ORIGINAL ARTICLE

A method of radar target detection based on convolutional neural network

Wen Jiang[1] · Yihui Ren[1] · Ying Liu[1] · Jiaxu Leng[1]

Abstract

Radar target detection (RTD) is one of the most significant techniques in radar systems, which has been widely used in the field of military and civilian. Although radar signal processing has been revolutionized since the introduction of deep learning, applying deep learning in RTD is considered as a novel concept. In this paper, we propose a model for multitask target detection based on convolutional neural network (CNN), which works directly with radar echo data and eliminates the need for time-consuming radar signal processing. The proposed detection method exploits time and frequency information simultaneously; therefore, the target can be detected and located in multi-dimensional space of range, velocity, azimuth and elevation. Due to the lack of labeled radar complex data, we construct a radar echo dataset with multiple signal-to-noise ratio (SNR) for RTD. Then, the CNN-based model is evaluated on the dataset. The experimental results demonstrated that the CNN-based detector has better detection performance and measuring accuracy in range, velocity, azimuth and elevation and more robust to noise in comparison with traditional radar signal processing approaches and other state-of-the-art methods.

Keywords Radar target detection · Radar signal processing · Deep radar detection · Deep learning models · Convolutional neural network

1 Introduction

Radar echoes returned from targets are usually submerged in noise, clutter or jamming signal. Radar target detection (RTD) is a fundamental but significant process in order to differentiate and measure targets from increasingly complex background. In traditional radar signal processing, effective methods such as matched filtering, Doppler processing and clutter suppression are mostly adopted to improve signal-to-noise ratio (SNR) and signal-to-clutter ratio (SCR) [1], and constant false alarm rate (CFAR) detection is a well-studied and widely used approach which is based on hypothesis testing for detection in demanding noisy environments [2]. However, due to the diversified

development trend of target types and complex detection environment, selecting an optimal parameter set for traditional method is extremely challenging; thus, developing reliable and robust methods for RTD seems to be inevitable [3].

As a subset of machine learning, deep learning methods attempt to create models and extract features automatically from large complex datasets. It has brought a dramatical breakthrough in various domains including image processing, speech recognition, natural language processing, etc. [4–7]. As one of the most important technologies of deep neural network (DNN), currently, convolutional neural network (CNN) is widely used in computer vision tasks [8], such as semantic segmentation [9], image classification [10] and object detection [11]. In the field of remote sensing image segmentation, Wang et al. [12] created an 11-layer CNN for image segmentation of polarimetric synthetic aperture radar. Chen et al. [13] proposed an improved semantic segmentation neural network which adopts dilated convolution, a fully connected fusion path and pre-trained encoder for the semantic segmentation task of high-resolution remote sensing images. Zhang et al. [14]

✉ Wen Jiang
jiangwen19@mails.ucas.edu.cn

[1] School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 101400, China

introduced a neural network architecture search method for sematic segmentation of high-resolution remote sensing images. In the field of medical, an effective segmentation method for skin lesion segmentation is also presented in [15]. For image classification, Öztürk et al. [16] presented an effective CNN-based classification method for gastrointestinal tract images. Zhang et al. [17] designed a 13-layer CNN for fruit category classification. For object detection, Zhang et al. [18] proposed a two-stream contextual CNN to adaptively capture body part information of face. Zhu et al. [19] proposed a contextual multi-scale region-based CNN in face detection.

1.1 Related work

Thanks to the outstanding success achieved by AlexNet [20], VGGNet [21], GoogleNet [22], ResNet [23], etc., DNN architecture has spread to almost all fields, including RTD. In the past decades, many artificial neural network (ANN)-based and DNN-based approaches have been proposed to solve the problem of RTD in various complex scenarios. To differentiate the targets from noise, Gandhi et al. [24] firstly employed ANNs to detect signals in non-Gaussian noise. Amores et al. [25] deduced that the ANN approach could improve radar detector robustness. Rohman et al. [26] presented an adaptive ANN-CFAR detector to improve the performance of RTD in non-homogeneous noise. Akhtar et al. [27] presented an ANN-CFAR detector for fluctuating target detection in noisy background. Compared with noise background, detecting targets within clutter is a more common but challenging task. Cheikh et al. [28] assessed the problem of RTD using different architectures of ANNs in a K-distributed clutter. Akhtar et al. [29] proposed a more general training strategy to extract fluctuating targets in K-distributed clutter. Pan et al. [30] proposed a deep CNN approach for marine small target detection in strong sea clutter background. The above references provided solutions to a binary classification problem, namely differentiating target present or absent.

In addition, researchers tried to utilize multiple types of inputs to deep learning models for RTD, such as pulse–range maps, range–Doppler spectrums, time–frequency images, etc. Pan et al. [30] adopted pulse–range images as inputs of a deep CNN model for RTD. Wang et al. [31] designed a CNN-based target detector on range–Doppler spectrum and compared the proposed method against traditional CFAR detector. Brodeski et al. [32] introduced a CNN-based architecture for automotive radar detection using range–Doppler data to detect and localize targets. Gustavo et al. [33] implemented a time–frequency block by the Wigner–Ville distribution (WVD) and designed a WVD-CNN detector for RTD. Su et al. [34] adopted short-time Fourier transform (STFT) to transform IPIX measured data and designed a CNN-based method for maritime targets detection.

1.2 Motivation and contributions

At present, the difficulty in RTD mainly lies in target high-resolution feature extraction, environment clutter suppression, anti-jamming measures, especially for strong active jamming, "low-slow-small" target detection, etc. [2]. Traditional CFAR-based detection methods are based on statistical theory, which considers the target or environment models as a stochastic process. However, due to the diversified trend of target and environment models, selecting an optimal parameter set and declaring potential objects in complex environment with a high probability of detection (P_D) but a low false alarm rate (P_{fa}) still remains to be an extremely challenging task.

