

RESEARCH

Open Access



An improved dynamic programming tracking-before-detection algorithm based on LSTM network

Fei Song^{1,2} , Yong Li^{1*}, Wei Cheng¹ and Limeng Dong¹

*Correspondence:
alexsongfei096@mail.nwpu.
edu.cn

¹ School of Electronic
Information, Northwestern
Polytechnical University,
Xi'an 710072, China

² School of Electronic
Engineering, Xi'an Aeronautical
Institute, Xi'an 710077, China

Abstract

The detection and tracking of small and weak maneuvering radar targets in complex electromagnetic environments is still a difficult problem to effectively solve. To address this problem, this paper proposes a dynamic programming tracking-before-detection method based on a long short-term memory (LSTM) network (LSTM-DP-TBD). With the predicted target motion state provided by the LSTM network, the state transition range of the traditional DP-TBD algorithm can be updated in real time, and the detection and tracking effect achieved for maneuvering small and weak targets is also improved. Utilizing the LSTM network to model the moving state of the target, the moving features of the maneuvering target can be learned from the noisy input data. By incorporating these features into the traditional DP-TBD algorithm, the state transition set can be adjusted in time with the changes in the moving state of the target so that the new algorithm is capable of effectively recursively accumulating the movement trend of the maneuvering small and weak target. Simulation results show that the new algorithm is able to effectively accomplish the task of detecting and tracking maneuvering small and weak targets, and it achieves improved detection and tracking probabilities.

Keywords: Dynamic programming, Tracking before detection, LSTM, State transition set

1 Introduction

In an actual complex electromagnetic environment, for small targets or weak targets affected by electromagnetic interference, radar antennae may receive very weak target echo signals. The traditional detection-before-tracking (DBT) method has been unable to reliably achieve detection and tracking. To solve the problem of detecting and tracking small and weak targets, the tracking-before-detection (TBD) method was recently proposed. The TBD method does not set the threshold of each frame to detect targets; instead, through the accumulation of multiframe echo data, it utilizes the differences among the correlations between the targets and noise or clutter in multiple time frames to obtain the target detection results and produce a target tracking trajectory. We can simply use exhaustive methods to solve such problems, but it is almost impossible to implement them because as the number of frames increases,

the computational burden quickly becomes unsustainable. To reduce the computational burden and make it feasible, researchers have successively proposed a TBD algorithm based on dynamic programming (DP-TBD) [1, 2], a TBD algorithm based on the Hough transform (HT-TBD) [3], a TBD algorithm based on particle filtering (PF-TBD) [4, 5], and a TBD algorithm based on random finite sets (RFS-TBD) [6, 7]. Among them, the DP-TBD algorithm has become a research hotspot in recent years because of its clear thought process, easy implementation and excellent performance.

The essence of dynamic programming is to transform a high-dimensional multistage decision optimization problem into several low-dimensional interrelated subproblems and solve them. The optimization dimensions decrease, and thus the computational burden becomes smaller.

According to its value function, DP-TBD can be classified into value functions based on amplitudes, value functions based on posterior probability densities and value functions based on log-likelihood ratios. The principle of the first kind of algorithm is relatively simple; it does not possess prior clutter information, and its detection performance is not affected by target amplitude fluctuations. However, its signal-to-noise ratio (SNR) cannot be too low, and it is only applicable to targets with approximately linear motion. The second and third types of algorithms can detect a maneuvering target with a very low SNR, but they need to know the prior clutter distribution. In addition, the third type is more suitable for environments with non-Gaussian noise.

Barniv [8] first proposed the use of the DP algorithm to achieve TBD and analyzed the resulting target detection performance by using the likelihood function as the value function. Arnold [9] further developed similar algorithms and proposed an in-frame DP search method that is capable of detecting targets below 0 dB. After that, Tonissen et al. [10] proposed taking the signal amplitude of the target as the value function of the DP-TBD algorithm for the first time; this approach is able to detect the moving target of the fluctuation model. According to the extreme value theory (EVT) and the generalized extreme value theory (GEVT), they obtained the conclusion that the statistical distribution of the value function after DP-TBD accumulation is similar to the Gumbel distribution. Johnston et al. [11] analyzed the mechanism of DP-TBD algorithm and obtained explicit expressions of asymptotic false-alarm probability and track detection probability by using EVT. Buzzi et al. [12] studied the application of the DP-TBD algorithm based on generalized likelihood ratio detection (GLRT) in an airborne radar model.

In recent years, researchers have conducted a lot of research on DP-TBD algorithm. One important direction is the improvement of merit function (MF) to reduce the effect of MF diffusion. Succary et al. [13] proposed a merit function based on the system memory coefficients to improve the system performance. Zhu et al. [14] analyzed the causes of the MF loss, noted that missing target detection information is helpful for preventing the MF loss, and proposed a candidate plot-based DP-TBD (CP-DP-TBD) method, which provided candidate plots carrying missing target detection information through an improved MF transfer program. Wen et al. [15] proposed an improved Doppler-supervised DP-TBD architecture. The architecture uses the dual-domain MF to integrate both the inverse shadow amplitude in SAR images and the Doppler energy in the RD spectrum to achieve more accurate state estimation.

Improvements to state transition constraints have also been extensively studied. Grossi et al. [16] proposed a two-step approach in which measurements of the likelihood ratio exceeding the main threshold in each frame were retained in the review stage, and final state transition decisions were made through the generalized likelihood ratio test. Xing et al. [17] proposed a DP-TBD algorithm with adaptive state transition set, which introduced Kalman filtering and target state transition probability into the traditional algorithm to improve the search efficiency of maneuvering targets. Zheng et al. [18] used the exponential smoothing prediction method to estimate the state of candidate targets according to the historical trajectory, and substituted the estimated state into the state transition probability model.

Extensions of second-order Markov chain for state transitions have also been studied. Hu et al. [19] proposed that the subsequent observation values can be used for correction when determining the state transition of the target, and the direction weighting method was introduced to reduce the false tracks. Wang et al. [20] proposed to use a second-order Markov model to model the target state transition process of the previous two-frame, and on this basis to transform the traditional DP optimization into a series of two-dimensional optimization. Fu et al. [21] proposed an improved second-order DP algorithm, which estimated the current state of pixels on the image plane by adding the maximized optimal MF of the previous two frames and the observed data of the current frame. Meanwhile, in order to inhibit the MF diffusion, the sequential and reverse observation data were connected end to end to form a ring structure. In addition, some scholars have extended the application of DP-TBD algorithm. Li et al. [22] used keystone transformation (KT) and phase gradient autofocusing (PGA) algorithms for offset compensation to improve the SNRs of moving targets. And an incoherent integration method combining DP-TBD and joint intensity-spatial constant false-alarm rate (J-CA-CFAR) was proposed. Lu et al. [23], aiming at the problem that sea targets need relatively long coherent integration times (CITs), which is not conducive to the detection and tracking of aerial targets, proposed selecting the pulse number in the CIT by using prior airborne target motion knowledge for coherent accumulation processing; then, they used the DP-TBD method to realize the noncoherent accumulation of detection and tracking for aerial targets.

