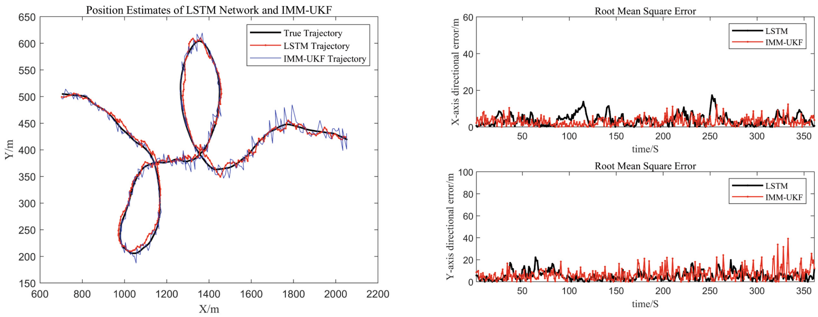
where X_0 is the actual position of the target, X_i is the output value of the LSTM neural network, and N is the number of Monte Carlo simulations.

5 Results and Discussion

The simulation results of the state estimation and position error of the LSTM neural network in bearing-only underwater target trajectory tracking are briefly discussed in this section. To conduct a comparison study of the tracking trajectory, a segment of randomly generated trajectory was input to the LSTM neural network, and the measurement noise of the observation stations was gradually increased, allowing for a comparative study of the tracking trajectory.

5.1 Simulation Experiment 1

The test trajectory commenced at position (500,500) with an initial velocity of (2,0) and lasts for 380 s. The motion model used for simulation comprises both the CV and CT models. During the simulation, the process noise intensity is set to 0.5, and the covariance of the measurement noise is 0.01 rad (about 0.573°). The errors in azimuth were converted into new coordinates for the target trajectory and input into the LSTM neural network. Thirty Monte Carlo simulations were performed under this condition. Figure 5(a) compares the tracking trajectories of the LSTM neural network and the IMM-UKF algorithm, both of which effectively track the maneuvering target. The LSTM algorithm shows less fluctuation in trajectory than the IMM-UKF algorithm. Figure 5(b) shows the root mean square error of the LSTM neural network algorithm and the IMM-UKF algorithm in the X-axis and Y-axis directions, respectively.



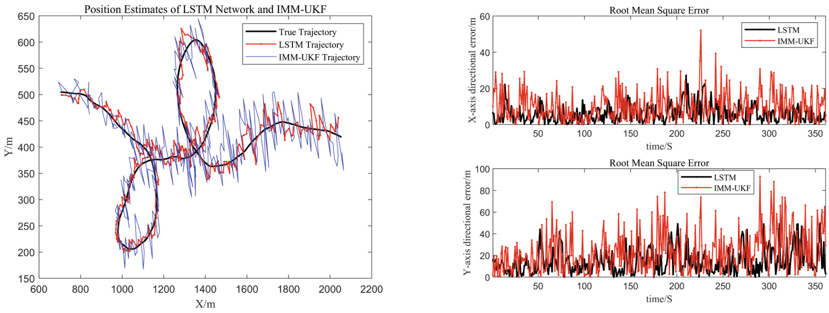
(a) Target movement trajectory (b) Root mean square error

Fig. 5. The target trajectories and position errors estimated by LSTM neural network and IMM-UKF algorithm (R = 0.01rad).

5.2 Simulation Experiment 2

When the covariance of the measurement noise is increased to 0.1 rad for both stations, while keeping other initial conditions unchanged, the tracking accuracy of the LSTM neural network and IMM-UKF are observed. As shown in Fig. 6(a), the predicted trajectory of the LSTM neural network can still track the actual trajectory accurately, while the result of the IMM-UKF algorithm fluctuates significantly more. In Fig. 6(b), it can be seen that the root mean square error of the LSTM algorithm increases slowly in the X-axis and Y-axis directions, showing better tracking performance. In contrast, the increase in measurement noise has a significant impact on the root mean square error of the IMM-UKF algorithm.

Table 2 displays the root mean square error data of the X-axis and Y-axis under different measurement noise conditions. The results indicate that the LSTM neural network outperforms the IMM-UKF algorithm, demonstrating significantly lower variations in the root mean square error as the measurement noise increases.



(a) Target movement trajectory (b) Root mean square error

Fig. 6. The target trajectories and position errors estimated by LSTM neural network and IMM-UKF algorithm (R = 0.1rad).

Table 2. RMSE of the target position estimated by the two methods when R = 0.01and 0.1rad.