The above studies aiming at the problem of RTD used various forms of input which can be regarded as "two-dimensional images." All inputs were preprocessed by methods of radar echo signal processing. Some researchers have proved that conventional radar signal processing serves as the preprocessing of the training data, which can help to sharpen features, thus improving the detection performance. But in fact, these preprocessing methods, such as pulse compression and Doppler processing, are intrinsically convolution operation, and in nature, deep learning models can automatically extract features from radar complex data. In addition, the primary objective of RTD is not only to distinguish whether the received signal undertested containing echo from the target or just corresponds to the noise and clutter, but also to obtain multi-dimensional location and velocity information. Therefore, realizing a complete "end-to-end" learning scheme for multitask RTD is reasonable and feasible.

In this paper, we propose a novel CNN-based detector for RTD and apply radar echo data directly to locate the target in multi-dimensional space of range, velocity, azimuth and elevation. The proposed approach eliminates the need for time-consuming radar signal processing and presents better detection performance in comparison with the classical radar signal processing methods. However, due to the specificity of RTD tasks, currently, the actual radar dataset is not widely published or accessible. To overcome the lack of labeled radar complex data, we constructed a radar echo dataset with multiple SNR. The shortcoming of our method is that we evaluated our model on the simulated radar dataset and hence it may not perform well on the actual dataset. In the future, we shall try to collect realistic radar data and evaluate and improve our model further.

The main contributions of our work are summarized as follows:

- We propose a Feature Extraction Net which exploits both time and frequency information and extracts range–Doppler information as feature maps from raw radar echo data.
- A multitask target detection method is designed, in which a RD Detection Net is adopted to measure range and radial velocity of target and an Angle Detection Net is used to predict azimuth and elevation.
- In order to perform velocity measurement and angle estimation, we construct a three-channel sum–difference pattern radar echo dataset with multiple SNR for training and testing.

The rest of this paper is organized as follows: Sect. 2 introduces traditional signal processing for RTD, which is often used for performance comparison. In Sect. 3, the proposed CNN-based approach for RTD is described. In Sect. 4, the performance of the proposed model is evaluated in simulation data. Section 5 discusses the challenges and research opportunities. Finally, conclusion is presented in Sect. 6.

2 Traditional signal processing method for RTD

In a typical pulsed coherent radar system, the received multiple coherent pulses echoes are processed in general radar signal processing method including matched filter, clutter suppression, Doppler processing, CFAR detector, etc. Figure 1 shows a traditional radar signal processing procedure of radar detection.

After emission of each pulse, the incoming echoes are sampled with a given rate and a pulse compression is performed through a matched filter operation over fast time domain which could obtain narrow pulse width and high resolution of range profile. MTI and MTD are effective approaches for clutter suppression. Doppler processing is applied over multiple pulses by applying fast Fourier transform (FFT) over slow time at each range cell and thus range–Doppler spectrum can be obtained. Next, the echoes

in the range–Doppler spectrum, whose energy exceeds the detection threshold, are disclosed by the CFAR detector. Then, the position and velocity information can be evaluated.

In target detection, the Neyman–Pearson criterion is widely used for decision making in order to minimize the P_{fa} and maximize the P_D [35]. In a general CFAR detector, the square-law detected range samples are sent serially (cell by cell) into a shift register of length $2n + 1$. The statistic value z, which is proportional to the estimate of total noise power, is formed by processing the contents in $2n$ reference cells surrounding the cell under test (CUT) whose content is y. A target is declared to be present if its energy exceeds the detection threshold αz:

$$\begin{cases} y > \alpha z, & target\ present \\ y \leq \alpha z, & target\ absent \end{cases} \tag{1}$$

Here, a constant scale factor α is used to achieve a desired constant P_{fa} for a given window of size $2n$ when the total background noise is homogeneous. The processor configuration varies with different CFAR schemes. Figure 2 presents the diagram of several typical CFAR processors. Take the CA-CFAR detector as an example, as shown in Fig. 2, consisted of $2n$ cells surrounding the CUT. The average level in the CUT is estimated by taking the arithmetic sum of the reference cells as

$$z_{CA} = \left(\sum_{i=1}^{n} x_i + \sum_{i=n+1}^{2n} x_i \right) \Big/ 2 \tag{2}$$

This average value z_{CA} is scaled by a factor α to yield an adaptive threshold αz and then compared with the value of CUT y to estimate whether the target is present or not.

Radar is always required to operate in a variety of significantly diversified scenes. Selecting a single optimal parameter set is extremely important, because predefined parameters, such as threshold, margin, sizes of the reference and guard windows, would determine the detection performance. Therefore, developing a data-driven deep learning approach for target detection seems to be inevitable and reasonable.
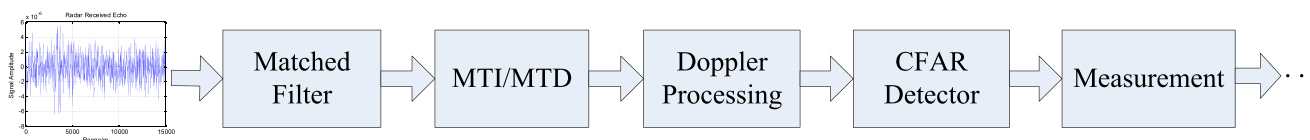
Received Echoes