The above studies optimized the traditional DP-TBD algorithm in terms of the MF loss, state transition constraint, second-order Markov chain, applied preprocessing to improve the SNR and achieve a better CIT, and achieved certain effects. However, the detection and tracking of weak targets with strong maneuverability has not been effectively realized. This is because the range of state transition set applied in the above DP-TBD algorithms is manually preset, or estimated by smoothing algorithm, or estimated by Markov chain. When the target is non-cooperative and its motion state is difficult to estimate, the state transition set obtained by the traditional method is difficult to adapt to the state changes. If the preset range is smaller, the target cannot be effectively detected, while if it is larger, a heavier burden is imposed on the algorithm calculation.

To solve this problem, it is difficult to use the above traditional methods. Considering the rapid development of deep learning technology in recent years, especially the long short-term memory (LSTM) network, which can recursively process historical data and model historical memory, is suitable for processing time series with strong

correlation and uncertain length of sequence information. Inspired by this, this paper studies the combination of LSTM network and traditional DP-TBD algorithm to address the above puzzle. With the powerful learning ability of LSTM network, the long-term dependence features of target motion and measurement can be learned from the training of a large number of training data, and then the target motion state can be accurately estimated in the prediction stage according to the observed value of the current frame to the target and its state information of the historical frame. Therefore, we propose to integrate LSTM network into DP algorithm structure to form LSTM-DP-TBD architecture. On the basis of accurately predicting the motion state of the target, this architecture can improve the state transition set in DP-TBD to be determined by the predicted motion state parameters. As a result, this architecture can effectively solve the problem of adaptive setting of state transition sets without the need for clutter and noise prior distribution information and preset values, so as to enhance the ability to detect and track weak targets with strong mobility.

The contribution of the work can be summarized in the following:

1. In order to solve the problem that the state transition set needs adaptive change when the traditional DP-TBD algorithm face with non-cooperative target, inspired by LSTM network technology, we propose a new LSTM-DP-TBD target tracking architecture which combines DP-TBD and LSTM network. We model LSTM networks for motion state estimation of non-cooperative target. Based on the long-term dependence of its learning, the architecture can accurately estimate the motion state of the target according to the current observed value and historical information, and realize the dynamic self-adaptation of the state transition set in the structure of DP-TBD algorithm after embedding in the system. The advantages of this architecture are that it is not necessary to know the prior distribution of the motion model and noise of the target and the default value of the transfer set in advance.
2. We use a large amount of training data generated on sampling the widely used non-linear maneuvering radar target time series model.
3. From the qualitative and quantitative simulation results, it is proved that the detection and tracking performance of this architecture is stronger than that of traditional DP-TBD methods in TBD target tracking tasks.

2 Related work

Target tracking algorithm based on video is the fastest and most comprehensive development direction of target tracking technology. It is to establish the position relation of the object to be tracked in the continuous video sequence and obtain the complete motion trajectory of the object. In this process, the expression ability of image features plays a crucial role in video target tracking. Generally, video tracking problems can be divided into classification tasks and estimation tasks. The former is mainly to divide the image area into foreground and background to provide the rough position of the target in the image robustly. The latter is the estimated target state, which is commonly represented by a boundary box in a video image.

In the past few years, the focus of video object tracking research is object classification. One of the most concerned is the classifier based on correlation filtering. This kind of method calculates the reliable confidence in a dense two-dimensional grid through the cyclic matrix, and its regression model can be given by the discrete Fourier transform, so that the speed of training and testing can be greatly improved. Many of these methods have been proved to be very successful in video target tracking, such as MOSSE [24], KCF [25], DSST [26], etc. This kind of methods have a very prominent speed advantage, but they commonly used image features represented by HoG and CN make the performance improvement become difficultly.

Depth features represented by convolutional neural network (CNN) have stronger feature expression, generalization and migration capabilities. Some studies have proposed using CNN features for visual tracking. Qi et al. [27] proposed to build different weak tracers by applying correlation filters to the output of different layers of CNN, and then hedged them into a stronger tracer by online decision theory hedging algorithm. Yang et al. [28] proposed an improvement to the online discriminant approach in terms of providing more compact and richer training data and introducing statistic-based losses to obtain more discriminant features.

Accurate target estimation is mainly embodied in the accurate estimation of the target tracking box, which is a complex task and requires advanced understanding of the target attitude. Early accurate target estimation has not been achieved, and most methods adopt simple multi-scale detection strategy. Qi et al. [29] proposed that gradient histogram (HOG) features were used to train SVM classifiers for selection, and then segmentation algorithm was used to determine the appropriate size of the tracking box. Qi et al. [30] proposed to adaptively utilize level set segmentation and boundary box regression techniques to obtain more compact boundary boxes.

The recent research direction of target estimation is to learn prior knowledge by a large number of offline training. Such methods are mainly represented by the popular Siamese network structure in recent years. Siamese structure uses two CNN networks with shared weights to obtain the feature vectors of two input images, calculate their similarity through cross-correlation, and then track the target by searching the image area most similar to the target template, which can effectively achieve end-to-end training. This kind of method first received attention from SiamFC [31], which trained Siamese network as image similarity learner in offline stage, and then estimated the similarity online in tracking stage. Paul et al. [32] proposed the duplicate detector Siam R-CNN, which integrated Faster R-CNN into the Siamese architecture. Through determining whether the region proposal is the same as the template region, and regressing of the boundary box of the target, the schema can redetect template objects anywhere in the image. The fusion of target classification and estimation is also studied. Martin et al. [33] proposed a tracking system composed of dedicated target estimation and target classification. Through offline learning, the target estimation component is trained to predict the intersection over union (IoU) overlap between the target and the estimated boundary box, thus incorporating high-level knowledge into the target estimation. Bhat et al. [34] proposed a discriminant model prediction architecture for tracking, which consists of two branches: a target classification branch for distinguishing targets from the background, and a boundary box estimation branch for predicting accurate target boxes, both of

which input depth features from a common backbone network. By discriminating learning losses in the learning target model and optimizing strategies based on the steepest descent method, it can make full use of the background information and has the online discrimination ability to update the target model with new data. Shen et al. [35] proposed an improved unsupervised tracking framework of Siam tracker through forward and backward tracking video, aiming at learning time mapping on classification branch and regression branch. Some scholars have studied the new application directions of Siamese network. Qi et al. [36] proposed a face tracking method based on Siamese CNN. The L-CNN and G-CNN were designed to capture and authenticate face information from the local and global levels, respectively, and a boundary box tracking method for faces was realized. Liu et al. [37] extended boundary box estimation to multi-target UAV tracking, and used boundary box estimation, heat map tracking and ID feature updating to complete multi-target detection.

