


Article

Study on Active Tracking of Underwater Acoustic Target Based on Deep Convolution Neural Network

Maofa Wang ¹, Baochun Qiu ^{1,*} , Zeifei Zhu ¹, Huanhuan Xue ^{1,2} and Chuanping Zhou ¹

¹ School of Mechanical Engineering, Hangzhou Dianzi University, Hangzhou 310018, China; wmf@hdu.edu.cn (M.W.); zzf.3691@163.com (Z.Z.); xhh20190028@hdu.edu.cn (H.X.); zhoucp@hdu.edu.cn (C.Z.)

² College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China

* Correspondence: qiubaochun@hdu.edu.cn

Abstract: The active tracking technology of underwater acoustic targets is an important research direction in the field of underwater acoustic signal processing and sonar, and it has always been issued that draws researchers' attention. The commonly used Kalman filter active tracking (KFAT) method is an effective tracking method, however, it is difficult to detect weak SNR signals, and it is easy to lose the target after the azimuth of different targets overlaps. This paper proposes a KFAT based on deep convolutional neural network (DCNN) method, which can effectively solve the problem of target loss. First, we use Kalman filtering to predict the azimuth and distance of the target, and then use the trained model to identify the azimuth-weighted time-frequency image to obtain the azimuth and label of the target and obtain the target distance by the time the target appears in the time-frequency image. Finally, we associate the data according to the target category, and update the target azimuth and distance information for this cycle. In this paper, two methods, KFAT and DCNN-KFAT, are simulated and tested, and the results are obtained for two cases of tracking weak signal-to-noise signals and tracking different targets with overlapping azimuths. The simulation results show that the DCNN-KFAT method can solve the problem that the KFAT method is difficult to track the target under the weak SNR and the problem that the target is easily lost when two different targets overlap in azimuth. It reduces the deviation range of the active tracking to within 200 m, which is 500~700 m less than the KFAT method.

Keywords: DCNN; active sonar; tracking; Kalman filtering; underwater acoustic targets



Citation: Wang, M.; Qiu, B.; Zhu, Z.; Xue, H.; Zhou, C. Study on Active Tracking of Underwater Acoustic Target Based on Deep Convolution Neural Network. *Appl. Sci.* **2021**, *11*, 7530. <https://doi.org/10.3390/app11167530>

Academic Editor: Valentina E. Balas

Received: 24 May 2021

Accepted: 12 August 2021

Published: 17 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In marine activities, active sonar is widely used in underwater moving target detection, recognition, tracking, seabed scanning, navigation, and communication [1–7]. Active sonar is an important device for active tracking of underwater acoustic targets. By periodically transmitting specific signals and analyzing the echo signals received by the array, the target's position, speed, distance, and other characteristic information can be obtained [8]. In recent years, due to the strong demand for marine development research and marine safety, higher requirements have been put forward for the application of active sonar in tracking [9]. Therefore, the research on the active tracking method of underwater moving target has important engineering significance and practical significance.

The current active tracking method for underwater acoustic targets [10] has the following two drawbacks: when the azimuths of two different moving targets overlap, the target data cannot be correctly correlated, leading to the loss of one target; and when the target echo signal is weak and below the detection threshold set by the target verdict, it is more difficult to track such weak SNR targets. In the active tracking of underwater acoustic targets, accurately detecting targets with weak SNR from the signal and correctly correlating the data of different targets has become a hot issue of research. The Kalman filter active tracking (KFAT) method combined with Kalman filter is the most commonly used

method for active tracking of hydroacoustic targets, which transforms the azimuth and distance information of the target from polar coordinates to Cartesian coordinates, and then establishes a motion model of target for processing [11], which can achieve the prediction of azimuth and distance of the tracked target and denoise the tracking results to improve the tracking accuracy [12]. Lei [13] proposed that when tracking a moving target, in the case of decorrelation processing on the conversion deviation of position and distance, the linear part (position measurement) and the nonlinear part (distance-rate measurement) are respectively passed through Kalman filtering and unscented Kalman filter (UKF) for processing. Bar-Shalom [14] studied the application of probabilistic data association (PDA) technology in different target tracking schemes, especially targets with low signal-to-noise ratio (SNR). The accuracy of the target azimuth and distance obtained after filtering depends on the original data measurement, which reduces the deviation of the initial measurement and can improve the accuracy of active tracking.

The information on target scattering characteristics carried in the active sonar echo signal can be used to detect and distinguish different targets [15]. Researchers have used target scattering characteristics for target detection or classification in the field of underwater acoustics [16,17]. In 2007, Young [18] proposed a method to extract features from the target echo signal by imitating auditory perception, and classify the target using Gaussian classification in machine learning. Deep learning has the advantage of automatically extracting target feature information from raw data through learning training and can be used for multi-target recognition and classification. The convolutional neural network (CNN) is a commonly used structure in deep learning that has greatly improved productivity and efficiency in areas such as computer vision, natural language processing, text and speech recognition, and object detection [19]. In 2015, Yang [20] used CNN and existing auditory perception models to extract features of target radiated noise using mel-scale frequency cepstral coefficients to simulate the function of a complete auditory system to identify ship target radiated noise. The results show the feasibility of applying deep learning to the field of underwater acoustics. In 2019, Yao [21] proposed a deep learning method, constructed an underwater acoustic signal feature extraction model based on a generative confrontation network, combined with a deep neural network classifier for modulation recognition, and could effectively extract classification features from underwater acoustic signals. Deep learning has a wide range of application prospects in the field of underwater acoustic targets [22].

In this paper, we propose a DCNN-KFAT method for underwater acoustic moving target tracking. After the target is detected in the early stage of tracking, we calculate the target azimuth and distance, and then use beamforming to perform target azimuth-related weighting on the echo signal, convert the obtained target frequency domain data into time domain data, generate a data set with the target time-frequency image as a sample, and finally use DCNN to train the data set to generate a model to identify the target. In the follow-up process of tracking the target, we use Kalman filter to predict the target azimuth and distance, weight the received echo signal, generate all possible azimuth-related signal time-frequency images, and use the trained model for recognition. We determine the target category according to the output of the model, and finally get the target azimuth and distance. According to the current target recognition result, it is associated with the existing target, and the target tracking information is updated. The rest of the article is as follows: Section 2 describes the active tracking method proposed in this article, Section 3 describes the simulation test, Section 4 describes the discussion, and Section 5 gives the conclusion.

2. Kalman Filter Active Tracking Based on Deep Convolutional Neural Network

After the active sonar emits sound waves, the echo signals received continuously during the period include various reverberation and sound wave scattering caused by impurities. The difference in target geometry will also cause the difference in the echo signal. We can use the information carried in the target echo signal to track the target. Active sonar tracking a moving target is a process of predicting the location, searching

for the target, judging the target, and associating the data. The schematic diagram of the DCNN-KFAT process proposed in this paper is shown in Figure 1.

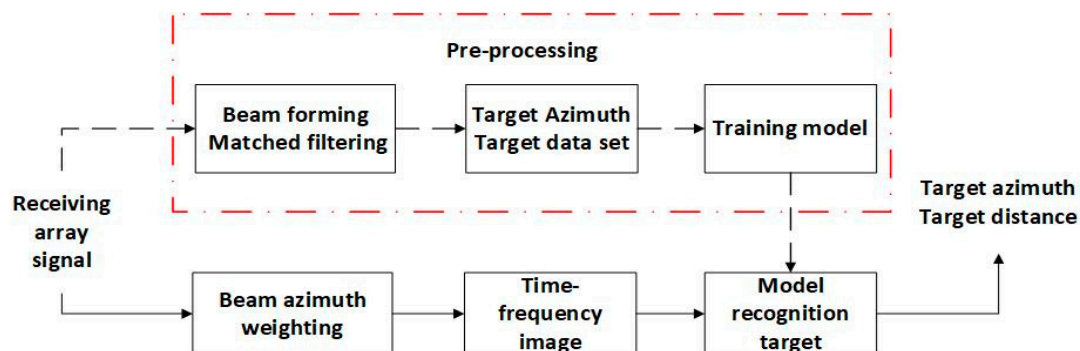


Figure 1. DCNN-KFAT flow chart. The preprocessing step detects the target and generates the target data set, and trains to obtain the model; in the follow-up tracking process, the generated signal time spectrogram with different azimuth weights is recognized by the trained model; finally the target azimuth and distance are obtained.

We use periodic pulse signals as active sonar transmission signals, and receive echo signals with a uniform linear array. In the preprocessing step, the target is detected by beamforming and matching filtering the received echo signal, and the time-frequency spectrogram of the target is obtained after azimuth-weighting the signal according to the detected target azimuth; we label the different targets and generate the dataset, and DCNN is used to train the dataset.

In the subsequent tracking step, the echo signals are weighted by different azimuths to obtain a time-frequency spectrum image of the signal to be detected, and the trained model is used for identification to obtain the azimuth and distance of the target. All steps are described in detail in the rest of this section.

2.1. Active Sonar Echo Signal Preprocessing to Generate Data Set

In active tracking, the first step is to determine the moving target to be tracked and generate a target data set. The specific process is shown in Figure 2. Firstly, we generate a weighting matrix containing the target orientation information based on the target detected from the original array signal, and multiply it with the array signal matrix to obtain the frequency domain data of the target echo signal; then convert the frequency domain data to time domain data and do the short-time Fourier transform to generate the time-frequency images of the target echo signal; finally, the time-frequency images of different targets are labeled and stored in the data set.