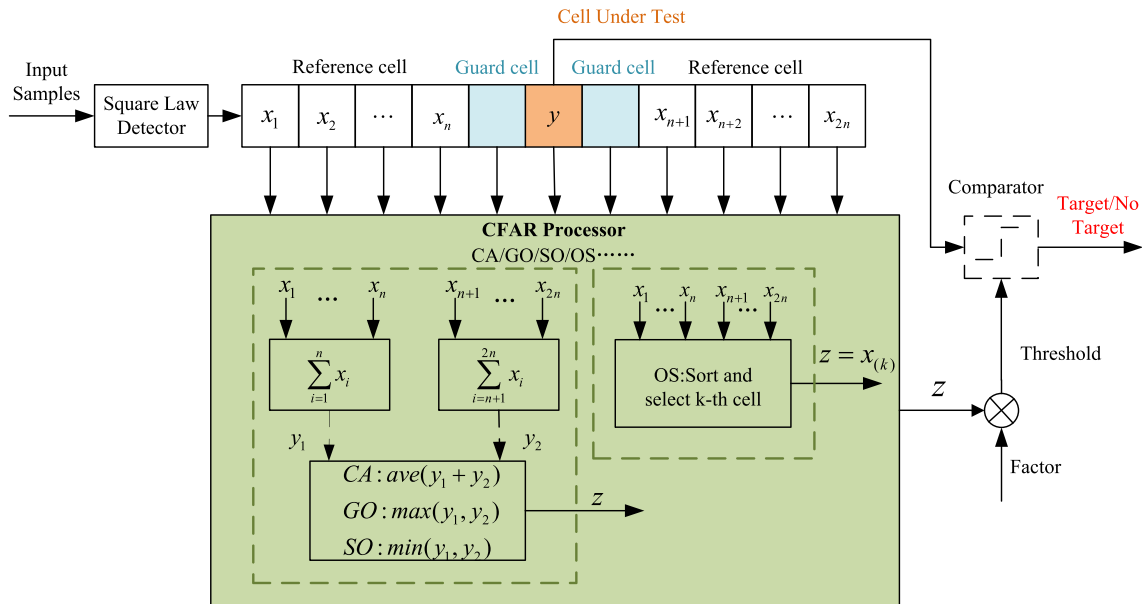**Fig. 1** Typical signal processing diagram of a radar system

Fig. 2 Block diagram of several typical CFAR processors

3 The proposed CNN-based method for RTD

In this section, we present a CNN-based RTD scheme which works directly on the raw radar data to detect target. Radar echo signal is a one-dimensional discrete complex sequence. Firstly, we construct the radar data cube as input of network by preprocessing the one-dimensional discrete echo sequence, which could better reflect the characteristics of samples. Then, a novel CNN-based model for RTD is designed to detect and locate the target in multi-dimensional space of range, radial velocity, azimuth and elevation. The block diagram of the procedure is depicted in Fig. 3, and the various stages are described in the following subsections.

3.1 Processing of input data for RTD

Radar emits multiple pulses in a coherent processing interval (CPI), and the received echoes are integrated together coherently or incoherently, which can be

processed as a one-dimensional discrete sequence. There are many commonly used processing methods for discrete time sequences, which determine the diverse forms of input for detection network. However, compared with other input forms, using radar complex data (real and imaginary) directly can make better use of the original information contained in the echo signal.

We consider using the three-channel sum–difference pattern to construct radar echo data sets, which enables us to measure angle, respectively, in azimuth and elevation plane. The values of sampling points are usually stored in a matrix form. Figure 4 shows the raw radar echo cube, which is collected during the simulation process with the ground truth label of the detection: range, velocity, azimuth angle and elevation angle. Dimensions of the radar echo cube are $Ns \times Np \times Nc$, where Ns denotes the number of sampling points within a range gate, Np denotes the number of pulses in a CPI and Nc denotes the number of received channels. Since echo data are complex number,
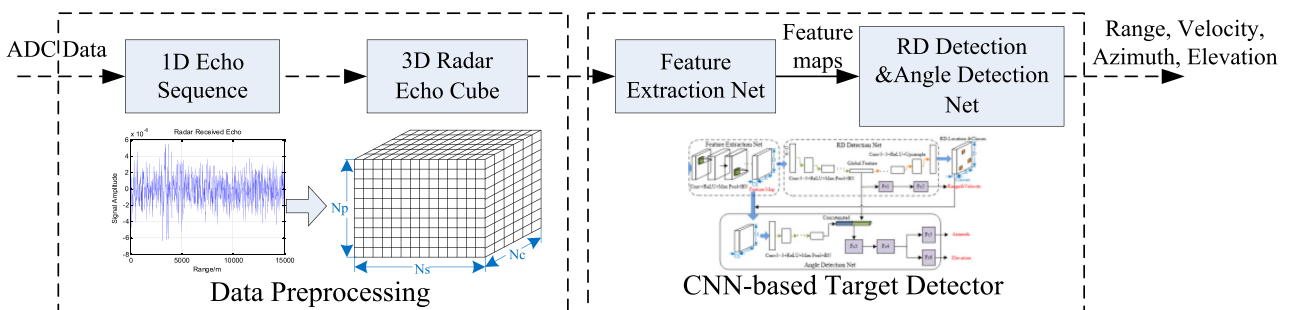

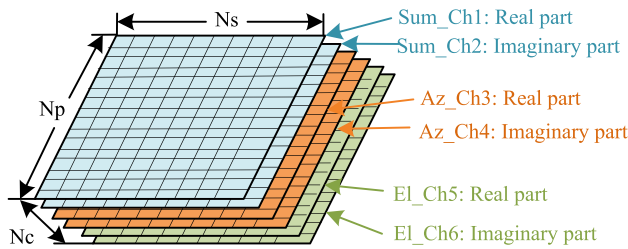**Fig. 3** Block diagram of the CNN-based target detector

Fig. 4 Raw radar echo cube as the input of network

each sample is split into a real part and an imaginary part as the input to the network and thus $Nc = 6$.

We obtain and process suitable two-dimensional samples from the input cube of radar echo data for RTD. In the transverse direction (fast time domain), the sampling points are determined by the pulse repetition frequency and the sampling frequency, while in the longitudinal direction (slow time domain), the pulses in a CPI are listed in order. In classical radar signal processing, echo data in fast time domain are used to calculate range and those in slow time domain which represents Doppler bins are used to calculate velocity. Similarly, we slide the convolution kernels with preset size to extract information in time and frequency domain with rectangular samples as shown in Fig. 5. Rectangular samples may not necessarily be square, but it

also needs to determine the sliding step and the size of data samples according to the actual data.