However, these mainstream video target tracking methods are rarely applied to radar dim target tracking. This may be due to several reasons.

1. The radar weak signal tracking problem (non-imaging radar) concerned in this paper actually belongs to the late data processing stage in the radar system, that is, before the radar threshold detection processing in the early stage. The input is the point-track data of range-Doppler domain with clutter and noise interference. Common solutions are based on the target motion state estimation methods, such as KF, EKF, PF, TBD, etc.
2. Deep learning methods require a large number of publicly labeled training data sets, but it is difficult to obtain real radar received data in the scenario of radar tracking dim target. Currently, data sets generated by simulation are commonly used [7, 38].

Based on this, this paper does not adopt the current mainstream video target tracking methods, but leverages the powerful target state prediction ability of LSTM to improve the detection and tracking ability of the traditional DP-TBD structure to dim targets with relatively strong maneuverability.

3 Method

3.1 LSTM based on deep learning

In recent years, deep learning has made great progress in many applications, especially in the field of video target tracking in computer vision, including pedestrian surveillance [39], vehicle monitoring [40], biological sequence tracking [41] and other applications.

RNNs form an important branch of deep learning. Due to their special structure and characteristics. RNNs are particularly suitable for processing time-dependent sequence information. Therefore, an RNN is able to solve the target state tracking problem.

However, the structure of the basic RNN cannot store long-term sequence signals in memory, and serious gradient disappearance or gradient explosion problems may occur [42]. The main solution is to use an LSTM network, which can process long sequence signals more effectively.

LSTM is an RNN with an enhanced memory function [43]. The memory unit contains four parts: an input gate, a forgetting gate, an output gate and a self-circulation connection.

LSTM remembers or discards memory cell states by controlling the outputs of the three gates. The combination effect produced by the four parts enables the network to store or access sequence information for a long time, thus mitigating the gradient vanishing problem.

In this article, the utilized LSTM structure is described as follows [44, 45]:

$$f_t = \sigma(W_{fh}h_{t-1} + W_{fx}x_t + b_f), \quad (1)$$

$$i_t = \sigma(W_{ih}h_{t-1} + W_{ix}x_t + b_i), \quad (2)$$

$$o_t = \sigma(W_{oh}h_{t-1} + W_{ox}x_t + b_o), \quad (3)$$

$$\tilde{c}_t = \tanh(W_{ch}h_{t-1} + W_{cx}x_t + b_c), \quad (4)$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tilde{c}_t, \quad (5)$$

$$h_t = o_t \otimes \tanh(c_t), \quad (6)$$

where σ is the sigmoid function and \otimes denotes elementwise multiplication.

We can see that LSTM is able to be interpreted as resetting the memory according to the forgetting gate, writing to the memory according to the input gate, reading from the memory according to the output gate, and finally forming the output and a hidden state. The values of the middle memory cell and all gates depend on the input at the current time, as well as all parameters. For a multilayer LSTM network, the hidden state of the first layer is treated as the input of the second layer.

To train the LSTM network, it is necessary to use loss a function to measure the error generated by the network output. The common loss function is the mean squared error function:

$$L(x, \hat{x}) = \sum (x_i, \hat{x}_i)^2, \quad (7)$$

where x is the true output value and \hat{x} is the output value predicted by the network.

During the training process, the random gradient descent optimization algorithm is generally used to obtain the gradient of the network parameters, and a variable learning rate is set to control its continuous change in the direction that reduces the loss function until the minimum loss function is found; the results are the convergence parameters.

3.2 Traditional DP-TBD algorithm

It is generally assumed that K frames of data are contained in a DP-TBD processing batch, and the target moves in an x - y two-dimensional plane. At time k , the motion state of the target is:

$$x_k = (px_k, vx_k, ax_k, py_k, vy_k, ay_k), \quad (8)$$

where px_k, py_k represent the position of the target in the x and y directions at time k, vx_k, vy_k represent the speed of the target in the x and y directions at time k, and ax_k, ay_k represent the acceleration in the x and y directions at time k, respectively.

The measurement at each moment is a two-dimensional pixel plane. Assuming that the measurement plane has $N_x \times N_y$ resolving units, the measurement plane at time k can be expressed as an $N_x \times N_y$ matrix:

$$z_k = [z_k(i, j)], \quad i = 1, \dots, N_x, j = 1, \dots, N_y \quad (9)$$

The implementation steps of the algorithm are as follows.

1. Initialization: For the discrete target state shown in Eq. (8),

$$I_1(x_1) = U(z_1|x_1), \quad (10)$$

$$S_1(x_1) = 0, \quad (11)$$

where $I_1(x_1)$ is the accumulation value function corresponding to the target state x_1 in frame 1; $S_1(x_1)$ is a transition function, which is used to store the target state transition relationship between each pair of frames. $U(z_1|x_1)$ is the value function of the measurement plane.

2. Recursive accumulation: When $2 \leq k \leq K$, the state x_k has

$$I_k(x_k) = \max_{x_{k-1} \in \varphi(x_k)} [I_{k-1}(x_{k-1}) + L(x_k|x_{k-1})] + U(z_k|x_k), \quad (12)$$

$$S_k(x_k) = \arg \max_{x_{k-1} \in \varphi(x_k)} [I_{k-1}(x_{k-1}) + L(x_k|x_{k-1})], \quad (13)$$

where $\varphi(x_k)$ represents the state transition set of the target state x_k during a frame time, that is, the set of all possible positions from frame k-1 to frame k. Let the number of transition states of the target state be 16 [11]; then, the set of possible positions is

$$\varphi(x_k) \in \{[px_k + vx_k - \delta_x, py_k + vy_k - \delta_y]; \delta_x, \delta_y = -2, -1, 0, 1\}, \quad (14)$$

$L(x_k|x_{k-1})$ represents the transition cost function of the target state from frame k-1 to frame k.

3. End of the iterative process: The threshold is set as V_K , and

$$\hat{x}_K = \arg \max_{x_K \in R} I_K(x_K), \quad (15)$$

$$s.t. I_K(\hat{x}_K) > V_K, \quad (16)$$

4. Trace back: If $I_K(\hat{x}_K) > V_K$, let $k = K - 1, \dots, 1$; then,

$$\hat{x}_k = S_{k+1}(k+1), \quad (17)$$

Thus, the target track estimated by the DP-TBD algorithm is $\{\hat{x}_1, \dots, \hat{x}_k\}$.