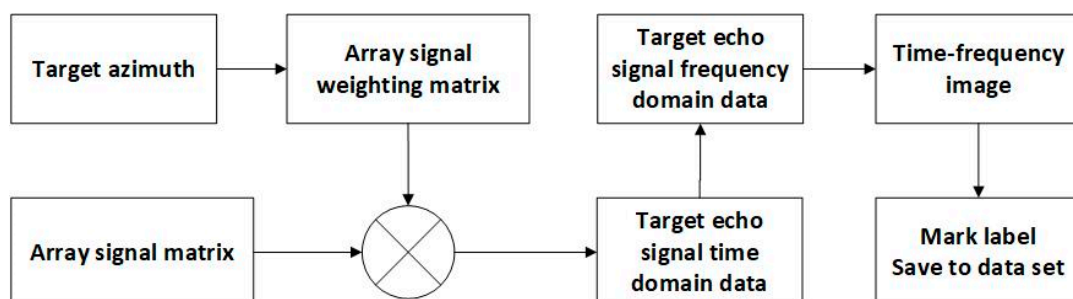


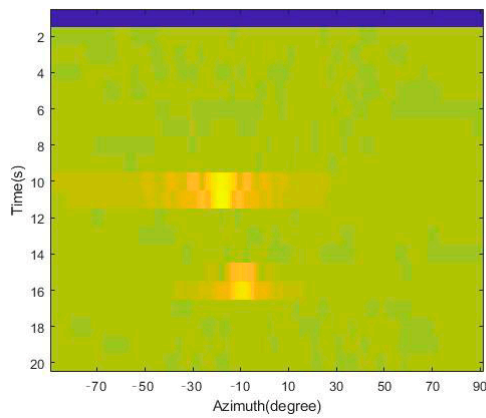
Figure 2. Data set generation process. Generate a weighting matrix according to the target azimuth and multiply it with the array signal to obtain the frequency domain matrix of the target echo signal, then convert it into a time domain matrix, generate a time-frequency image by short-time Fourier transform, and finally label the tag and store it in the data set.

After the active sonar transmits the signal, the array continues to receive the echo signal within a period, accumulate 1 s echo signal $Y(t)$ through Fourier transform to generate a matrix in frequency domain,

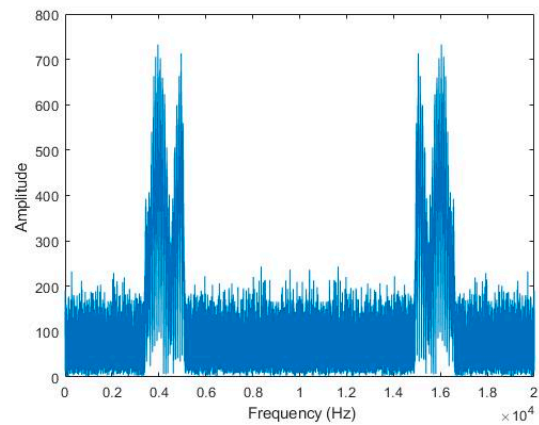
$$F(\omega) = \mathcal{F}[Y(t)] = \int_{-\infty}^{\infty} Y(t)e^{-j\omega t} dt \tag{1}$$

We obtain the spatial energy spectrum through matched filtering and beamforming. The spatial energy spectrum accumulation in the period is shown in Figure 3a. Suspected targets are filtered out through threshold detection, and the approximate position (distance r , azimuth θ) of the target is calculated according to the speed of sound c and echo arrival time t , providing prior information for confirming the target, and determining the tracking target through data association. After the target is determined, a weighting matrix W is generated according to the detected target azimuth θ , as shown in Figure 3b.

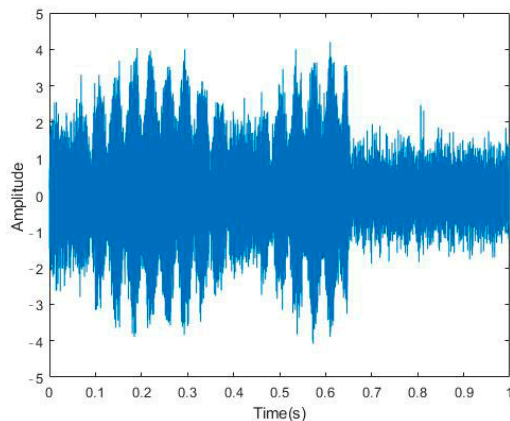
$$W = e^{-j2\pi\omega(m-1)d \sin \theta / c} \quad m = 1, 2, \dots, M \tag{2}$$



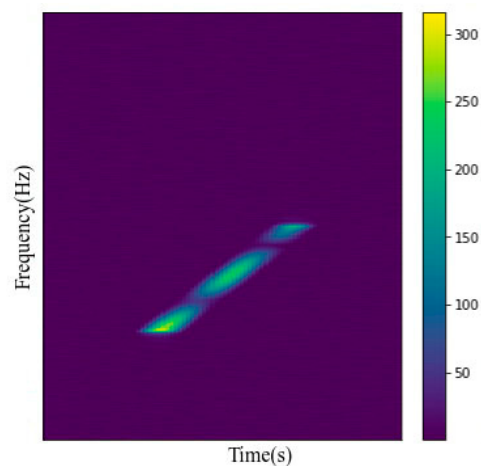
(a) Spatial spectrum of beams within a period—discovering different targets based on energy threshold detection.



(b) Frequency domain waveform after weighting—filtering out the target information in the frequency domain and performing azimuth weighting.



(c) Time domain waveform after weighting.



(d) Target time-frequency image sample.

Figure 3. Time-frequency images related to the target’s azimuth.

We multiply the weighting matrix with the echo signal matrix $F(w)$ to obtain the weighted target frequency domain data $F'(w)$,

$$F'(w) = F(w) \cdot W \quad (3)$$

After the transformation, the signal is concentrated on the azimuth of the target, which minimizes interference from other directions. We convert the weighted target frequency domain data $F'(w)$ into time domain data $f(t)$ through the inverse Fourier transform, as shown in Figure 3c.

$$f(t) = \mathcal{F}^{-1}[F'(w)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} F'(w) e^{j\omega t} dw \quad (4)$$

Since the signal duration of each beamforming is 1 s, in order to fully include the target echo signal, the integration duration is set to 3 s.

$$F(t) = \sum f(t) \quad (5)$$

Finally, we use short time Fourier transform (STFT) on the integrated target time domain data to generate the target time-frequency image.

$$STFT_F(t, w) = \int_{-\infty}^{\infty} [F(t)g^*(t-u)] e^{-j2\pi w t} dt \quad (6)$$

In Formula (6), $g^*(t-u)$ is the window function. The time-frequency image of the target echo signal is shown in Figure 3d. In the time-frequency image, the horizontal axis is time and the vertical axis is frequency. Through the Formulas (1)–(6), we fuse the target position information into the time-frequency image, which contains the change characteristics of the target echo signal in the frequency domain over time, and can be used to distinguish different targets and identify the position of the target.

In view of the shortcomings of fewer underwater acoustic target samples, two methods are used to increase the number of samples in the data set. The first is to offset the target time-domain data $Y(t)$ starting integration time t by a small amount, and the second is to offset the azimuth θ to change the weighting matrix W . We use these two methods to obtain more samples.

According to the target category, we label the time-frequency image of target samples and store it in the data set. In the data set, all samples contain the target's echo data, distance, and azimuth information. The data set will be used to train the deep convolutional neural network model.

2.2. The Structure of the DCNN Model

In this article, we input the target data set into deep convolutional neural network (DCNN) for training. The basic CNN consists of three structures: convolution, activation, and pooling [23]. DCNN is usually composed of multiple above-mentioned structures connected before and after and adjusted within the layer. The three key features of CNN are the local acceptance area, weight sharing and downsampling process, which effectively reduces the number of network parameters and alleviates the over-fitting problem of the model [24]. Convolution is the most basic and most important level. Convolution operation can extract the features of the image [25]. Through the convolution operation, certain features of the original signal can be enhanced and noise can be reduced. Pooling layers can reduce the amount of data processing while retaining useful information, and sampling can obfuscate the specific location of features [26]. Pooling layers are generally divided into mean pooling and maximum pooling. The advantages of CNN are sharing the convolution kernel, no pressure on high-dimensional data processing, no need to manually select features, and training the weights, that is, the feature classification effect is good. The disadvantages are the need to adjust parameters, the need for a large sample size,

and training is best to use GPU. According to the characteristics of the target time-frequency image, we design a DCNN model. Figure 4 shows the model structure.

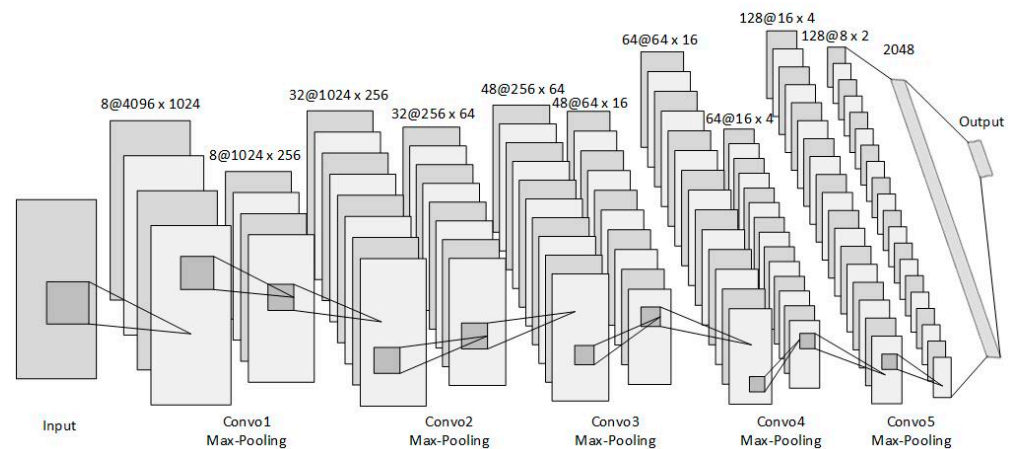


Figure 4. Schematic diagram of DCNN structure. After five convolutional pooling of the input data, the final output is generated through a fully connected layer.

There are in total five convolutional layers in the DCNN model we designed, a fully connected layer and an output layer. As shown in Formula (7), The activation function we use is rectified linear unit (ReLU), and its function is to perform nonlinear mapping on the output of the convolutional layer. It is characterized by fast convergence and simple gradient finding, which can prevent the gradient from disappearing.

$$ReLU(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (7)$$

Figure 5 shows the flow of the input data. The data flows into the ‘conv1’ convolution layer through the input layer, and then into the pooling layer. In this layer, there are eight convolution kernels.