In fact, processing radar raw echo data as one-dimensional input for classification or regression will greatly reduce the computational complexity of the model, but it will also split the spatial–temporal correlation and even destroy certain texture information of targets. Compared with one-dimensional data, two-dimensional data can make use of temporal and spatial correlation simultaneously, therefore improving discrimination ability and detection accuracy. CNN is good at extracting features automatically from large-scale raw data and classifying two-dimensional graph structure data. It is also necessary to ensure that enough amount of data and enough features are available for detection.

3.2 Design of the CNN-based target detector

The CNN-based detector consists of three parts: Feature Extraction Net, RD Detection Net and Angle Detection Net. Figure 6 shows the proposed structure of CNN-based detector aiming to detect targets in multi-dimensional space and obtain the information of range, velocity, azimuth angle and elevation angle with radar raw data. Detection steps are given as follows:
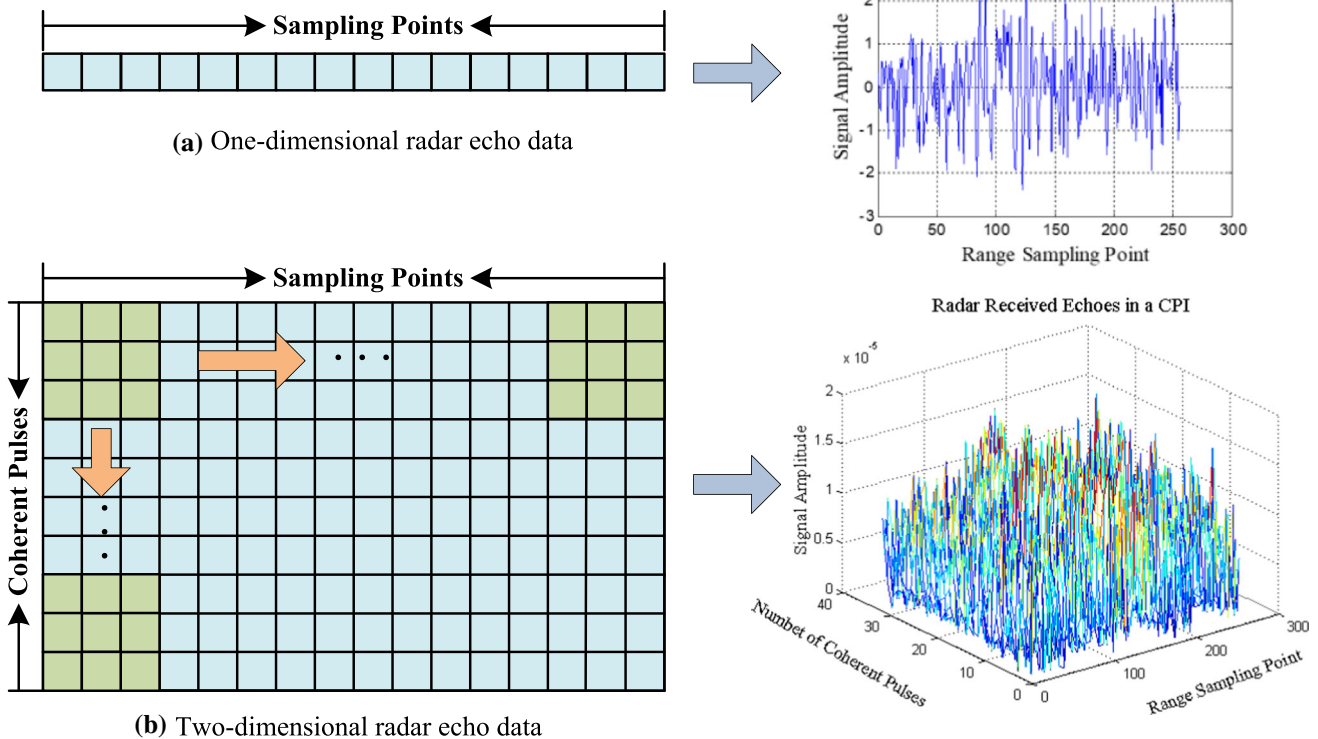

(a) One-dimensional radar echo data

(b) Two-dimensional radar echo data

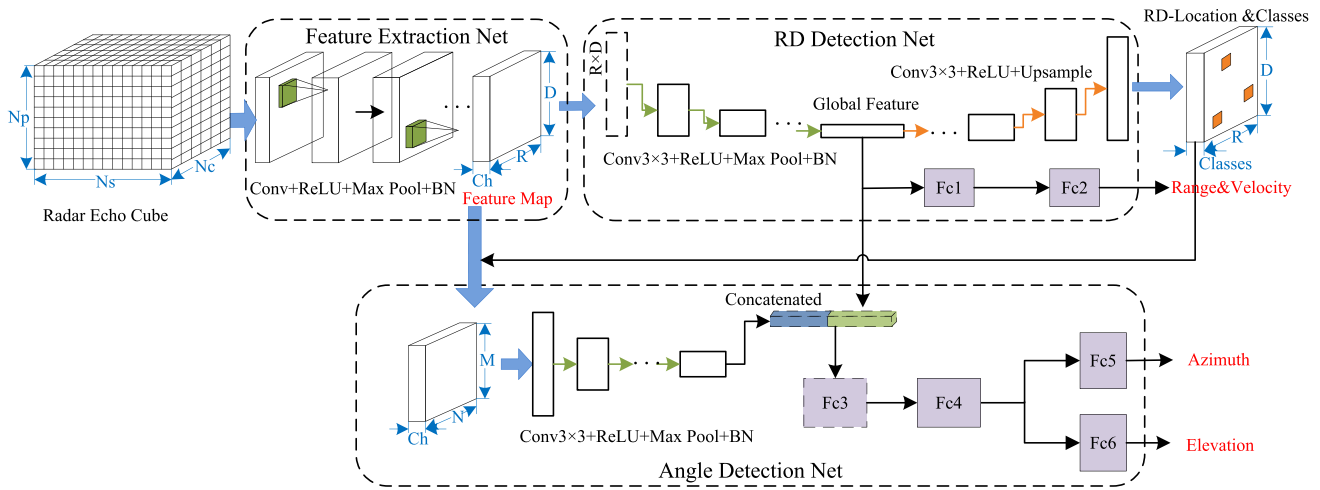Fig. 5 Direct input in two-dimensional form