From the above implementation steps, it can be seen that the key to the DP-TBD algorithm is to select an appropriate value function. The selection criterion can reflect the motion correlation difference between the target and the clutter characteristics.

Three common methods can be used to select the target value function.

1. Value function based on the target amplitude: The essence of the application of this function in the DP-TBD algorithm is to use the trajectory correlation of the target to complete the interframe incoherent accumulation of target states. However, its application that the amplitude of the target to be higher than the average amplitude of the noise.
2. Value function based on the posterior probability density function: Essentially, the DP-TBD algorithm approximately estimates the posterior probability density function in the discrete state space. Therefore, the posterior probability density function can be directly used as the value function to express the probability of the target track. Thus, the target state sequence that can achieve the maximum value is the most likely target track. In reference [46], the recurrence formula of the value function based on the posterior probability density function was derived as follows:

$$I_k(x_k) = \max_{x_{k-1} \in \varphi(x_k)} [I_{k-1}(x_{k-1}) + \log p(x_k|x_{k-1})] + \lg\left(\frac{p(z_k|x_k)}{P(z_k|H_0)}\right), \quad (18)$$

where the log-likelihood function $\log p(z_k|x_k)$ indicates the probability that the cell amplitude comes from the target. The transfer cost function $\log p(x_k|x_{k-1})$ indicates the motion characteristics of the target track.

3. Value function based on the likelihood ratio: Arnold [8] of Stanford University first proposed the log-likelihood ratio value function:

$$I_k(x_k) = \max_{x_{k-1} \in \varphi(x_k)} [I_{k-1}(x_{k-1}) + \log p(x_k|x_{k-1})] + \log p(z_k|x_k), \quad (19)$$

Under Gaussian noise, the detection performances of the second and third class value functions are equivalent, and the third class has better nonlinear statistical properties under non-Gaussian noise.

Another key point is that the directly set state transition value determines the ability of the DP-TBD algorithm to detect and track maneuvering targets. The traditional algorithm does not take the real-time changes exhibited by the motion state of the target into account, and its value range is directly determined by the preset maximum and minimum speeds. However, if the target's mobility is stronger than this range, the detection and tracking performance of the algorithm become seriously degraded.

4 Our approach

Considering detection performance and ease of implementation, in this paper, we choose to achieve the second kind of value function.

We focus on the second key point. The state transition set used in the recursive accumulation step of the traditional DP-TBD algorithm is determined by the preset

speed range, which leads to poor detection and tracking performance when applied to targets with strong maneuverability. In this paper, an LSTM network is innovatively incorporated into the recursive accumulation process of the DP-TBD algorithm. The powerful online learning ability of LSTM is used to estimate the motion state of the potential target so that the state transition set in the recursive accumulation step of the DP-TBD algorithm can be adjusted according to the changes exhibited by the actual motion state of the target.

The advantages of LSTM are that it not only has the ability to process long-term information but also does not have too many restrictions, so it can obtain a better tracking effect for a maneuvering target. The designed network structure is shown in Fig. 1 below:

As shown in Fig. 1, an LSTM network with two stacked layers is used to complete the estimation process from the target observation data y_k to the target motion state x_k , and its hidden layers are represented by memory units C_k^p . The loss function of the network parameter optimization step is defined as follows:

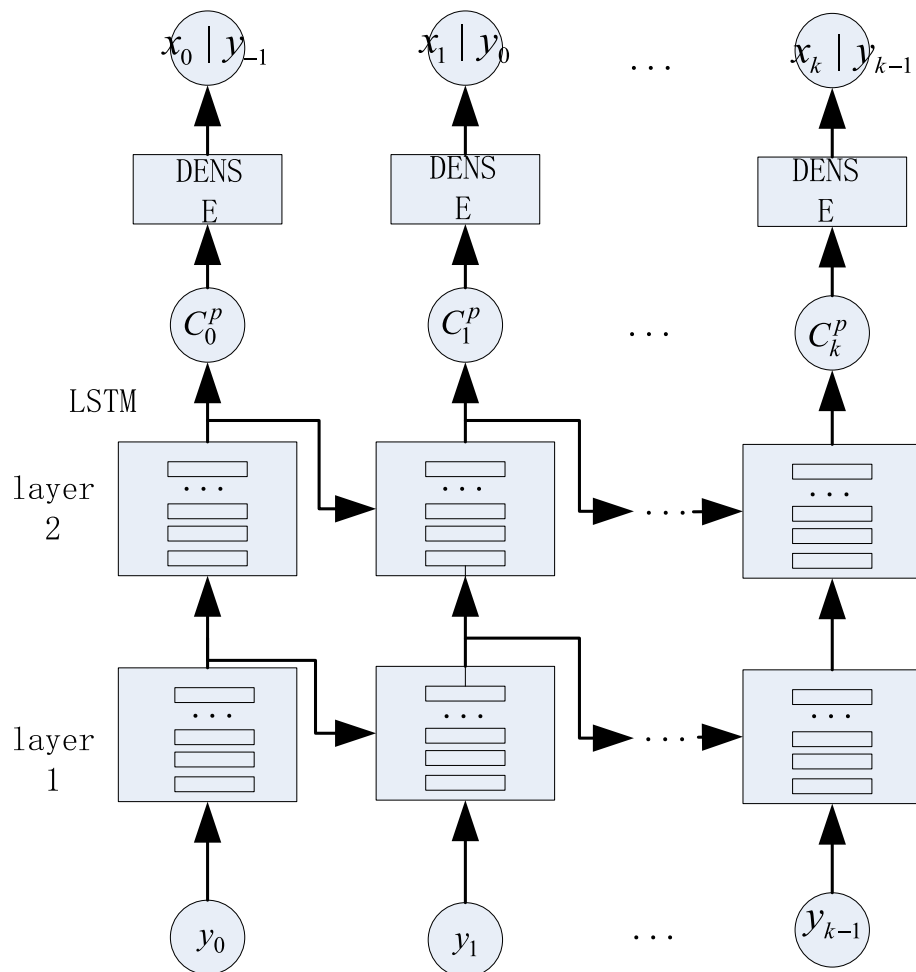


Fig. 1 Schematic diagram of the designed LSTM network structure

$$L(x, \hat{x}, \theta_p) = \sum_{i=1}^T (x_i - \hat{x}_i | y_{i-1}), \quad (20)$$

After obtaining the predicted result, formula (14) of the aforementioned state transition set is adjusted as follows:

$$\varphi(x_k) \in \{[px_k + \dot{v}x_k \cdot (1 - \delta_x), py_k + \dot{v}y_k \cdot (1 - \delta_y)]; \delta_x, \delta_y = -2, -1, 0, 1\}, \quad (21)$$

where $\dot{v}x_k, \dot{v}y_k$ all come from the target states \dot{x}_k predicted by the LSTM network.