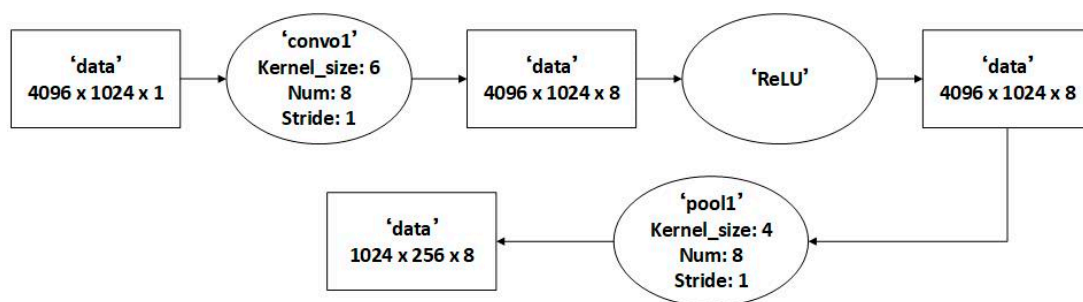


Figure 5. Conv1 convolutional layer process.

The size of the input data is 4096×1024 , the size of the convolution kernel is 6×6 , and the padding is set to “same”, the output feature map has the same size as the input data, and the output feature map has eight dimensions. We use ‘ReLU’ as the activation function. The convolution kernel of the pooling layer is 4×4 , the step is 1, and the image generated by the feature mapping after pooling is $1024 \times 256 \times 8$, that is, the dimension is 8 and the size is 1024×256 .

The data flow of the ‘conv2’ convolutional layer of the DCNN model is shown in Figure 6. The ‘conv2’ convolution layer takes the feature map output by the ‘conv1’ convolution layer as input, and the size is $1024 \times 256 \times 8$. The size of the convolution kernel of the convo2 convolution layer is 5×5 , the number is 32, and the output feature map is $1024 \times 256 \times 32$. The activation function also uses ‘ReLU’. The convolution kernel

of the pooling layer is set to 4×4 , the step size is 1, and the image generated by the feature map after pooling is $256 \times 64 \times 32$, that is, the dimension is 32 and the size is 256×64 .

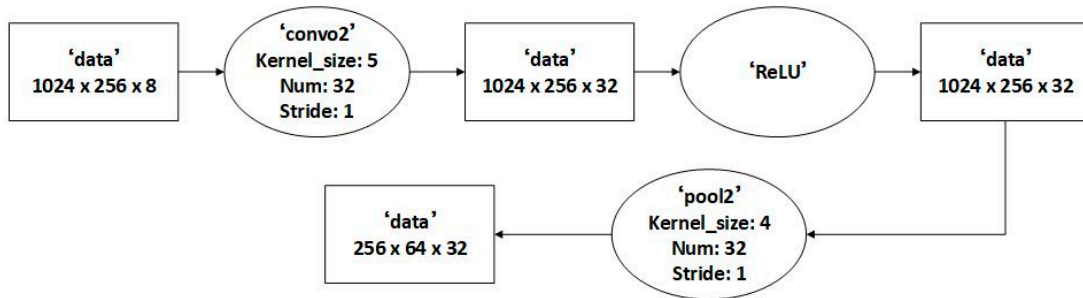


Figure 6. Convo2 convolutional layer process.

As shown in Figures 7–9, the data from ‘convo2’ passes through the ‘convo3’ convolution layer and the ‘convo4’ convolution layer, and then enters the ‘convo5’ convolution layer.

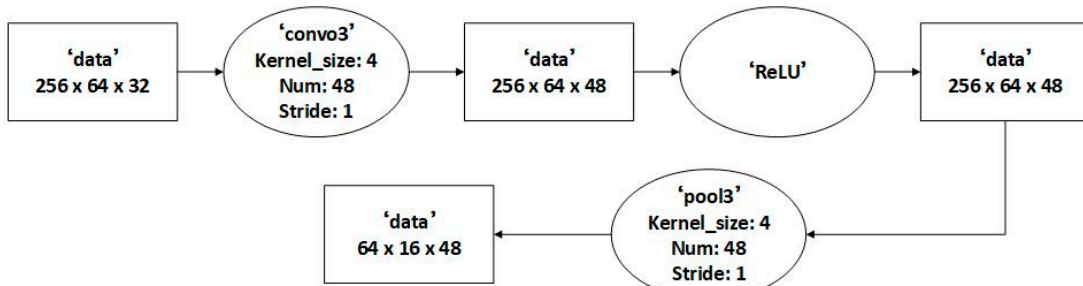


Figure 7. Convo3 convolutional layer process.

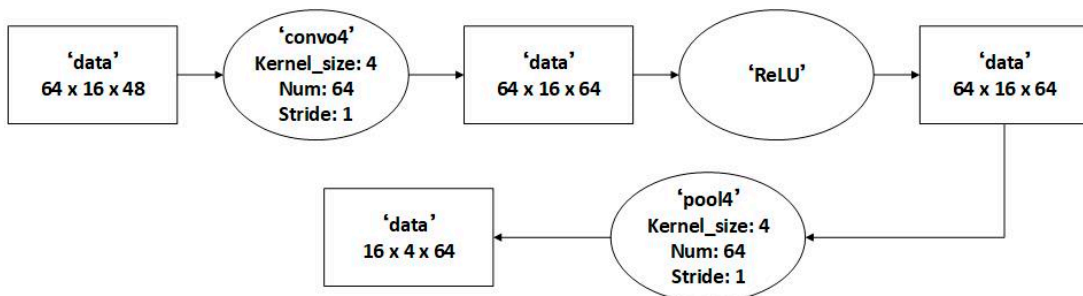


Figure 8. Convo4 convolutional layer process.

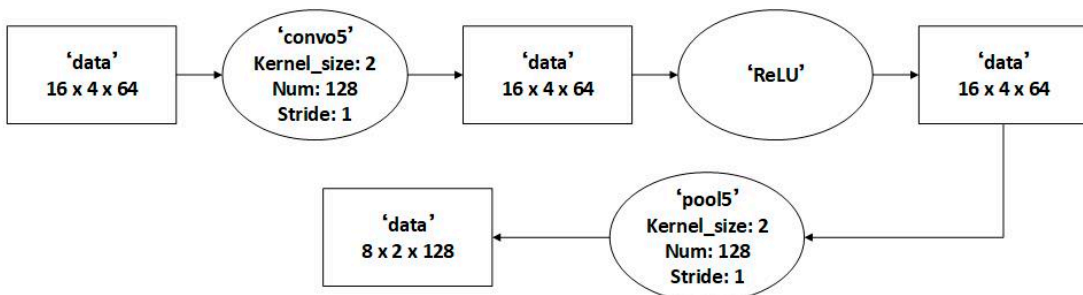


Figure 9. Convo4 convolutional layer process.

The parameters of the convo5 convolutional layer are as follows, the input is $16 \times 4 \times 64$ feature input, the activation function 'ReLU', after convolution pooling, the output is $8 \times 2 \times 128$.

The fully connected layer is used to "flatten" the input, that is, to make the multidimensional input one-dimensional. It is often used in the transition from the convolutional layer to the fully connected layer. The fully connected layer has 2048 neuron nodes and is connected to the convo5 convolutional layer. The output layer uses the 'Softmax' function as the classifier. According to the number of tracking targets, set the corresponding number of nodes. The output y_1, y_2, \dots, y_n from each node in the previous layer is used as the confidence level to generate a new output. The incentive function of each node is *softmax*,

$$\text{softmax}(y)_j = \frac{e^{y_j}}{\sum_{j=1}^n e^{y_j}} \quad (8)$$

The output of the Softmax function is between 0 to 1, and the sum of the output values is 1, that is,

$$\sum_{j=1}^n e^{y_j} = 1 \quad (9)$$

According to the Formula (9), the probability that a certain target exists on the beam azimuth can be calculated. After the DCNN model is built, the target data set can be input into the model for training, and the trained model can be used to identify the target.

2.3. Tracking Process

In this article, we simplify the sonar working environment to two-dimensional plane observation. In the early stage of tracking, we obtain the target's motion state including distance r and azimuth θ through initial measurement, convert the target's information from polar coordinate form (r, θ) to Cartesian coordinates (x, y) , and establish the target state equation and motion equation.

$$X_{k+1} = FX_k + Gw_k \quad (10)$$

$$Z_k = HX_k + \vartheta_k \quad (11)$$

In Formula (10), X_k is the target motion state matrix in period k , F is the state transition matrix, and G is the noise driving matrix. w_k and ϑ_k are uncorrelated white noise with zero mean, and their variance matrices are Q and R respectively. w_k is the input noise and ϑ_k is the observation noise. H is the observation matrix, and Z_k is the corresponding observation signal matrix $\{Z_1, Z_2, \dots, Z_k\}$.

According to the existing data, the velocity components v_x and v_y of the target in the x and y directions are calculated. The position and velocity matrix of the target can be expressed as

$$X = [x, v_x, y, v_y] \quad (12)$$

In the follow-up tracking process, the trained model is used to identify the azimuth-weighted time-frequency image of the echo signal. In order to reduce the amount of calculation, Kalman filtering is used to predict the motion state of the target. As shown in Formula (13), the minimum variance estimated value \hat{X}_k obtained from this observation is used to predict the motion state X_{pre} of the target in the next cycle.

$$X_{pre} = F\hat{X}_k \quad (13)$$

Formula (14) is the prediction covariance matrix, and P_k is a quantitative description of the pros and cons of the prediction quality.

$$P_k = FP_{k-1}F^T + GQG \quad (14)$$

Formulas (13) and (14) describe the time update process of Kalman filtering. According to the obtained target motion state prediction information, the arrival time t of the target echo and the azimuth angle θ of the target are calculated. Then we intercept the echo signal matrix near the predicted time ($t \pm 1$) s, take the predicted azimuth as the center ($\theta \pm 5^\circ$), and generate the time-frequency image of the signal related to the azimuth and the time to be detected according to Formulas (1)–(6) in Section 2.1. Part of the time-frequency image of the signal to be identified is shown in Figure 10.