Fig. 6 Architecture of the proposed CNN detector for RTD

Feature Extraction Net: Extract range–Doppler information as feature maps from raw radar echo data, detection of range, velocity, angles in subsequent network rely on these feature maps. Feature Extraction Net exploits both time and frequency information from raw radar signals by a specially designed CNN. Dimensions of the input are $Ns \times Np \times Nc$, while those of the output are $R \times D \times Ch$. This section consists of multiple convolutional layers. Convolution layer is calculated as:

$$y_i^{k+1} = f\left(\sum_j x_j^k \otimes w_{ij}^k + b_i^k\right) \tag{3}$$

where x represents the input data of the convolutional layer, w represents the weights of convolution operator, b denotes the bias and f denotes the activation function.

The pooling layer has the most significant effect on reducing the size of the input data according to the selected neighborhood values. When dimensions of input data x are $m \times n$, the kernel size is $p \times q$ and the step is t, the max pooling operation is calculated as:

$$P_{ij} = \max(x_{i \cdot t + r, j \cdot t + s}) \quad \begin{array}{ll} r = 0, 1, 2, ..., p-1 & i \le (m-p)/t \\ s = 0, 1, 2, ..., q-1 & j \le (n-q)/t \end{array}. \tag{4}$$

The ReLU layer is calculated as:

$$R(x) = \max(0, x) = \begin{cases} x, & x \ge 0 \\ 0, & x < 0 \end{cases} \tag{5}$$

RD Detection Net: Detect targets and predict range and velocity of all targets in the range–Doppler domain. The input of the RD Detection Net is the range–Doppler–channel data ($R \times D \times Ch$), while the outputs are the range and radial velocity by regression and a list of detections in range–Doppler feature map with their associated classes. A

global feature vector from the middle layer of the RD Detection Net is extracted for the Angle Detection Net. In addition to convolutional layers, upsampling layers are also used in this section in order to regain the same size as original input.

Angle Detection Net: Predict the azimuth angle and the elevation angle of each target detected by the RD Detection Net. The input form of the Angle Detection Net is a cropped region from the range–Doppler feature maps which is centered at the location provided by the RD Detection Net. A global feature vector is concatenated with the convolutional output vector, which contributes to obtain azimuth and elevation of each detection. The outputs of the Angle Detection Net are values of the azimuth angle and the elevation angle by regression.

Configurations of the CNN-based radar target detector are as follows:

1. The Feature Extraction Net consists of multiple computational layers. Six convolutional layers, six ReLU layers, three max pooling layers and batch normalization with each convolutional layer form a feature extractor which is able to automatically extract feature maps from the input radar data. The size of the echo data sample is determined by the size of the sliding window. The size of each convolutional filter is 3×3; each max pooling layer has a 2×2 spatial pooling area with a stride of 2.

2. The range–Doppler feature maps are then filtered by the convolution layers where the size of each convolutional filter is 3×3 and a global feature vector of the size 4096 is extracted by the middle layer of the RD Detection Net. The global feature vector is then passed through two fully connected layers to obtain range and velocity. The latter part of the RD Detection Net consists of convolutional layers, ReLU layers and

Table 1 Specifications of layers in the proposed CNN

Net Name	Layer No.	Convolution kernels	Pooling after convolution
Feature Extraction Net	Conv1	$3 \times 3@32$	–
	Conv2	$3 \times 3@32$	2×2 max pooling
	Conv3	$3 \times 3@64$	–
	Conv4	$3 \times 3@64$	2×2 max pooling
	Conv5	$3 \times 3@128$	–
	Conv6	$3 \times 3@128$	2×2 max pooling
RD Detection Net	Conv7	$3 \times 3@256$	2×2 max pooling
	Conv8	$3 \times 3@512$	2×2 max pooling
	Conv9	$3 \times 3@256$	2×2 up sampling
	Conv10	$3 \times 3@128$	2×2 up sampling
	Fc1	@512	
	Fc2	@2	
Angle Detection Net	Conv11	$3 \times 3@256$	2×2 max pooling
	Conv12	$3 \times 3@512$	2×2 max pooling
	Fc3	@2046	
	Fc4	@512	
	Fc5	@1	
	Fc6	@1	

upsampling layers in order to regain $R \times D \times$ clc output. Notice that the 2-class network with target and not-target outputs is considered in this task. Therefore, the output layer is equipped with soft-max activation function.

3. For each detection, the Angle Detection Net crops a $N \times M \times Ch$ frame from the input feature map which is centered at the location provided by the RD Detection Net. The crop is then filtered with convolution layers, providing the $1 \times 1 \times 4096$ output. This output vector is concatenated with the $1 \times 1 \times 4096$ global feature vector. The concatenated vector is then passed through two fully connected layers; subsequently, azimuth angle and elevation angle are predicted by two independent fully connected layers.

Specifications of the layers, including kernel size, number of kernels, pooling size, etc., are summarized in Table 1.