The main steps of the improved DP-TBD algorithm are as follows.

1. Initialization. When $k=1$,

$$I_1(x_1) = \log p(z_1 | x_1), \quad (22)$$

$$S_1(x_1) = 0, \quad (23)$$

$$x_1 = (px_1, vx_1, ax_1, py_1, vy_1, ay_1), \quad (24)$$

2. Recursive accumulation. When $2 \leq k \leq K$, for the state,

A. A state prediction is acquired, which can be obtained through the above LSTM network:

$$\dot{x}_k = (\dot{p}x_k, \dot{v}x_k, \dot{a}x_k, \dot{p}y_k, \dot{v}y_k, \dot{a}y_k), \quad (25)$$

Substituting $\dot{v}x_k, \dot{v}y_k$ into Eq. (21), the state transition set $\dot{\varphi}(x_k)$ adjusted by the prediction is obtained.

B. Recursive accumulation is performed:

$$I_k(x_k) = \max_{x_{k-1} \in \varphi(x_k)} [I_{k-1}(x_{k-1}) + \log p(x_k | x_{k-1})] + \log p(z_k | x_k), \quad (26)$$

$$S_k(x_k) = \arg \max_{x_{k-1} \in \varphi(x_k)} [I_{k-1}(x_{k-1}) + \log p(x_k | x_{k-1})], \quad (27)$$

The state transition set $\dot{\varphi}(x_k)$ is determined by step A above.

3) Termination of judgment.

$$s.t. I_k(\hat{x}_K) > V_K, \quad (28)$$

$$\hat{x}_K = \arg \max_{x_K \in R} I_k(x_k), \quad (29)$$

4) Track retracing. Letting $k = K - 1, \dots, 1$, we have

$$\hat{X}_k = S_{k+1}(k + 1), \quad (30)$$

5 Experiments

In this section, to demonstrated the tracking performance of the designed LSTM-DP-TBD algorithm for nonlinear small and weak targets, we use CS model simulation data and compare the tracking performance of our algorithm with that of the traditional DP-TBD algorithm for nonlinear dim targets under a series of different SNR conditions.

The current statistical model (CS) is a typical nonlinear motion model that can describe the motion state of a maneuvering target. It is able to effectively simulate the state change exhibited by the target when a maneuvering mutation occurs. The radar sampling period is set as T , and the state equation of the CS model is

$$x_k = F(x_{k+1}) + \bar{a}H + u_k, \quad (31)$$

where F is the state transition matrix of the target, which is expressed as

$$F_k = \begin{bmatrix} \Psi & 0_{3 \times 3} \\ 0_{3 \times 3} & \Psi \end{bmatrix}, \quad (32)$$

where $0_{3 \times 3}$ is a zero matrix with three rows and three columns, and the expression of Ψ is as follows:

$$\Psi = \begin{bmatrix} 1 & T & (e^{-\alpha T} + \alpha T - 1)/\alpha^2 \\ 0 & 1 & (-e^{-\alpha T} + 1)/\alpha \\ 0 & 0 & e^{-\alpha T} \end{bmatrix}, \quad (33)$$

where α is the maneuvering frequency, and the maneuverability reflected by the CS model varies with its value.

In Eq. (31), \bar{a} is the mean acceleration value. u_k represents state noise that follows a normal distribution $u_k \sim N(0, \sigma_u^2)$. $H = [H_1 \ H_2]^T$; H_1 and H_2 are expressed as:

$$H_1 = H_2 = \begin{bmatrix} (1 - \alpha T + \alpha^2 T^2/2 - e^{-\alpha T})/\alpha^2 \\ (-1 + \alpha T + e^{-\alpha T})/\alpha \\ 1 - e^{-\alpha T} \end{bmatrix}, \quad (34)$$

The target observation equation is

$$z_k = \begin{cases} A_k + v_k, & Y \\ v_k, & N \end{cases}, \quad (35)$$

where the Y branch represents the case with the target at frame k , and the N branch represents the case without the target at frame k . A_k is the target amplitude; v_k represents observation noise and follows a normal distribution $v_k \sim N(0, \sigma_v^2)$.

The size of the radar observation area is set as $N_x \times N_y = 100 \times 100$, the resolution unit is $\Delta x = \Delta y = 2$, the total frame length is $K = 10$, and the radar scanning time interval is $T = 1.2s$.

By using this model for simulation, first, the training dataset needed to train the aforementioned LSTM network can be obtained. Specifically, a random observation target is generated within a certain observation time frame, and the initial state is randomly set within a certain range. According to the target state equation of the

model, a target state sequence with 60 random paths is generated, and a corresponding observation sequence is generated according to the target observation equation. The dimensions of the target state are the six dimensions mentioned above.

In the implementation of the LSTM network, a two-layer stacked LSTM network is adopted, and the number of hidden states in each layer is set to 256. To prevent overfitting, each LSTM layer is followed by a dropout layer with a ratio of 0.3. A 1-to-1 network structure is chosen; that is, 1 data point is input to obtain the next predicted data point. After this, 796,166 network parameters are set, and the best values of these parameters need to be found through training. The training loss function is the aforementioned loss function, and the adaptive moment estimation (Adam) optimization algorithm is adopted. The training dataset generated above is used to train and test the LSTM network.

Second, the validation data used to verify the performance of the algorithm can be obtained. The initial state of the target is set to $x_1 = (8, 3, 0, 5, 2, 0)^T$. The target is set to execute a strong steering maneuver in the observation area.

In this paper, the designed LSTM-DP-TBD algorithm is compared with the traditional DP-TBD algorithm in terms of the following aspects. (1) The amplitude distributions of the value function after K accumulation frames are compared to show the difference between the value function aggregation effects of the two algorithms. (2) The target detection probability P_d and tracking probability P_t are compared. P_d is defined as the probability of detecting the target after K accumulation frames, allowing for an error of one resolution unit. After detecting the target, P_t is defined as the probability that the estimated state obtained after track recovery is within one resolution unit of the real state in each frame. These probabilities are used to evaluate the detection and tracking performance of the two algorithms.

Firstly, simulation experiment 1 is first carried out: when $\text{SNR} = 10$ dB is given, the value function distributions of the two DP-TBD algorithms are compared.

The value function distribution based on the traditional posterior probability value function of the DP-TBD algorithm is shown in Fig. 2, the preset speed range is 3–0 times, and K frames are accumulated. As can be seen from the figure, the traditional DP-TBD algorithm produces an obvious agglomeration effect, which brings difficulties to the subsequent termination decision steps.