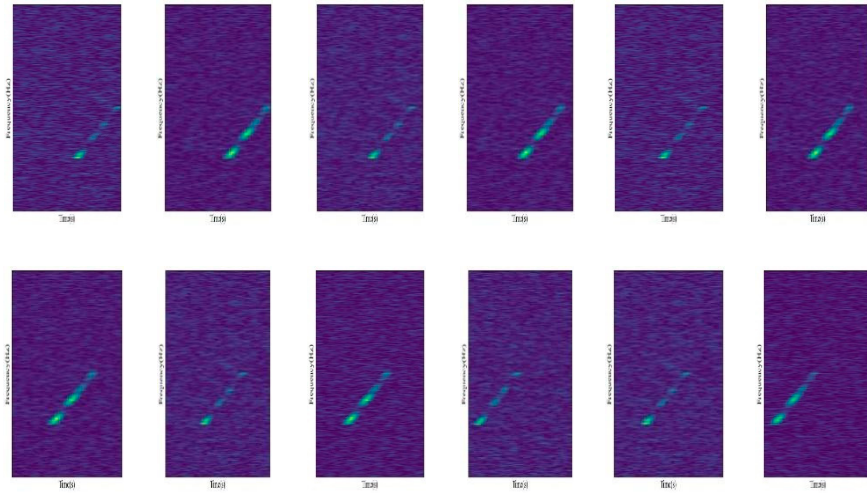


Figure 10. Time-frequency image of the signal to be identified, based on the Kalman filtering and calculating the filtered target state.

We use the trained model to recognize the time-frequency image of the signal to be detected, obtain the azimuth and distance (r_k, θ_k) of the target in this tracking period, and convert it to Cartesian coordinates (x_k, y_k). The Kalman filter is used to correct the deviation of the initial motion state Z_{k+1} of the target obtained in this observation. The filter gain is calculated before correction

$$K_{k+1} = P_{k+1}H^T [HP_{k+1}H^T + R]^{-1} \quad (15)$$

Then calculate the filtered target state

$$\hat{X}_{k+1} = X_{pre} + K_{k+1}(Z_{k+1} - HX_{pre}) \quad (16)$$

Finally update the covariance matrix for the calculation of the next cycle

$$P_{k+1} = [I_n - K_{k+1}H]P_k \quad (17)$$

According to Formulas (15)–(17), the Kalman filter is used to predict the azimuth and distance of the target during the tracking process, and the deviation is corrected to obtain more accurate target state information. Finally, the target status information is obtained in each cycle is updated in the tracking system to complete the tracking.

3. Simulation and Verification of Real World Signal

3.1. Setting of Simulation Signal

We use the bright spot echo model [27] to construct the echo signal of the active sonar, and preprocess the echo signal to generate a data set of the simulated target. The bright spot echo model is often used in the simulation of active sonar echo signals [28], which saves time and improves efficiency in technical verification.

The transmitted signal will echo when it encounters the target. According to the target bright spot model theory, in the case of high frequency, the echo of a complex target is

formed by the superposition of several wavelets. Each wavelet can be regarded as a wave emitted from a certain scattering point, and this scattering point is a bright spot. It can be a real bright spot or an equivalent bright spot. The echo signal of a single bright spot target can be expressed as

$$H(\vec{r}, w) = A(\vec{r}, w) e^{j(w\tau + \varphi)} \quad (18)$$

In Formula (18), $A(\vec{r}, w)$ is the target scattering intensity factor, which is related to frequency, and narrowband signals can take the center frequency value. τ is the delay factor that determined by the sound path ζ of the equivalent echo center point relative to a reference point. $\tau = 2\zeta/c$, c is the sound speed. φ is the phase factor, which is the phase jump generated when the echo is formed. The underwater complex target echo can be regarded as the result of the superposition of several independent bright spot signals. When the number of target bright spot echoes is N , the LFM signal is used as the transmit signal, and the total target echo signal $S(t)$ after superposition can be expressed as,

$$H(\vec{r}, w) = \sum_{n=1}^N A_n(\vec{r}, w) e^{j(w\tau_n + \varphi_n)} \quad (19)$$

The echo of the simulated target is composed of a set of three different parameters, $A_n, \tau_n, \varphi_n (n = 1 \dots N)$. The echo signal received by the receiving array of the active sonar, in addition to the echo and environmental noise reflected by the target, will also receive the scattered waves generated by the random scatterers in the ocean to the emitted acoustic signal and the seabed reverberation. In the simulation of this article, we ignored these effects. Therefore, when the signal is used as the transmission signal, the time-domain form of the target echo signal received by a single array element can be expressed as,

$$S(t) = \sum_{i=1}^P A_i s(t - \tau_i) e^{-j\varphi_i} \quad (20)$$

Then the signal matrix received by the array with the number of elements M is

$$Y(t) = [S_1(t), S_2(t), \dots, S_M(t)] \quad (21)$$

In the simulation experiment, we set up a simulated underwater acoustic environment and set Gaussian white noise as the background noise of the marine environment. Active sonar can only work normally and recognize the target when the difference between the received signal level and the background interference level is greater than or equal to the detection threshold of the device. In this article, we set the active sonar transmitting transducer and the receiving array at the same place, and the environmental noise is isotropic background interference. The SNR of the active sonar received signal is,

$$DT = SL - 2TL + TS - (NL - DI) \quad (22)$$

In Formula (22), SL is the emission sound source level, TL is the transmission loss from the transmitter to the target, TS is the target intensity of the target, the receiving directivity index of the receiving array is DI , the detection threshold of the sonar processing device is DT , the background interference is environmental noise, and its sound level is NL within the working bandwidth of the device.

We use linear frequency modulation (LFM) as the transmitting signal of active sonar. The LFM signal can not only improve the anti-interference ability and target recognition efficiency, and more effectively carry out underwater target detection, but also the time-

frequency image of its echo signal is suitable for convolutional neural network training and recognition. The time function of the LFM signal can be expressed as,

$$S(t) = \begin{cases} Ae^{j2\pi(f_0t + \frac{1}{2}kt^2)} & -T/2 \leq t \leq T/2 \\ 0 & \end{cases} \quad (23)$$

In Formula (23), A is the amplitude of the LFM signal, f_0 is the center frequency, k is the frequency change rate of the signal, and the pulse width of the signal is $t = [-T/2, T/2]$.

We set the transmission frequency modulation signal $f_0 = 2500$ Hz, $k = 2500$ Hz/s, and the transmission duration of the transmission signal is 1 s. The time-domain waveform and frequency-domain waveform of the transmitted signal are shown in Figure 11.

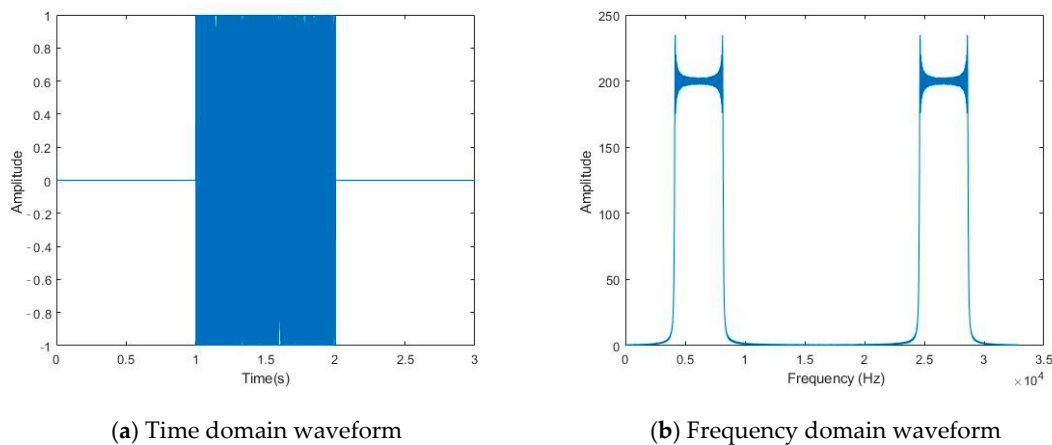


Figure 11. Simulation of time domain and frequency domain waveforms of the transmit signal.

After transmitting the signal, active sonar waits for 1 s to receive the signal, and the receiving time is 20 s. Waiting for 1 s reduces the effect of reverberation.

3.2. Simulation Target Data Set Generation and Model Training

We complete the programming under the TensorFlow 2.0 framework. The neural network model built by TensorFlow 2.0 can realize cross-platform model deployment and is more flexible than TensorFlow 1.0. TensorFlow 2.0 has certain requirements for hardware configuration. We use GPU for model training and use a small server in the laboratory to complete the above work. The graphics card is configured with two Tesla T4s and the GPU memory is 2×15 Gb.

In the simulation, according to the data set generation method given in Section 2.1, we generated a data set of moving targets with different echo characteristics. The number of samples for each target in the data set is 1800, and samples of ocean background noise are generated at the same time, the number is 1500, and the total number of samples in the data set is $(1800 \times \text{num} + 1500)$, where num is the number of tracking targets. Some sample images in the data set are shown in Figure 12. The order of the samples is randomly shuffled.

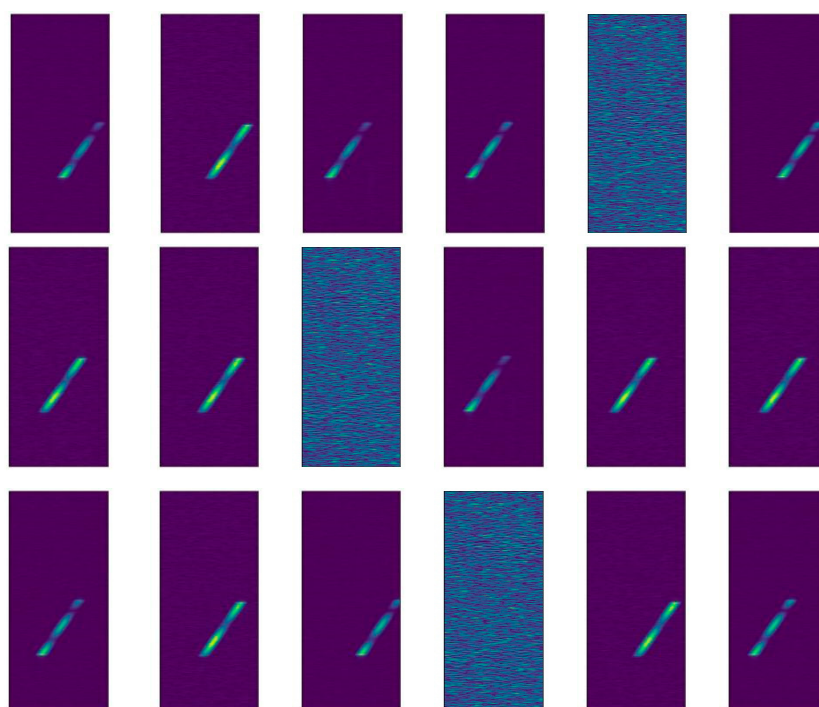


Figure 12. Partial samples of the data set. The data set contains echo signals of two different targets and ocean background noise.