3.3 Loss functions

The task of RTD addressed in this work is a multitask regression model. The L1-loss and the L2-loss are widely applicable to regression problem. The L1-loss function is used to calculate the mean absolute error between predicted values and true ones, which has good stability with stable gradient, but non-differentiable. The L1-loss is defined as:

$$L_1(x,y) = \frac{1}{n}\sum_{i=1}^{n}|y_i - f(x_i)|. \tag{6}$$

The L2-loss denotes the mean square error of predictions and true values which could guarantee the advantages of continuous and smooth. The L2-loss converges fast with analytical solution. However, it is more sensitive to outliers than L1-loss, which could lead to gradient explosion. The L2-loss function is given as follows:

$$L_2(x,y) = \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2. \tag{7}$$

The smooth L1-loss combines the advantages of the L1-loss and the L2-loss, which is more robust to outliers and defined as:

$$\text{Smooth} \quad L_1(x,y) = \frac{1}{n}\sum_{i=1}^{n}\begin{cases} 0.5*(y_i - f(x_i))^2, & |y_i - f(x_i)| < 1 \\ |y_i - f(x_i)| - 0.5, & \text{otherwise} \end{cases} \tag{8}$$

To address the issue of multitask RTD, we train the proposed network using a weighted smooth L1-loss. After adding each item, the cost function, thus, becomes

$$Loss = \lambda_1 Loss_{RD} + \lambda_2 Loss_{Az} + \lambda_3 Loss_{El} \tag{9}$$

where $Loss_{RD}$ denotes the loss of the RD Detection Net, $Loss_{Az}$ denotes the azimuth angle predicting loss, $Loss_{El}$ denotes the elevation angle predicting loss and λ_1, λ_2, λ_3 are the weights of different losses. During supervised training, the network tries to minimize the reconstruction error.

3.3.1 Experimental results and performance analysis

In this section, we demonstrate the effectiveness of the CNN-based target detector. The labeled radar echo dataset is constructed for training and testing. Then, our model is evaluated under different signal-to-noise ratios (SNRs) and compared with traditional radar signal processing approaches. The experimental procedures are described below.

3.4 Dataset preparation and implementation details

The effectiveness of deep learning models highly depends on the training process and the availability of high-quantity data. However, due to the specificity of RTD tasks, currently, real-world radar data for RTD are not widely accessible. In order to evaluate the considered target detector, a large set of labeled radar complex data is generated, which contains the radar sensor responses to a known point target located at a variety of locations with different velocity. The simulation of dataset is summarized as follows.

1. We consider a typical pulsed radar system in which the transmitted waveform is emitted at a regular interval. The radar carrier frequency is 10 GHz. It generates received signals of 32 coherent pulses in a CPI, which consists of echoes of a target and noise. The transmitted pulses are chirp signal, the pulse repetition frequency is 10KHz and the bandwidth of the chirp signal is 5 MHz. The sampling frequency of the received signals is 10 MHz, and the beam width is 3°. The target is assumed to be slowly fluctuating and follow a standard Swerling 1 distribution. The noise is white Gaussian noise which is an independent complex Gaussian random variable with mean zero.

2. In total, the simulated dataset contains 140,000 data frames. Dimensions of the raw radar echo frame are $256 \times 32 \times 6$. By setting different power of noise, radar echoes with different SNRs can be generated. We set SNR as -6 dB, -2 dB, 0 dB, 2 dB, 6 dB, 10 dB and 13 dB, respectively. A total of 99,225 examples are used to train the CNN detector, 30,775 examples for validation and 10,000 examples for testing.

Since the parameters of our CNN-based detector have to be trained before being applied to detect the target, we optimized the following hyperparameters: *kernel size*, *number of kernels per layer*, *the network depth*, *learning rate* and *dropout rate*. We also tried different combinations of filter size and filter number in each layer to get the best accuracy. The network model is trained with back-propagation and Adam optimizer with initial learning rate 0.0001. In our experiment, we set the dropout rate to 0.4

and the batch size is 100 frames. For angle measurement, ground truth of range and Doppler is used to obtain the 3×3 center crop region. Various $\lambda_1, \lambda_2, \lambda_3$ in Eq. (9) are exploited, and $\lambda_1 = 0.5, \lambda_2 = 1, \lambda_3 = 1$ are selected finally. PyTorch is used for model implementation.

3.5 Experimental results

Measuring error and detection accuracy are used to evaluate our proposed detection model; the measuring error of each item is defined as follows:

$$
\begin{aligned}
R_{\text{error}} &= |R_{de} - R_{gt}| \\
V_{\text{error}} &= |V_{de} - V_{gt}| \\
Az_{\text{error}} &= |Az_{de} - Az_{gt}| \\
El_{\text{error}} &= |El_{de} - El_{gt}|
\end{aligned}
\tag{10}
$$

where R_{de}, V_{de}, Az_{de} and El_{de} denote range, velocity, azimuth and elevation angle of the obtained detection and $R_{gt}, V_{gt}, Az_{gt}, El_{gt}$ are the corresponding ground truth. The accuracy threshold of each detected item is related to parametric resolution and SNR, which can be expressed as $\Delta/\sqrt{SNR}$, where Δ represents parametric resolution of each item. In our setting, for example, when SNR = 10 dB, $\Delta_R/\sqrt{SNR} = 4.743m, \Delta_V/\sqrt{SNR} = 1.482m/s, \Delta_{Az}/\sqrt{SNR} = 0.9°, \Delta_{El}/\sqrt{SNR} = 0.9°$.

During the training process, the total loss of the CNN-based detector and detection performance of each item changes with the number of iterations, which is shown in Fig. 7. The training epoch is 100.

