Improved IMM Algorithm based on RNNs

Lichuan Deng, Da Li* and Ruifang Li
School of Information Engineering, Wuhan University of Technology, Wuhan 430070, Hubei, China
*leyrain@whut.edu.cn

Abstract. The Interactive Multi-Model (IMM) algorithm uses multiple motion models to simultaneously track the target, which effectively solves the problem of model mismatch when a single model tracks the maneuvering target, and is widely used in maneuvering target tracking tasks. However, the Interactive Multi-Model recognition motion model is not accurate enough, and there is a certain delay in the maneuver recognition of the target, which leads to a reduction in tracking accuracy. To solve this problem, considering that deep neural networks are very good at processing classification tasks, we introduce it into target tracking tasks, combining the respective of deep neural networks and traditional tracking filtering methods for maneuvering target tracking. We use the Recurrent Neural Networks to identify the motion model of the target and propose an improved LSTM-IMM model algorithm based on the interactive multi-model algorithm. Finally, we compare the traditional interactive multi-model algorithm and verify the algorithm using Monte Carlo simulation. The results show that the proposed algorithm has improved the recognition accuracy and recognition speed of the model, and the tracking accuracy has been improved.

1. Introduction
In the process of tracking, using a simple single motion model to track the target can easily lead to large tracking errors or missing targets due to motion model mismatch. The solution to this problem is to use an IMM algorithm [1][2][3][4]. The IMM model uses multiple motion models to track the target at the same time, which overcomes the problem of poor tracking performance of the single model. However, the shortcoming of the IMM model algorithm is that the recognition motion model is not accurate enough and there is a delay in the maneuver recognition of the target.

Deep learning has the ability to extract abstract features from complex data[5]. Recurrent Neural Networks (RNNs) perform well on processing sequence problems, which is widely used in machine translation, weather forecasting, video motion recognition and so on. The target tracking problem is actually a sequence problem. The state vector of the target has a causal relationship in the time series. So RNNs can be used to handle target tracking tasks. Based on the above points, we propose to use RNNs to improve the IMM algorithm. An LSTM-IMM model is proposed. The LSTM part is used to identify the target motion model of the current state, and the IMM part performs filter tracking. The simulation results show that the newly proposed LSTM-IMM model recognizes the target model more quickly and accurately, and effectively reduces the estimation error.

The paper is organized as follows: Section 2 introduced related work, The LSTM-IMM model is introduced in section 3, We compare the LSTM-IMM model with the traditional IMM model in section 4, and we have summarized our work in section 5.
2. Related Work
With the improvement of computer computing power, deep learning began to show its own advantages. More and more deep learning methods are beginning to replace traditional algorithms or use deep learning combined with traditional methods to handle tasks[6][7][8][9][10].

Zhu H et al.[6] proposed the NACT-IMM algorithm, using a BP neural network to extract features from the tracking trajectory of the target, which is used to predict the turning rate of the CT model at the current time, has achieved good results. The model has certain advantages for target tracking with frequent turning and high turning angle. Bækkegaard S et al.[7] uses Recurrent Neural Networks to extract vessel data from the Automatic Identification System (AIS), classifies the target vessels, and uses the results of the classification to assist in subsequent trajectory tracking. The paper demonstrates that RNNs have the ability to classify trajectories based on kinematic data. In response to the complex and variable environment of ground wave radar detection and the maneuvering of ships, Zhang L et al.[8] uses neural networks to learn the motion laws of ships and uses them to make fusion decisions between target ship attributes and motion models. Sun Y et al.[9] proposed to use the deep learning method to mine the tracking data of each sensor, and introduce the auxiliary knowledge to establish the correlation between the motion feature and the specific target to improve the accuracy of the target tracking, which achieved good results. Zhang Z et al.[10] directly used the modified LSTM network FF-LSTM to predict the target trajectory of the aircraft flight, and achieved good results.

3. LSTM-IMM algorithm
The IMM algorithm is a popular target tracking algorithm. It is inaccurate for the estimation of the motion model probability and the delay is large for the IMM model. Because the IMM algorithm is inaccurate in estimating the probability of the motion model, and the recognition delay is large. We use the LSTM neural network to improve this problem and propose the LSTM-IMM algorithm. The LSTM network is used to predict the motion model probability and input the results into the IMM model for interaction to improve the accuracy. Our algorithm structure is shown in Figure 1.

![Figure 1. The structure picture of LSTM-IMM](image)

Measurement data is sent to the IMM model, In the input interaction stage, the mixed state estimation $\hat{Y}_0$ and $\hat{P}_0$ of each model is calculated according to the predicted probability of the model and the state estimation of the target. The different models are then filtered in parallel. In the interactive output stage, the weights of the respective models are obtained according to the probabilities of the respective models predicted by the LSTM. The input of the LSTM neural network is the result of the interactive output, and the output is the current probability of each model.