Then the data set is input to the DCNN model designed in Section 2.2 for training, and the accuracy of the model is tested. When training the model, 60% of the data in the data set are used as the training sample, and 40% are used as the test sample. Part of the training parameters are set as follows, the batch size is 40, the training optimizer is “Adam”, and the loss function is the multiclassification loss function “categorical_crossentropy”. The accuracy and loss of the DCNN model after 50, 100, and 200 trainings are shown in Figures 13–15.

In Figure 13, after the DCNN model is trained 50 times, the accuracy rate rises to about 0.95, and the loss decreases to about 0.26.

In Figure 14. After the DCNN model was trained 100 times, the accuracy rate gradually increased to 0.97, and the loss decreased to 0.18.

In Figure 15, after the DCNN model was trained 200 times, the accuracy rate increased to 0.986 and the loss decreased to 0.06. Through training and testing, it is shown that the DCNN model designed in this paper can efficiently learn the characteristics of the target echo signal, and the model recognition accuracy rate is high.

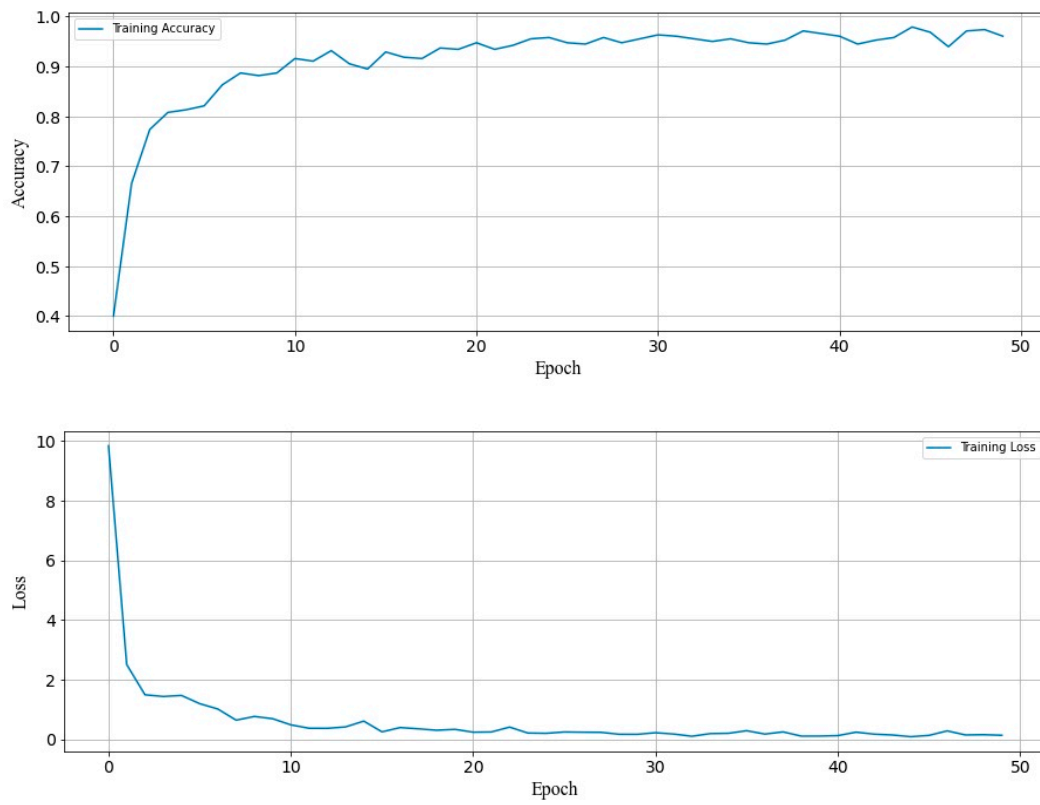


Figure 13. Accuracy and loss after training 50 times.

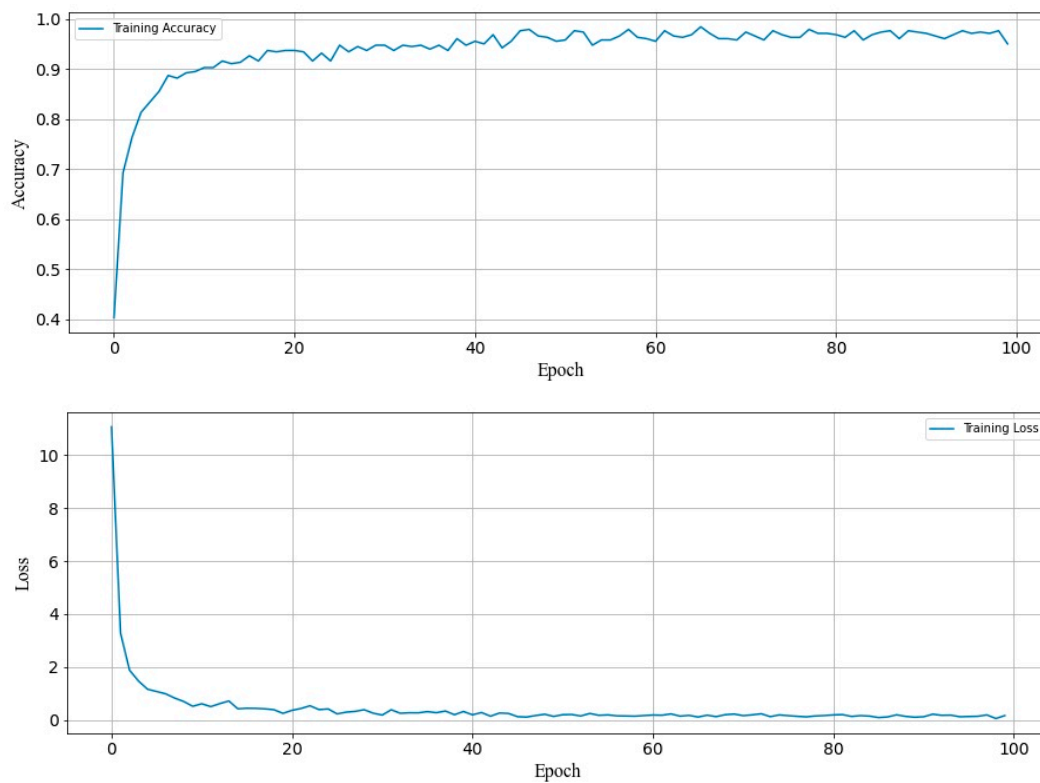


Figure 14. Accuracy and loss after training 100 times.

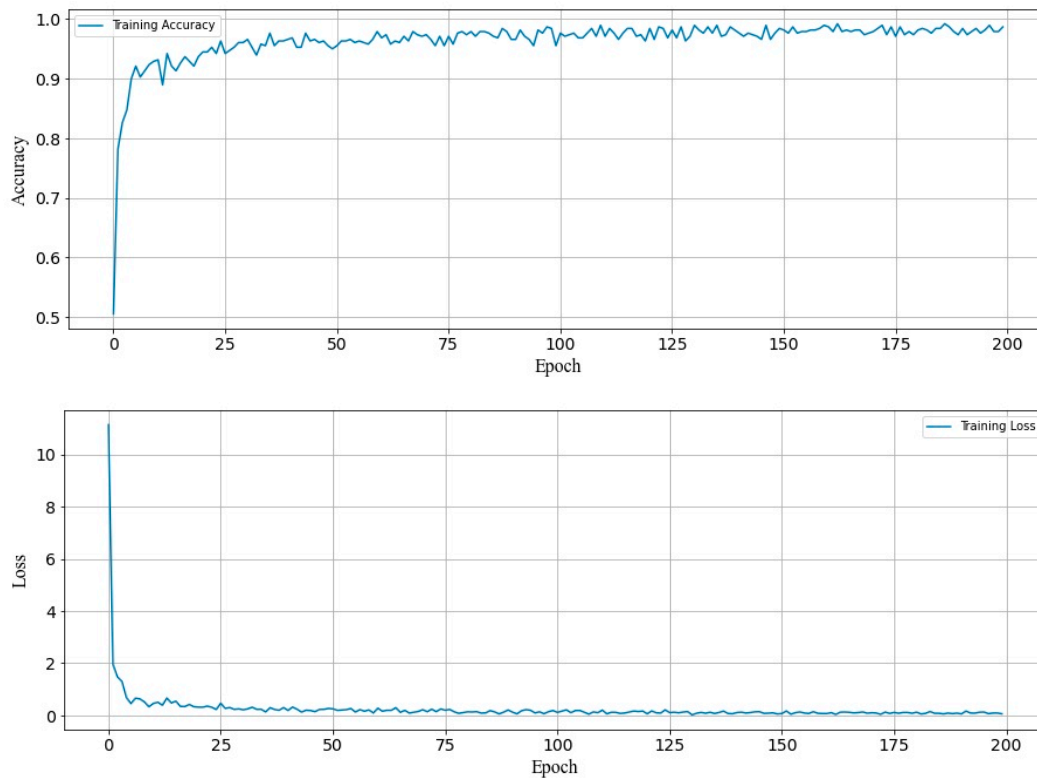


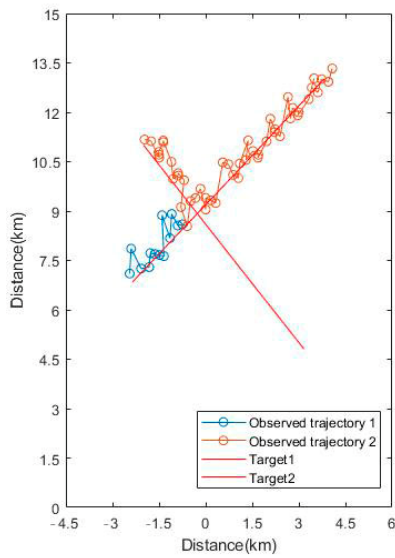
Figure 15. Accuracy and loss after training 200 times.