For measuring error of each item, Table 2 shows the average detection error of range, velocity, azimuth and elevation of the proposed approach under different SNRs. According to Table 2, as SNR increases, the detection error drops. The average errors of range and velocity slightly
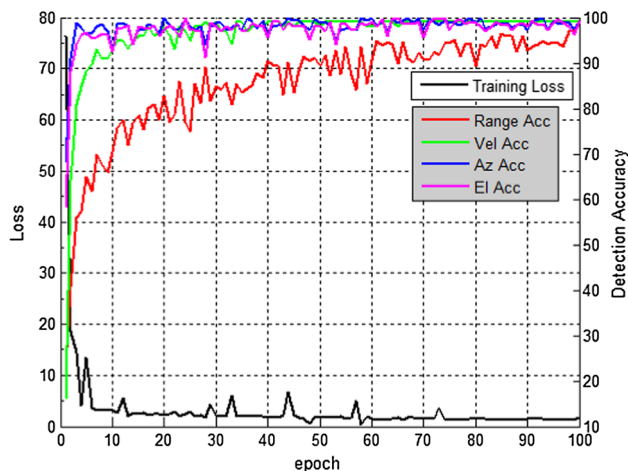

Fig. 7 The loss and detection accuracy of CNN-based detector in each epoch

Table 2 Average measuring error of CNN-based method

Average error	SNR = -6	SNR = − 2	SNR = 0	SNR = 2	SNR = 6	SNR = 10	SNR = 13
Range (m)	1.1642	1.0182	0.9968	0.8963	0.8798	0.8847	0.8813
Velocity (m/s)	0.3966	0.3415	0.2904	0.2510	0.2288	0.2274	0.2233
Azimuth (°)	0.0161	0.0104	0.0079	0.0065	0.0042	0.0027	0.0019
Elevation (°)	0.0167	0.0108	0.0083	0.0070	0.0043	0.0027	0.0019

Table 3 Average measuring error of conventional radar processing method

Average error	SNR = − 6	SNR = − 2	SNR = 0	SNR = 2	SNR = 6	SNR = 10	SNR = 13
Range (m)	3.2585	3.3725	3.2950	3.3050	3.4325	3.3675	3.340
Velocity (m/s)	1.2893	1.2147	1.2002	1.1994	1.2126	1.2118	1.1763
Azimuth (°)	0.0141	0.0135	0.0128	0.0117	0.0109	0.0099	0.0097
Elevation (°)	0.0172	0.0163	0.0133	0.0129	0.0124	0.0114	0.0100

drop as SNR increases, but when SNR is higher than 6 dB, the overall trend remains steady. The average errors of azimuth and elevation drop as the SNR increases, but both in a small scale, and the average errors of range and velocity are larger than those of azimuth and elevation.

In conventional radar processing methods, the received echoes are processed by matched filtering, Doppler processing, CA-CFAR detection, sum–difference angle measurement, etc. For the CA-CFAR detector, there are six reference cells, four guard cells on each side of range domain, and five reference cells, three guard cells on each side of Doppler domain. The threshold of CFAR comparator changes with various SNR. According to Table 3, the average errors of range and velocity are slightly varied in different SNRs, while the average errors of azimuth and elevation drop as the SNR increases.

We compare the measuring errors of CNN-based detector with that of the conventional method in Fig. 8. It can be shown that errors have been greatly reduced in range and velocity, although in Fig. 8(c)(d), the angle errors are higher than classical one when SNR = − 6 dB, as SNR increases, both angle errors of our proposed method are lower than those of classical method.

3.6 Performance analysis

Only when measuring errors of four items, the below respective threshold is the detection considered as valid, which is defined as

$$T_d = \left(R_{error} \leq \frac{\Delta_R}{\sqrt{SNR}} \right) \& \left(V_{error} \leq \frac{\Delta_V}{\sqrt{SNR}} \right) \& \left(Az_{error} \leq \frac{\Delta_{Az}}{\sqrt{SNR}} \right) \& \left(El_{error} \leq \frac{\Delta_{El}}{\sqrt{SNR}} \right) \quad (11)$$

Accuracy of detection D_a is defined as

$$D_a = \frac{N_{td}}{N_{total}} \quad (12)$$

where N_{td} denotes the number of correctly detected targets and N_{total} denotes the total number of simulated targets.

According to evaluation criterion which is presented in Eq. (12), the detection accuracy is obtained and listed in Table 4. Then, we compare the detection performance of CNN-based detection method with classical method under different SNRs, which is shown in Fig. 9.

It is evident that our CNN-based detector shows a better detection accuracy than the conventional radar signal processing method. As the SNR rises, the detection accuracy of the two methods rises. The detection accuracy of the CNN-based detector is larger than that of the conventional one when SNR is lower than 10 dB. Evidently, our proposed CNN-based detection model is able to get rich information about target from raw radar echo data and it is more robust to white noise and continues to outperform the conventional radar processing methods under various SNR conditions.