Measurement noise(rad)	LSTM		IMM-UKF	
	X-axis/m	Y-axis/m	X-axis/m	Y-axis/m
0.01	3.6730	5.5457	3.0345	6.5580
0.1	6.5730	13.2359	10.2078	22.4999

6 Conclusion

In this study, a bearing-only underwater target trajectory tracking algorithm based on LSTM neural network and dual observation stations system is presented for underwater maneuvering target model abrupt changes and filter scattering problems. By training on a trajectory segment that containing multiple models, the algorithm can memorize the features of the motion model and improve its response to abrupt changes in the target motion. This method can be used in scenarios such as fish detection, ship navigation, and underwater tracking attack systems. Simulation results show that the LSTM neural network provides greater stability and tracking performance, with tracking accuracy comparable to traditional filtering algorithms in ideal marine environments. Furthermore, in the presence of increasing observation noise, the LSTM neural network algorithm outperforms the traditional filtering algorithm in terms of accuracy and stability. However, the results of all algorithms show varying degrees of decay in a noisy marine scenario. As a result, achieving accurate state estimates in complex marine environments has wide space for development.

Acknowledgement. This research was supported by the Special Program of Guangxi Science and Technology Base and Talents under Grant (No. AD21220098) and Innovation Project of GUET Graduate Education (No. 2023YCXS030, No. 2023YCXS021).

References

1. Li Y, Zhao Z.: Passive tracking of underwater targets using dual observation stations. In: 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), pp.867–872 (2019)
2. Li, X., Willett, P., Baum, M., et al.: PMHT approach for underwater bearing-only multisensor–multitarget tracking in clutter. IEEE J. Oceanic Eng. **41**(4), 831–839 (2016)
3. Kalman, R.E., Bucy, R.S.: New results in linear filtering and prediction theory. J. Basic Eng. **83**(1), 95–108 (1961)
4. Ghansah, B., Benuwa, B.B., Essel, D.D., et al.: A review of non-linear Kalman filtering for target tracking. Int. J. Data Anal. (IJDA) **3**(1), 1–25 (2022)
5. Zhang, M., Li, K., Hu, B., et al.: Comparison of Kalman filters for inertial integrated navigation. Sensors **19**(6), 1426 (2019)
6. Julier, S.J., Uhlmann, J.K.: New extension of the Kalman filter to nonlinear systems. Signal Proc. Sens. Fusion Target Recogn. **19**, 182–193 (1997)
7. Ali, W., Li, Y., Raja, M.A.Z., et al.: Generalized pseudo Bayesian algorithms for tracking of multiple model underwater maneuvering target. Appl. Acoust. **166**, 107345 (2020)
8. Cundong T, Zhiping W.: Research and simulation of path estimation based on extended kalman filtering algorithm in video image. In: 2021 4th International Conference on Information Systems and Computer Aided Education (ICISCAE), pp. 482–484 (2021)
9. Garcia, R.V., Pardal, P.C.P.M., Kuga, H.K., et al.: Nonlinear filtering for sequential spacecraft attitude estimation with real data: Cubature Kalman Filter, Unscented Kalman Filter and Extended Kalman Filter. Adv. Space Res. **63**(2), 1038–1050 (2019)
10. Julier, S., Uhlmann, J., Durrant-Whyte, H.F.: A new method for the nonlinear transformation of means and covariances in filters and estimators. IEEE Trans. Autom. Control **45**(3), 477–482 (2000)
11. Wu R.: Performance of UKF filtering algorithm in nonlinear target tracking environment based on big data technologies. In: 2021 International Conference on Application of Intelligent Systems in Multi-modal Information Analytics (MIMA), pp. 717–722 (2021)
12. Li, X., Zhao, C., Yu, J., et al.: Underwater bearing-only and bearing-Doppler target tracking based on square root unscented Kalman filter. Entropy **21**(8), 740 (2019)
13. Arasaratnam, I., Haykin, S.: Cubature kalman filters. IEEE Trans. Autom. Control **54**(6), 1254–1269 (2009)
14. Shen, C., Xiong, Y., Zhao, D., et al.: Multi-rate strong tracking square-root cubature Kalman filter for MEMS-INS/GPS/polarization compass integrated navigation system. Mech. Syst. Signal Process. **163**, 108146 (2022)
15. Ali, W., Khan, W.U., Raja, M.A.Z., et al.: Design of nonlinear autoregressive exogenous model based intelligence computing for efficient state estimation of underwater passive target. Entropy **23**(5), 550 (2021)
16. Qi, J., Sun, K., Wang, J., et al.: Dynamic state estimation for multi-machine power system by unscented Kalman filter with enhanced numerical stability. IEEE Trans. Smart Grid **9**(2), 1184–1196 (2016)