3.3. Simulation

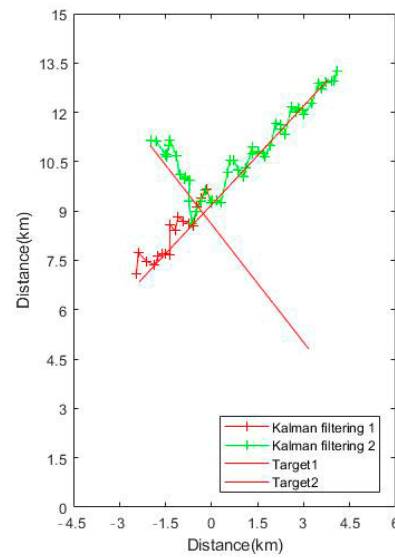
In the simulation verification, the DCNN-KFAT method proposed in this paper is tested as follows: (a) the tracking effect after the azimuth of two moving targets crossed; (b) the tracking effect on a weak target. The test results of the new method are compared and analyzed with the KFAT method.

In this article, we first test the distinguishing ability of the proposed method for different targets and target data association ability during active tracking, and set up a simulation environment containing two moving targets for verification. The two targets have different bright spot echo models, and the time-frequency image of the echo signal can reflect the characteristic information of the target. Figure 7 shows the simulation results of the KFAT method and the DCNN-KFAT method, the coordinate axis unit is km. In the Cartesian coordinate system, the initial position of target 1 is near $(-4.5 \text{ km}, 5 \text{ km})$, moving in the direction away from the active sonar, target 2 is near the initial position $(-2 \text{ km}, 11 \text{ km})$, moving in the direction close to the active sonar.

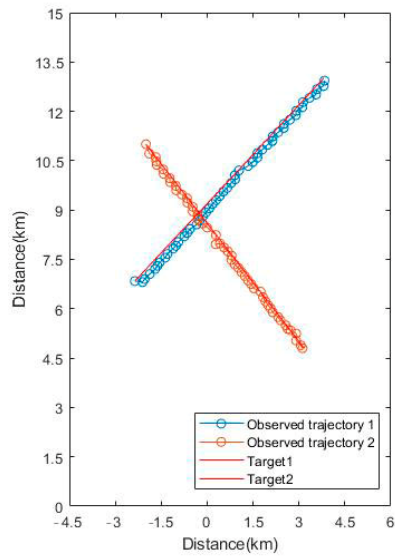
Figure 16a,b are the simulation results of the KFAT method. We represent the simulated motion trajectory of the target, the observation result, and the result after Kalman filtering in Cartesian coordinates. The KFAT method loses a target after the azimuths overlap. Figure 16c,d are the results of using the DCNN-KFAT method. The tracking result is more accurate than the KFAT method. After the azimuth overlaps, the two targets are successfully identified and can continue to be tracked.



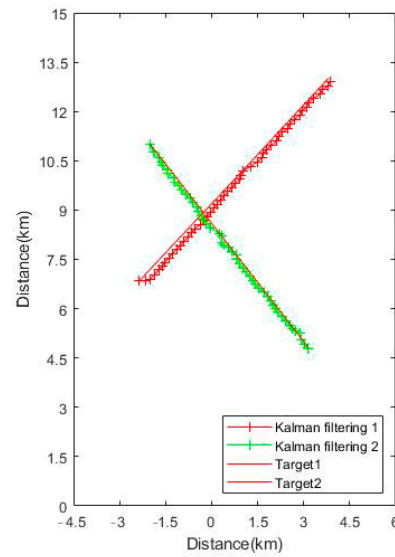
(a) DCNN-KFAT method. After two different targets overlap in azimuth, one of the targets is lost.



(b) KFAT method. The target position accuracy is improved after filtering.



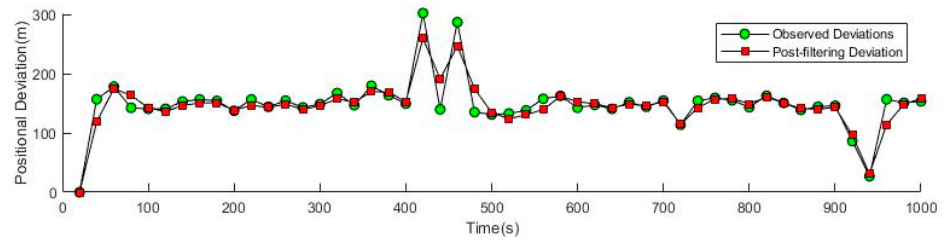
(c) DCNN-KFAT method successfully tracked the target after overlapping the azimuth of two different targets.



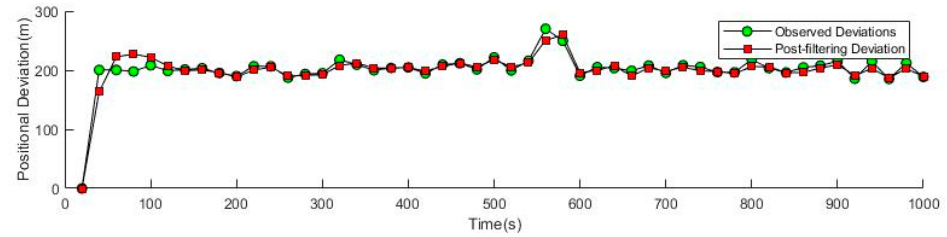
(d) DCNN-KFAT method. The target position accuracy is improved after filtering.

Figure 16. Tracking two azimuth crossing targets.

Figure 17a,b show the deviation of the DCNN-KFAT method for active tracking of two targets. The observation deviation of the DCNN-KFAT method is about 150~300 m, and it is stable at about 200 m. Among them, the deviation of target 1 increases to 300 m in the 400~500 s time period. Compared with their positions, in this time period, the two targets are very close, their azimuth angles are very close, and the distance from the receiving array is almost the same. When the position of the target is very close, since the echo signals of the active sonar are similar in the frequency domain, the deviation is likely to increase.



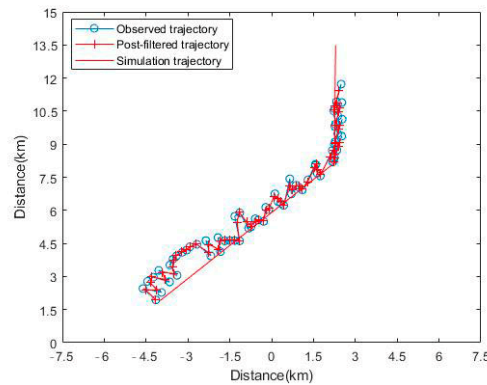
(a) Target 1—DCNN-KFAT. Target 1 position deviation before and after filtering.



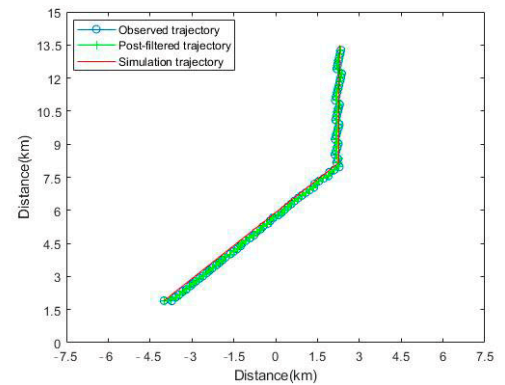
(b) Target 2—DCNN-KFAT. Target 2 position deviation before and after filtering.

Figure 17. Using the DCNN-KFAT method, the deviation is around 200 m.

In order to accurately test the effect of the DCNN-KFAT method in tracking low SNR signals, we continue to set up simulated moving targets for simulation testing. Figure 18 shows the results of the simulation test. The coordinate axis represents the distance in km. The red solid line in Figure 9 represents the set trajectory of the simulated target, and the target moves away from the active sonar. Figure 18a shows the observation results and filtered results using the KFAT method. As the target moves, the intensity of the target echo signal received by the array gradually decreases; when the echo signal intensity is lower than the detection range, the active sonar cannot continue to track the moving target.



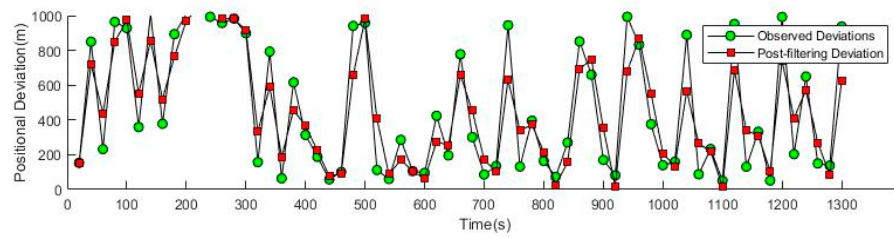
(a) KFAT. Simulation of the motion trajectory of the target before and after filtering.



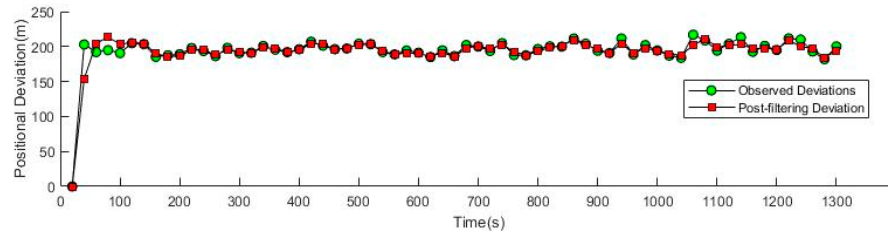
(b) DCNN-KFAT. Simulation of the motion trajectory of the target before and after filtering.

Figure 18. The target SNR gradually decreases and DCNN-KFAT tracks further than KFAT.

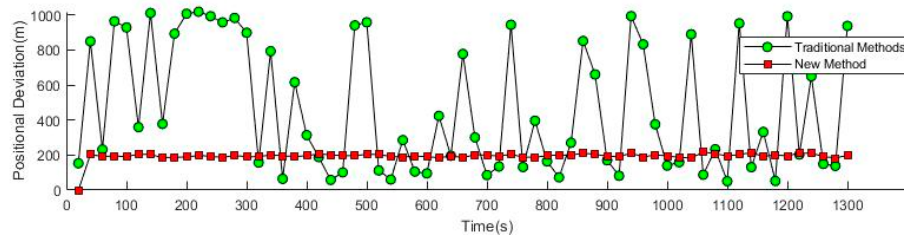
Figure 18b shows the observation results and filtered results using the DCNN-KFAT method. When the KFAT method is lost, the DCNN-KFAT method can still identify the distance and azimuth of the tracked target from the echo signal, and continue tracking the goal. In the early stage of tracking, the deviation of KFAT method is relatively large, while the deviation of DCNN-KFAT method is relatively small. We specifically analyzed the observation bias of the two methods, and the results are shown in Figure 19.