Assume that the target has $n$ motion models, corresponding to $n$ motion models. The steps of algorithm filtering are given below:

**Step 1:** Input model interaction

Calculate the mixed state estimate and mixed covariance estimate of each model:

$$\hat{Y}_0(k-1|k-1) = \sum_{i=1}^{n} c_i(k-1|k-1) \cdot \hat{Y}_i(k-1|k-1)$$

(1)
\[ P_{0j}(k-1|k-1) = \sum_{i=1}^{n} c_{ij}(k-1|k-1) \cdot \{ [\hat{Y}_i(k-1|k-1) - \hat{Y}_{0j}(k-1|k-1)]^T + P_i(k-1|k-1) \} \] (2)

Where \( j = 1...n \), \( c_{ij}(k-1|k-1) = \sum_{i=1}^{n} p_{ij} e_i(k-1)/\bar{e}_j \), \( \bar{e}_j = \sum_{i=1}^{n} p_{ij} e_i(k-1) \). \( p_{ij} \) is the transition probability of the motion models \( i \) to \( j \).

**Step 2:** Kalman filtering is performed simultaneously for each motion model, and the filtered output results \( \hat{Y}_j(k|k) \) and \( P_j(k|k) \) of each model are obtained.

**Step 3:** Use the trained LSTM model to predict the model probability and obtain the probability corresponding to each model \( c_j(k) \), \( j = 1...n \).

**Step 4:** Output interaction, weighted fusion of the output \( \hat{Y}_j(k|k) \) and \( P_j(k|k) \) of each model based on model probability, and calculate the total state output \( \hat{Y}(k|k) \) and \( P(k|k) \).

\[ \hat{Y}(k|k) = \sum_{j=1}^{n} \hat{Y}_j(k|k) c_j(k) \] (3)

\[ P(k|k) = \sum_{j=1}^{n} c_j(k) \left\{ P_j(k|k) + [\hat{Y}_j(k|k) - \hat{Y}(k|k)] \cdot [\hat{Y}_j(k|k) - \hat{Y}(k|k)]^T \right\} \] (4)

**Step 5:** Repeat step 1 to step 4 for recursive filtering to achieve maneuvering target tracking.

4. **Experiment and simulation results**

4.1. **Training LSTM neural network**

We use LSTM to predict the trajectory of the tracked target and output the probability of each motion model at the current time. According to the characteristics of the target tracking data, we find that directly using the state vector value of the target as the input feature will make the training more difficult. Therefore, the state features we choose are the amount of change in displacement in the x-coordinate direction, the amount of change in displacement in the y-coordinate direction, the amount of change in velocity in the x-coordinate direction, and the amount of change in velocity in the y-coordinate direction.

4.2. **LSTM-IMM algorithm simulation**

The simulation experiment uses three motion models for the experiment, namely CV, CT left turn and CT right turn model. The sampling rate is 1 second. The movement process of the target is: constant velocity motion between 0-10 seconds, left-turning between 10-20 seconds, constant velocity motion between 20-30 seconds, and right-turn between 30-40 seconds.

The number of Monte Carlo cycles is 400, and the RMSE is used using the error measure. The transition probabilities between the three motion models are shown in Equation (5).

\[ P = \begin{bmatrix} 0.9 & 0.05 & 0.05 \\ 0.1 & 0.8 & 0.1 \\ 0.1 & 0.1 & 0.8 \end{bmatrix} \] (5)
For the IMM algorithm, the initial probability of the model is \( u = [0.4 \ 0.3 \ 0.3] \), since the LSTM prediction model probability requires 10 sampling points, it means that the first 9 sampling points cannot be predicted. We set the motion model probabilities of the first 9 sampling points to \([0.4 \ 0.3 \ 0.3]\). The noise is 30m in both the x-axis and y-axis directions.

We first compare the performance of LSTM-IMM algorithm and IMM algorithm for motion model recognition. The experimental simulation results are as follows:

![Figure 2](image1.png)

**Figure 2.** Motion model probability map of LSTM-IMM algorithm (sampling rate = 1s)

![Figure 3](image2.png)