The value function distribution produced by the proposed LSTM-DP-TBD algorithm after K accumulation frames is shown in Fig. 3. It can be seen from the figure that the new LSTM-DP-TBD algorithm is able to effectively suppress the agglomeration effect, and the value function obtained after K accumulation frames is highlighted.

Secondly, using the proposed algorithm LSTM-DP-TBD, the traditional DP-TBD algorithm and the algorithm in reference [19], named D-DP-TBD, simulation experiment 2 is carried out to compare the target detection probabilities P_d and tracking probabilities P_t under a varying SNR. The results are obtained by conducting 2,000 Monte Carlo runs during the experiment.

As shown in Fig. 4, the detection probability P_d curves produced by the DP-TBD, D-DP-TBD and LSTM-DP-TBD algorithm as the SNR changes are compared. As can be seen from the figure, when the SNR is 2 dB, the P_d value of DP-TBD is close to 0, while that of D-DP-TBD is close to 0.1, and that of LSTM-DP-TBD is close to

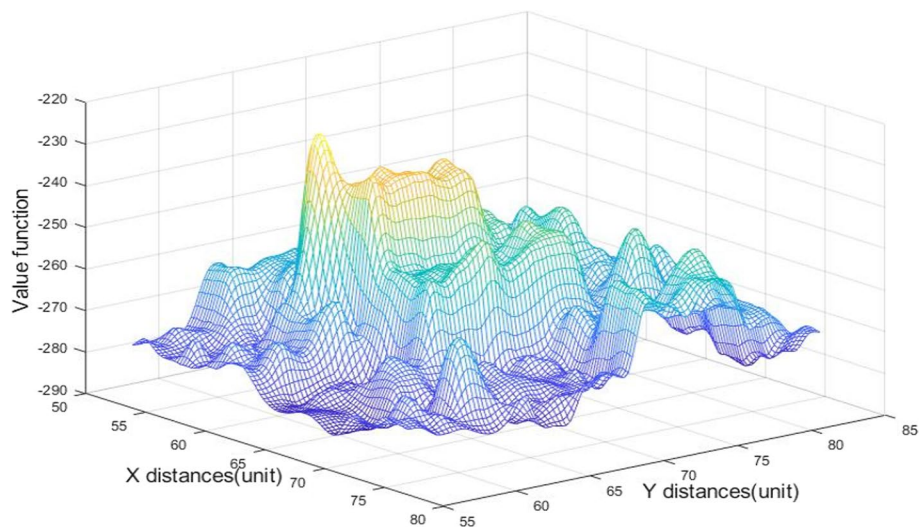


Fig. 2 Amplitude distribution diagram of the traditional DP-TBD algorithm value function when the SNR is 10 dB

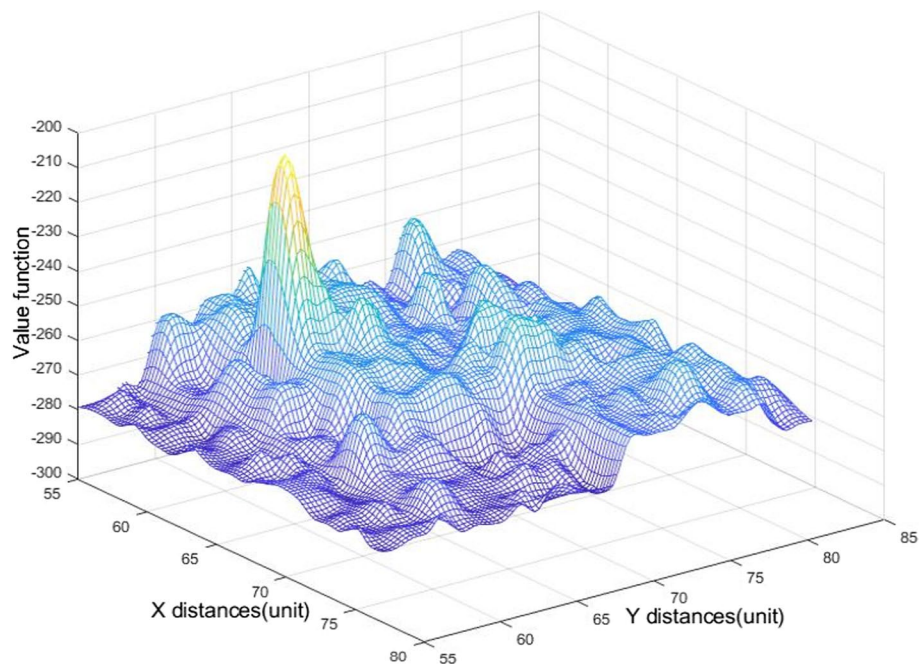


Fig. 3 Amplitude distribution diagram of the LSTM-DP-TBD algorithm value function when the SNR is 10 dB

0.2. when the SNR is 1 dB, the P_d values of DP-TBD and D-DP-TBD are both close to 0, while that of LSTM-DP-TBD is close to 0.1. This shows that LSTM-DP-TBD algorithm has better performance for low SNR signal detection. When the SNR is greater than 2 dB, the P_d values of all methods begin to rise. When the SNR is higher than 5 dB, the P_d of the LSTM-DP-TBD algorithm rises over 0.9, while that of the

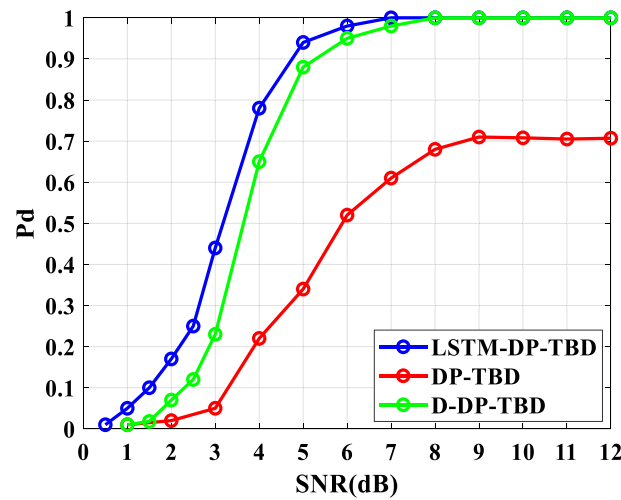


Fig. 4 The detection probability curves produced by the DP-TBD algorithms as the SNR changes