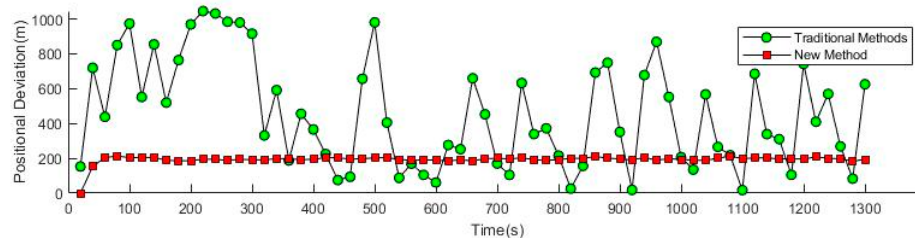
(a) Comparison of KFAT method deviation before and after filtering.



(b) Comparison of DCNN-KFAT method deviation before and after filtering.



(c) Comparison of observation deviation between KFAT and DCNN-KFAT.



(d) Comparison of KFAT and DCNN-KFAT deviation after filtering.

Figure 19. The deviations of KFAT and DCNN-KFAT before and after filtering are listed separately, and the deviation of DCNN-KFAT is significantly smaller than that of KFAT, with an improvement of 500~700 m.

Figure 19a shows the comparison of the tracking deviation of the KFAT method before and after filtering. We can see that the observation deviation range of the KFAT method is about 100~1000 m. In the initial tracking stage, the deviation is large, between 800~1000 m. After 300 s, that is, after 15 transmission cycles, the minimum observed deviation is around 100 m and the maximum is around 900 m. The deviation range of the Kalman filtering result is reduced to within 700 m, and as the tracking continues, the deviation range is gradually reduced to within 400 m.

Figure 19b shows the deviation comparison of the DCNN-KFAT method before and after filtering. We can see that the observation deviation is stable within 200 m, and there are slight fluctuations in the tracking process.

Figure 19c shows the comparison of the observation bias of the two methods. The deviation of the KFAT method fluctuates sharply. Between 100 and 1000 m, the observation bias of the DCNN-KFAT method is stable at about 200 m. Obviously, the stability of the DCNN-KFAT method is better. Through analyzing the data, it is found that the observation

deviation is mainly caused by the estimation deviation of the echo arrival time and the azimuth angle. The deviation caused by the azimuth angle increases with the increase of the distance. Compared with the KFAT method, the DCNN-KFAT method greatly reduces the observation bias and improves the accuracy of the original data.

Figure 19d shows the comparison of the deviations of the two methods after filtering. The deviation of the KFAT method after filtering fluctuates in a larger range, gradually narrowing from 100~1000 m to 150~700 m. The deviation of the DCNN-KFAT method after filtering remains stable at about 200 m.

It can be seen from the above simulation test results that the DCNN-KFAT method proposed in this paper is not only more accurate than the KFAT method in terms of target determination and target data association, but also greatly reduces the observation bias and improves the tracking accuracy.

3.4. Verification of Real World Signal

We further tested the performance of the DCNN-KFAT method using some pre-recorded real-world active sonar signals, with data derived from experimental data from a sea trial in May 2020. We intercepted part of the data containing the two targets whose azimuths overlapped during the motion for processing. The active sonar is set to transmit LFM signals at a frequency of $f = 320\sim 400$ Hz, and the duration of the transmitted signal is 10 s. The echo signal is received by a vertical array of 20 array elements. The time domain waveform and frequency domain waveform of the received target echo signal are shown in Figure 20. As can be seen from Figure 20, in practical applications, it is difficult to discover the target quickly from the time domain or frequency domain only due to the high energy of the low-frequency components in the ambient noise, which can cover the target echo signal.

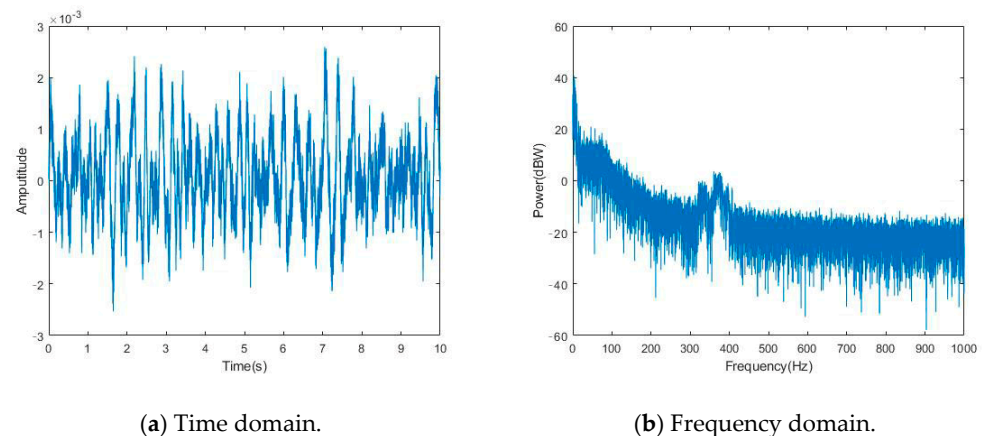


Figure 20. Time domain data and frequency domain data of the target echo signal.

We used the method proposed in Section 2.1 to generate the target echo signal dataset. Some of the data in the dataset are shown in Figure 21.

We processed the data using the DCNN-KFAT method to obtain tracking results and compared them with the KFAT method. The results are shown in Figure 22. The actual motion trajectories of the two moving targets recorded by GPS are shown as black lines.

In Figure 22, for two moving targets with overlapping azimuths, the KFAT method lost one of the targets after the overlap, while the DCNN-KFAT method was able to continue tracking for two different targets after the two targets overlapped in azimuth without the problem of target loss.

In Figure 23, the DCNN-KFAT method is more accurate than the KFAT method, with deviations in the range of 100~200 m, which is within the expected range of the simulation.

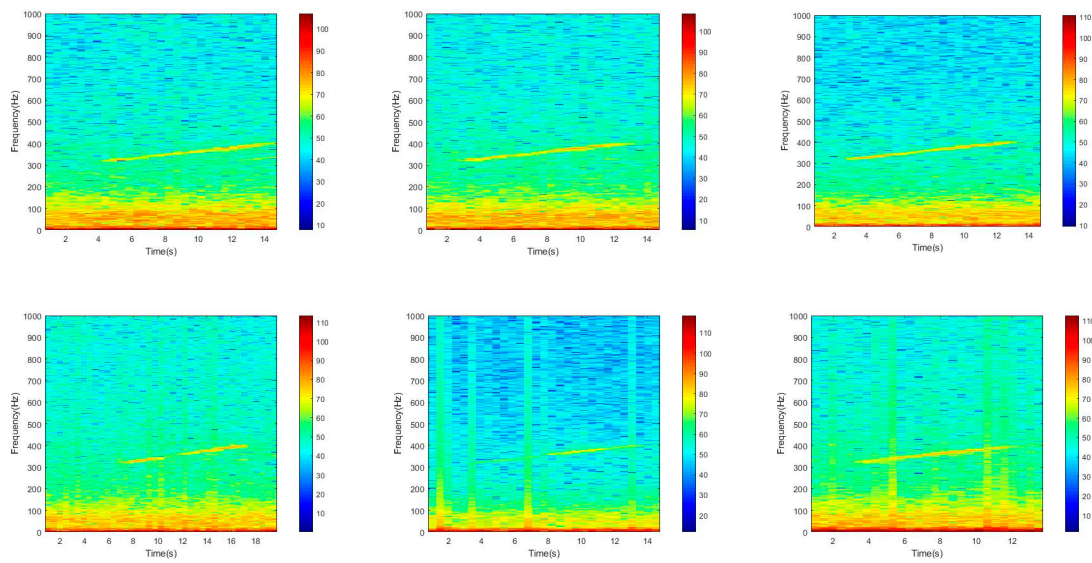


Figure 21. Data set of target echo signals, obtained after processing according to the method in Section 2.1, containing the time-frequency spectrograms of the echo signals of two different targets.

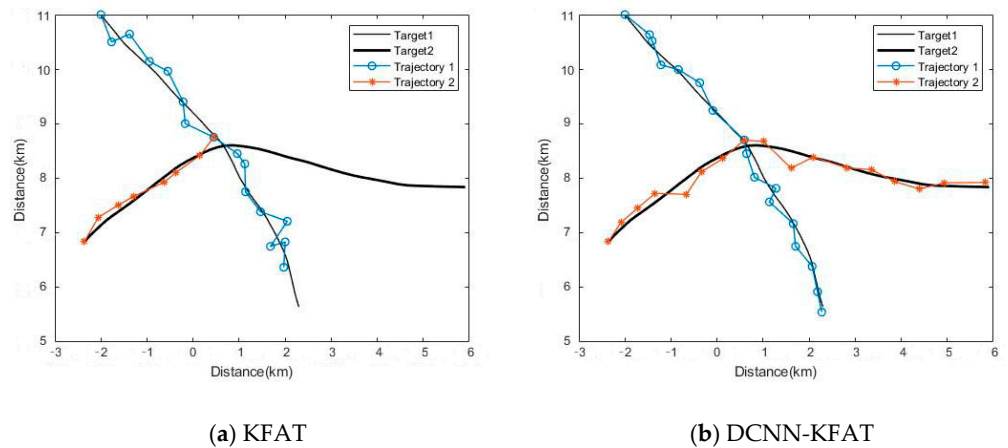


Figure 22. Results of processing the actual sea trial data using both methods. The KFAT method lost the orientation of one target.