**Figure 3.** Motion model probability estimation graph of IMM algorithm (sampling rate = 1s)

Figure 2 and Figure 3 are model distributions of LSTM-IMM algorithm and IMM recognition. It can be seen from the figure that the LSTM-IMM algorithm recognition model is more accurate and faster. Specifically, the target maneuver occurred for the first time at 20s. At this time, the LSTM-IMM algorithm has recognized the target maneuver and began to adjust the probability of the model. At the same time, the IMM algorithm did not recognize the target maneuver and did not adjust the model probability. In the following several target maneuver recognition, the IMM performance is better than LSTM-IMM. Figure 4 shows the location tracking RMSE of the LSTM-IMM and IMM algorithms. Figure 4(a) and Figure 4(b) are the RMSE in the x-axis and y-axis directions, respectively. It can be seen that the error of the LSTM-IMM is smaller than that of the IMM, which is consistent with the higher accuracy of the LSTM-IMM recognition motion model. And because the LSTM-IMM model recognizes the target's maneuver quickly, the error does not fluctuate significantly during the tracking process. Correspondingly, the tracking error of the IMM algorithm fluctuate significantly during the tracking process.
Then we change the sampling rate to 0.2 seconds, and the obtained LSTM-IMM algorithm model identification distribution and IMM algorithm model identification distribution are shown in Figure 5 and Figure 6 respectively. We found that the model recognition accuracy of the IMM algorithm decreased a lot after the sampling rate increased, while the LSTM-IMM model remained basically unchanged.

Figure 4. Position Tracking RMSE for LSTM-IMM and IMM Algorithms (sampling rate = 1s)

Figure 5. Motion model probability estimation graph of LSTM-IMM algorithm (sampling rate = 0.2s)

Figure 6. Motion model probability estimation graph of IMM algorithm (sampling rate = 0.2s)
Figure 7. Position Tracking RMSE for LSTM-IMM and IMM Algorithms (sampling rate = 0.2s)

Figure 7 is the tracking position RMSE of the LSTM-IMM model and the IMM model, where Figure 7(a) is the position RMSE in the x-axis direction and Figure 7(b) is the RMSE in the y-axis direction. It can be seen that the RMSE of the LSTM-IMM algorithm is lower than the IMM algorithm. Moreover, within 20-40 seconds and 60 to 80 seconds, the IMM algorithm has a very large estimation error for the motion model, and the RMSE in this time period is also large, reflecting that there is a positive correlation between the accuracy of the motion model recognition and the tracking error. That is, the higher the accuracy of model recognition, the higher the tracking accuracy of the algorithm. The LSTM-IMM algorithm effectively solves the problem that the IMM model is not accurate in estimating the motion model.

5. Conclusion
In view of the problem that the IMM recognition motion model is not accurate enough and there is a certain lag in the recognition process, we propose an improved LSTM-IMM model, which uses the deep learning method to identify the motion model, improve the recognition accuracy and speed of the motion model, as well as increase the target tracking accuracy. The experimental results show that the proposed model has improved the recognition accuracy and recognition speed of the motion model, and the target tracking accuracy is also improved. We only simulate several simple motion models. In the future, we will experiment on more motion models as well as the complex ones.

References
[1] Li D, Zou X, Lou P, et al. Iterative Unbiased Conversion Measurement Kalman Filter with Interactive Multi-model Algorithm for Target Tracking[C]// Asian Simulation Conference. Springer, Singapore, 2018.
[2] Han B, Huang H, Lei L, et al. An Improved IMM Algorithm Based on STSRCKF for Maneuvering Target Tracking[J]. IEEE Access, 2019, 7: 57795-57804.
[3] Bekhtaoui Z, Meche A, Dahmani M, et al. Maneuvering target tracking using q-learning based Kalman filter[C]//2017 5th International Conference on Electrical Engineering-Boumerdes (ICEE-B). IEEE, 2017: 1-5.
[4] Yoo K, Chun J, Shin J. Target tracking and classification for missile using interacting multiple model (IMM)[C]/2018 International Conference on Radar (RADAR). IEEE, 2018: 1-6.
[5] Wang L, Tang J, Liao Q. A Study on Radar Target Detection Based on Deep Neural Networks[J]. IEEE Sensors Letters, 2019, 3(3): 1-4.

[6] Zhu H, Xiong W, Cui Y, et al. Research on IMM Algorithm of New Adaptive Turn Model[J]. Computer Engineering and Applications, 2019,55(17): 252-258.

[7] Bækkegaard S, Bliksenkrone-Møller J, Larsen J J, et al. Target Classification Using Kinematic Data and a Recurrent Neural Network[C]/2018 19th International Radar Symposium (IRS). IEEE, 2018: 1-10.

[8] Zhang L, Lui X, Jiang Y, et al. Research on a specific target tracking method based on arc detection with HF surface wave radar[J]. Marine Sciences, 2016,40(06):133-138

[9] Sun Y, Dai Y, Ji Y, et al. New ideas for HFSWR maritime target tracking[J]. Marine Sciences, 2017,41(06):144-149.

[10] Zhang Z, Yang R, Zhang B, et al. Air Combat Trajectory Predict Simulation for Enemy Target Approach[J]. Journal of Air Force Engineering University (Natural Science Edition),2018,19 (2):33-37.