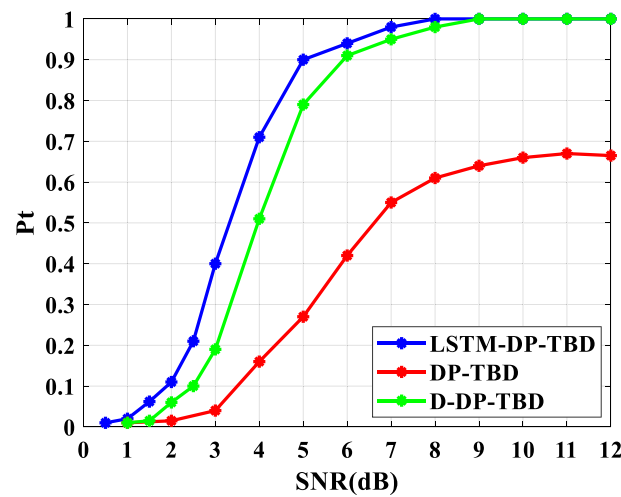


Fig. 5 The track probability curves produced by the DP-TBD algorithms as the SNR changes

D-DP-TBD algorithm tends to rise over 0.9 when the SNR is higher than 6 dB, and that of the DP-TBD algorithm tends to rise to 0.7 when the SNR is higher than 9 dB. Therefore, the detection performance of the LSTM-DP-TBD algorithm is obviously better than that of the compared algorithms.

As shown in Fig. 5, the tracking probability P_t curves produced by the DP-TBD, D-DP-TBD and LSTM-DP-TBD algorithms as the SNR changes are compared. As can be seen from the figure, when the SNR is higher than 5 dB, the P_t of the LSTM-DP-TBD algorithm rises over 0.9, while that of the D-DP-TBD algorithm tends to rise over 0.9 when the SNR is higher than 6 dB, and that of the DP-TBD algorithm tends to rise to 0.65 when the SNR is higher than 9 dB. Therefore, the tracking performance of the LSTM-DP-TBD algorithm is better than that of the compared algorithms.

6 Conclusion

In this paper, aiming at the problem that the state transition set used by the traditional DP-TBD algorithm in the recursive accumulation step is set as a fixed speed range, which leads to an insufficient tracking ability for small and weak targets with strong maneuvers, an LSTM network is applied to the DP-TBD algorithm, and a new LSTM-DP-TBD algorithm is proposed. Thus, the state transition set can be adjusted with the changes exhibited by the target state. The detection and tracking capability of the network for maneuvering targets is enhanced. The simulation results show that the proposed algorithm is superior in terms of suppressing the agglomeration effect and detecting and tracking. However, the LSTM-DP-TBD algorithm is computationally expensive, and determining how to apply it in practice requires further research.

Abbreviations

LSTM	Long short-term memory
DP-TBD	Dynamic programming-based tracking before detection
DBT	Detection before tracking
TBD	Tracking before detection
HT-TBD	Tracking-before-detection algorithm based on the Hough transform
PF-TBD	Tracking-before-detection algorithm based on particle filtering
RFS-TBD	Tracking-before-detection algorithm based on random finite sets
SNR	Signal-to-noise ratio
DP	Dynamic programming
EVT	Extreme value theory
GEVT	Generalized extreme value theory
GLRT	Generalized likelihood ratio detection
KT	Keystone transformation
PGA	Phase gradient autofocusing
CFAR	Constant false-alarm rate
CA-CFAR	Joint intensity-spatial CFAR
MF	Merit function
CP-DP-TBD	Candidate plot-based DP-TBD
CIT	Coherent integration time
RNN	Recurrent neural network
CS	Current statistical
P_d	Detection probability
P_t	Tracking probability

Acknowledgements

The authors would like to express their sincere thanks to the editors and anonymous reviewers.

Author contributions

YL, WC, LD and FS conceived and designed the experiments; FS performed the experiments; FS, WC and LD analyzed the data; FS wrote the paper; YL administrated the project. All authors read and approved the final manuscript.

Funding

This work was funded by the Fundamental Research Funds for the Central Universities under grant 3102019ZX015 and in part by the Fundamental Research Funds for the Central Universities under grant D5000220131.

Availability of data and materials

Unfortunately, the data are not available online. Kindly, for data requests, please contact the corresponding author.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Approved.

Competing interests

The authors declare that they have no competing interests.