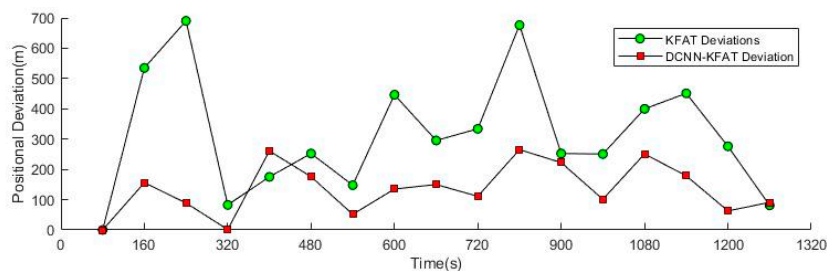


Figure 23. Comparison of the deviation of tracking results for target 1 between the two methods. The DCNN-KFAT method is more accurate than the KFAT method.

The DCNN-KFAT method proposed in this paper is further validated by processing and analyzing the prerecorded real-world signals. The new method is able to continue accurate tracking after two moving targets overlap in azimuth, and has less deviation in tracking accuracy than the KFAT method.

4. Discussion

In active sonar tracking of underwater acoustic targets, the initial measurement information, data association and target judgment of the target affect the tracking performance. In this article, we use DCNN to improve the performance of active tracking of underwater acoustic targets, especially to solve the problem of correctly associating target data after overlapping target azimuths. The samples in the data set are azimuth-weighted time-frequency images, which contain the target's echo feature information and azimuth information. Therefore, when tracking a target, the observation results given by the DCNN-KFAT method include the target's category and azimuth angle, which can accurately correlate the data.

We simulated the moving target using the bright spot echo model of active sonar, performed simulation experiments and validated them using some prerecorded data. We found that the KFAT method uses matched filtering and beamforming methods to determine the time when the target appears through threshold detection, and the deviation range is relatively large. DCNN-KFAT method uses the time-frequency image of the target to judge according to the intensity of the frequency band energy of the target echo signal, and the target arrival time can be calculated more accurately. Due to the limitations of experimental conditions, we only conducted simulation verification, and did not conduct sea trials. In the future, we hope to test in the actual marine environment to verify the performance of the DCNN-KFAT method in practical applications.

5. Conclusions

In this article, we generate echo signals of different underwater acoustic targets based on the bright spot echo model, and generate a data set of simulated target echo signals through weighting processing. The samples in the data set contain echo signal characteristics, target azimuth and distance information. Then we built a DCNN model to learn the echo signal of underwater acoustic targets. We trained and tested the model with a data set of analog signals, and the results showed that the accuracy of the model was high enough to be used for active tracking. Finally, we validate the proposed DCNN-KFAT method with simulations and pre-recorded sea trial data. By analyzing the simulation results, the method has a significant improvement in active tracking and can more accurately distinguish similar different targets. It is simpler and more accurate than the data association and target judgment of the KFAT method. The data recognized by DCNN-KFAT method include target category, target azimuth and target distance. In the process of target data association and target determination, the target data can be correlated very accurately. It solves the problem that KFAT loses a target after encountering the overlap of two target azimuths, and has a significant improvement in tracking accuracy and range.

The research results of this paper can be used for active tracking of underwater acoustic targets and building target datasets for deep learning training and recognition. The DCNN-KFAT method can improve the range and accuracy of tracking, and can solve the data correlation problem in the process of hydroacoustic target tracking, which can be used to improve the engineering application problem of lost targets. The next step will be to test in a real marine environment to verify the performance of the DCNN-KFAT method proposed in this paper in practical applications.

Author Contributions: Conceptualization, M.W. and Z.Z.; methodology, B.Q. and C.Z.; software, B.Q. and H.X.; validation, B.Q., H.X. and C.Z.; formal analysis, C.Z.; investigation, H.X.; resources, M.W. and B.Q.; data curation, H.X.; writing—original draft preparation, B.Q.; writing—review and editing, B.Q., M.W. and C.Z.; visualization, B.Q.; supervision, M.W. and Z.Z.; project administration, M.W.; funding acquisition, M.W. and Z.Z. All authors have read and agreed to the published version of the manuscript.

Funding: The research was funded by the Key Laboratory of Underwater Acoustic Environment, Institute of Acoustic, Chinese Academy of Science and Key R&D Program of Zhejiang Province (No. 2021C03013).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

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

References

1. Liu, G.; Ling, G.; Yan, Q. Review and prospect of active sonar detection techniques. *Tech. Acoustics*. **2007**, *26*, 335.
2. Stewart, J.; Westerfield, E. A Theory of Active Sonar Detection. *Proc. IRE* **1959**, *47*, 872–881. [[CrossRef](#)]
3. Howell, B.; Wood, S. Passive sonar recognition and analysis using hybrid neural networks. *Oceans* **2003**, *4*, 1917–1924. [[CrossRef](#)]
4. Parsons, M.J.G.; Parnum, I.M.; Allen, K. Detection of sharks with the Gemini imaging sonar. *Acoust. Aust.* **2014**, *42*, 185–190.
5. Herkül, K.; Peterson, A.; Paekivi, S.; Sander, P. Applying multibeam sonar and mathematical modeling for mapping seabed substrate and biota of offshore shallows. *Estuar. Coast. Shelf Sci.* **2017**, *192*, 57–71. [[CrossRef](#)]
6. Elfes, A. Sonar-based real-world mapping and navigation. *IEEE J. Robot. Autom.* **1987**, *3*, 249–265. [[CrossRef](#)]
7. Yusof, M.A.B.; Kabir, S. An overview of sonar and electromagnetic waves for underwater communication. *IETE Tech. Rev.* **2012**, *29*, 307. [[CrossRef](#)]
8. Abraham, D.; Gelb, J.M.; Oldag, A.W. Background and Clutter Mixture Distributions for Active Sonar Statistics. *IEEE J. Ocean. Eng.* **2011**, *36*, 231–247. [[CrossRef](#)]
9. van Vossen, R.; Beerens, S.P.; van der Spek, E. Anti-submarine warfare with continuously active sonar. *Sea Technol.* **2011**, *52*, 33.
10. Marage, J.-P.; Mori, Y. *Sonar and Underwater Acoustics*; Wiley: Hoboken, NJ, USA, 2013. [[CrossRef](#)]
11. Duan, Z.; Han, C.; Li, X.R. Comments on “Unbiased converted measurements for tracking”. *IEEE Trans. Aerosp. Electron. Syst.* **2004**, *40*, 1374. [[CrossRef](#)]
12. Lerro, D.; Bar-Shalom, Y. Tracking with debiased consistent converted measurements versus EKF. *IEEE Trans. Aerosp. Electron. Syst.* **1993**, *29*, 1015–1022. [[CrossRef](#)]
13. Lei, M.; Han, C. Sequential nonlinear tracking using UKF and raw range-rate measurements. *IEEE Trans. Aerosp. Electron. Syst.* **2007**, *43*, 239–250. [[CrossRef](#)]
14. Bar-Shalom, Y.; Kirubarajan, T.; Lin, X. Probabilistic data association techniques for target tracking with applications to sonar, radar and EO sensors. *IEEE Aerosp. Electron. Syst. Mag.* **2005**, *20*, 37–56. [[CrossRef](#)]
15. Lo, K.W.; Ferguson, B.G. Automatic detection and tracking of a small surface watercraft in shallow water using a high-frequency active sonar. *IEEE Trans. Aerosp. Electron. Syst.* **2004**, *40*, 1377–1388. [[CrossRef](#)]
16. Yang, H.; Byun, S.-H.; Lee, K.; Choo, Y.; Kim, K. Underwater Acoustic Research Trends with Machine Learning: Active SONAR Applications. *J. Ocean Eng. Technol.* **2020**, *34*, 277–284. [[CrossRef](#)]
17. Nguyen, H.-T.; Lee, E.-H.; Lee, S. Study on the Classification Performance of Underwater Sonar Image Classification Based on Convolutional Neural Networks for Detecting a Submerged Human Body. *Sensors* **2019**, *20*, 94. [[CrossRef](#)]
18. Young, V.W.; Hines, P.C. Perception-based automatic classification of impulsive-source active sonar echoes. *J. Acoust. Soc. Am.* **2007**, *122*, 1502–1517. [[CrossRef](#)]
19. Roads, C.; Nilsson, N. Principles of Artificial Intelligence. *Comput. Music. J.* **1980**, *4*, 64–65. [[CrossRef](#)]
20. Yang, H.; Shen, S.; Yao, X.; Sheng, M.; Wang, C. Competitive Deep-Belief Networks for Underwater Acoustic Target Recognition. *Sensors* **2018**, *18*, 952. [[CrossRef](#)]
21. Yao, X.H.; Yang, H.H.; Li, Y.Q. A method for feature extraction of hydroacoustic communication signals based on generative adversarial networks. In *Proceedings of the 2019 Academic Conference of the Underwater Acoustics Branch*; Nanjing, China, 24–25 May 2019; Chinese Society of Acoustics: Beijing, China, 2019.
22. Zhu, B.; Wang, X.; Chu, Z.; Yang, Y.; Shi, J. Active Learning for Recognition of Shipwreck Target in Side-Scan Sonar Image. *Remote. Sens.* **2019**, *11*, 243. [[CrossRef](#)]
23. Shin, H.-C.; Roth, H.R.; Gao, M.; Lu, L.; Xu, Z.; Nogues, I.; Yao, J.; Mollura, D.; Summers, R.M. Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. *IEEE Trans. Med. Imaging* **2016**, *35*, 1285–1298. [[CrossRef](#)]
24. Shi, W.; Gong, Y.; Tao, X.; Zheng, N. Training DCNN by Combining Max-Margin, Max-Correlation Objectives, and Correntropy Loss for Multilabel Image Classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *29*, 1–13. [[CrossRef](#)]
25. Hou, W.; Wei, Y.; Jin, Y.; Zhu, C. Deep features based on a DCNN model for classifying imbalanced weld flaw types. *Measurement* **2018**, *131*, 482–489. [[CrossRef](#)]
26. Yu, D.; Wang, H.; Chen, P. Mixed pooling for convolutional neural networks. In *Rough Sets and Knowledge Technology, Proceedings of the International Conference on Rough Sets and Knowledge Technology, Shanghai, China, 24–26 October 2014*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 364–375.
27. Wang, M.; Huang, X.; Hao, C. Model of an underwater target based on target echo highlight structure. *J. Syst. Simul.* **2003**, *1*, 21–25.
28. Hao, Y.G.; Zhang, Z.H.; Li, H.F. Active sonar target echo signal modeling techniques. *Command. Inf. Syst. Technol.* **2020**, *11*, 70–75.