Received: 5 November 2022 Accepted: 9 May 2023

Published online: 19 May 2023

References

1. Y. Barniv, O. Kella, Dynamic programming solution for detecting dim moving targets part II: analysis. *IEEE Trans. Aerosp. Electron. Syst.* **23**(6), 776–788 (1987)
2. W. Yi, M.R. Moreland, L.-J. Kong, et al., Multi-target tracking via dynamic-programming based track-before-detect, in *Proceedings of the Radar Conference (RADAR)*, IEEE, (2012), pp. 487–492.
3. B.D. Arlson, E.D. Evans, S.J. Wilson, Search radar detection and track with the Hough transform. *IEEE Trans. Aerosp. Electron. Syst.* **30**(1), 102–108 (1994)
4. M.G. Rutten, N.J. Gordon, S. Maskell, Recursive track-before-detect with target amplitude fluctuations. *Radar Sonar Navig. IEE Proc.* **152**(5), 345–352 (2005)
5. Y. Boers, H. Driessen, A particle-filter-based detection scheme. *Signal Process. Lett. IEEE* **10**(10), 300–302 (2003)
6. S.J. Davey, Comments on "Joint detection and estimation of multiple objects from image observations". *Signal Process. IEEE Trans.* **60**(3), 1539–1540 (2012)
7. M. Barbary, H. Mohamed, A. ElAzeem, Drones tracking based on robust Cubature Kalman-TBD-multi-Bernoulli filter. *ISA Trans.* **12**(114), 277–290 (2021)
8. Y. Barniv, O. Kella, Dynamic programming solution for detecting dim moving targets. *IEEE Trans. Aerosp. Electron. Syst.* **21**(1), 144–156 (1985)
9. J. Arnold, S.W. Shaw, H. Pasternack, Efficient target tracking using dynamic programming. *IEEE Trans. Aerosp. Electron. Syst.* **29**(1), 44–56 (1993)
10. S.M. Tonissen, R.J. Evans, Performance of dynamic programming techniques for track-before-detect. *IEEE Trans. Aerosp. Electron. Syst.* **32**(4), 1440–1451 (1996)
11. L.A. Johnston, V. Krishnamurthy, Performance analysis of a dynamic programming track before detect algorithm. *IEEE Trans. Aerosp. Electron. Syst.* **38**(1), 228–242 (2002)
12. S. Buzzi, M. Lops, L. Venturino, Track-before-detect procedures for early detection of moving target from airborne radars. *IEEE Trans. Aerosp. Electron. Syst.* **41**(3), 937–954 (2005)
13. R. Succary, H. Kalmanovitch, Y. Shurnik et al., Point target detection. *Infrared Technol. Appl.* **3**, 671–675 (2003)
14. Y.R. Zhu, Y. Li, N. Zhang et al., Candidate-plots-based dynamic programming algorithm for track-before-detect. *Dig. Signal Process.* (2022). <https://doi.org/10.1016/j.dsp.2022.103458>
15. L.W. Wen, J.S. Ding, Y. Cheng, Dually supervised track-before-detect processing of multichannel video SAR data. *IEEE Trans. Geosci. Remote Sens.* **60**(1), 238–252 (2022)
16. E. Grossi, M. Lops, L. Venturino, Track-before-detect for multiframe detection with censored observations. *IEEE Trans. Aerosp. Electron. Syst.* **50**(1), 2032–2046 (2014)
17. H. Xing, J. Suo, X. Liu, *A dynamic programming track-before-detect algorithm with adaptive state transition set*, International Conference in Communications, Signal Processing, and Systems; Springer: Singapore, 2020; p. 638–646
18. D. Zheng, S. Wang, C. Liu, *An improved dynamic programming track-before-detect algorithm for radar target detection*, 2014 12th International Conference on Signal Processing (ICSP); 2014; p. 2120–2124
19. H. Lin, S.Y. Wang, Y. Wan, Improvement on track-before-detect algorithm based on dynamic programming. *Air Force Radar Acad.* **24**(1), 79–82 (2010)
20. S. Wang, Y. Zhang, Improved dynamic programming algorithm for low SNR moving target detection. *Syst. Eng. Electron.* **38**(1), 2244–2251 (2016)
21. J. Fu, H. Zhang, W. Luo et al., Dynamic programming ring for point target detection. *Appl. Sci.* **12**, 1151 (2022). <https://doi.org/10.3390/app12031151>
22. C. Li, X. Bai, J. Zhao, et al., *An effective method for weak multi-target detection and tracking in clutter environment*, in *Proceedings of the 6th International Conference on Digital Signal Processing (ICDSP '22)*. Association for Computing Machinery, (2022), p.134–139. <https://doi.org/10.1145/3529570.3529593>.
23. X. Lu, T. Cheng, M. Deng, et al., in *A novel track -before-detect algorithm for airborne target with over-the-horizon radar*. 2022 IEEE Radar Conference (RadarConf22), (2022), p.01–06. doi: <https://doi.org/10.1109/RadarConf2248738.2022.9764334>.
24. D.S. Bolme, J.R. Beveridge, B.A. Draper, Y.M. Lui, Visual object tracking using adaptive correlation filters, in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA (2010), pp. 2544–2550. <https://doi.org/10.1109/CVPR.2010.5539960>
25. J.F. Henriques, R. Caseiro, P. Martins, J. Batista, Highspeed tracking with kernelized correlation filters. *TPAMI* **37**(3), 583–596 (2015)
26. M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, in *Accurate scale estimation for robust visual tracking*. BMVC, p. 678–696, 2014.
27. Y.K. Qi, S.P. Zhang, L. Qin, et al., in *Hedged Deep Tracking*. 2016 IEEE Conference on Computer Vision and Pattern Recognition, p. 868–886, 2016.
28. Y.F. Yang, G.R. Li, Y.K. Qi, et al., in *Release the Power of Online-Training for Robust Visual Tracking*. The Thirty-Fourth AAAI Conference on Artificial Intelligence, p. 1134–1146, 2020.
29. Y.K. Qi, H.X. Yao, X.S. Sun, et al., in *Structure-aware multi-object discovery for weakly supervised tracking*. 2014 ICIP, p. 540–567, 2014.
30. Y.K. Qi, L. Qin, S.P. Zhang et al., Robust visual tracking via scale-and-state-awareness. *Neurocomputing* **329**(1), 75–85 (2019)
31. L. Bertinetto, J. Valmadre, J. Henriques, in *Fully-convolutional siamese networks for object tracking*. 2016 CVPR, p. 1254–1267, 2016.
32. V. Paul, L. Jonathon, H.S. Philip et al., in *Siam R-CNN: Visual Tracking by Re-Detection*. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), p. 1050–1062, 2020.
33. D.Martin, B. Goutam, S.K.Fahad et al., in *ATOM: Accurate Tracking by Overlap Maximization*. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), p. 952–964, 2019.
34. G. Bhat, M. Danelljan, L.V. Gool, et al., in *Learning Discriminative Model Prediction for Tracking*. International Conference on Computer Vision, p. 472–489, 2020.

35. Q.H. Shen, L. Qiao, J.Y. Guo et al., in *Unsupervised Learning of Accurate Siamese Tracking*. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), p. 978–989, 2022.
36. Y.K. Qi, S.P. Zhang, F. Jiang et al., Siamese local and global networks for robust face tracking. *IEEE Trans. Image Process.* **29**(1), 85–97 (2020)
37. S. Liu, X. Li, H.C. Lu et al., in *Multi-Object Tracking Meets Moving UAV*. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), p. 1109–1123, 2022.
38. Y. Xiang, A. Alahi, S. Savarese, in *Learning to Track: Online Multi-Object Tracking by Decision Making*. IEEE International Conference on Computer Vision, p. 4705–4713, 2015.
39. J. Berclaz, F. Fleuret, E. Turetken et al., Multiple object tracking using k-shortest paths optimization. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(9), 1806–1819 (2011)
40. J.R. Perello-March, C.G. Burns, R. Woodman et al., Driver state monitoring: manipulating reliability expectations in simulated automated driving scenarios. *IEEE Trans. Intell. Transp. Syst.* **99**, 1–11 (2021)
41. N. Chenouard, I. Bloch, J.C. Olivo-Marin, Multiple hypothesis tracking for cluttered biological image sequences. *IEEE Trans. Softw. Eng.* **35**(11), 2736–2750 (2013)
42. R.J. Williams, J. Peng, An efficient gradient-based algorithm for on-line training of recurrent network trajectories. *Neural Comput.* **10**(4), 1045–1053 (1990)
43. S. Hochreiter, J. Schmidhuber, Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
44. F.A. Gers, J. Schmidhuber, F. Cummins, Learning to forget: continual prediction with LSTM. *Neural Comput.* **12**(10), 2451–2471 (2000)
45. K. Greff, R.K. Srivastava, J. Koutnik et al., LSTM: a search space odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **28**(10), 2222–2232 (2017). <https://doi.org/10.1109/TNNLS.2016.2582924>
46. W. Yi, Research on track-before-detect algorithms for multiple-target detection and tracking. Dissertation, Chengdu: University of Electronic Science and Technology of China, p. 44–46, 2012

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)