Accuracy of detection is also used to compare the proposed RTD models fairly with other state-of-the-art methods in the literature. Table 5 compares the performance of the proposed CNN-based approach with the conventional CA-CFAR method and the state-of-the-art ANN-based and CNN-based methods on our dataset with SNR = 10 dB. The detection and training procedure of GO-ANN (Akhtar et al.) and DRD (Brodeski et al.) are implemented on range–Doppler maps, which are generated by multiple raw radar echoes. They have achieved 95.2% and 96.7% detection accuracy on our dataset, respectively. Notice that the proposed method outperforms the conventional method and state-of-the-art ANN-based and CNN-based methods with detection performance of 98.5%.
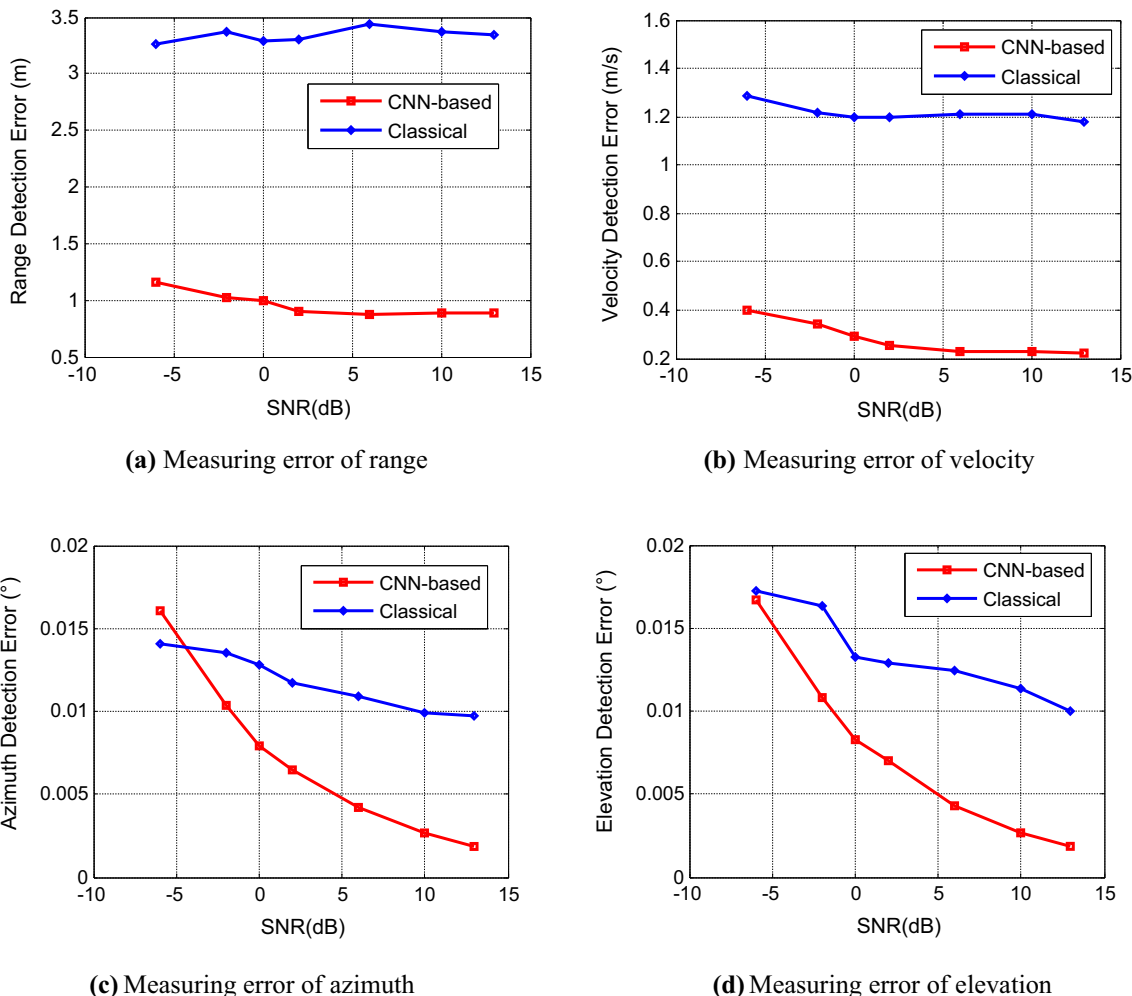
(a) Measuring error of range

(b) Measuring error of velocity

(c) Measuring error of azimuth

(d) Measuring error of elevation

Fig. 8 Comparison of average error under different SNRs

Table 4 Accuracy of our CNN-based and conventional detection methods under different SNRs

Accuracy	SNR = − 6	SNR = − 2	SNR = 0	SNR = 2	SNR = 6	SNR = 10	SNR = 13
CNN-based	0.6450	0.7225	0.7985	0.8445	0.9070	0.9850	0.9985
Classic method	0.5630	0.6335	0.6935	0.7775	0.8335	0.9805	0.9980

3.7 Ablation experiment

To research the need of passing the global feature vector and classes vector to Angle Detection Net, we perform an ablation experiment. The Angle Detection Net and the RD Detection Net are trained separately with the same input feature maps. The convolutional layers, the fully connected layers and the loss functions remain the same except the cropped part and concatenated part which are skipped. The network model is trained as before using back-propagation and Adam optimizer with initial learning rate 0.0001. The dropout rate is 0.4 and the batch size is 100 frames.

The azimuth and elevation measuring error of the proposed model and the separate Angle Detection Net are

presented in Fig. 10. It can be clearly observed the improvement in the azimuth–elevation estimation accuracy of the proposed method compared to the separate Angle Detection Net. This improved detection accuracy may be attributed to the additional information in the global feature vector from the RD Detection Net.

4 Discussion

Many of the research works in the literature express an evident trend that the traditional radar signal processing methods and deep learning schemes are deeply integrated in RTD application. On the one hand, conventional radar
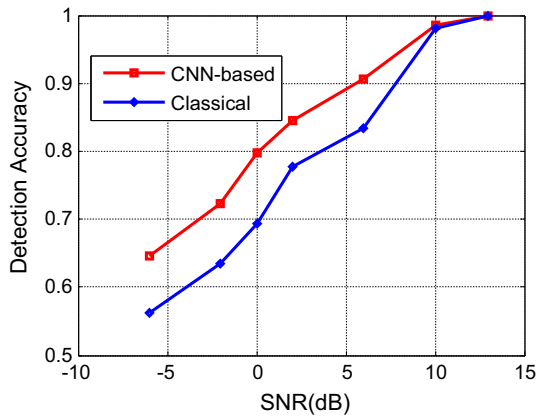
**Fig. 9** Detection accuracy of CNN-based and classical detector under different SNRs

signal processing serves as the preprocessing method of the training data, such as pulse compression, Doppler processing and STFT, which can help to improve the detection performance. On the other hand, neural networks integrated with radar signal processing methods, such as CFAR, working as training strategies, have contributed to the performance improvement. Further, based on the results obtained by our proposed method, we argue that deep learning models can provide an end-to-end framework to integrate sensing, processing and decision making.

Although there are some successful application paradigms of deep learning-based methods in the field of RTD, huge challenges still remain. One of the bottlenecks of applying deep learning models to RTD is the lack of labelled data. Unlike other application fields, high cost is involved in obtaining the radar data. Although radar modeling is a solution to overcome this issue, generating the surrogate data is also extremely challenging and computationally demanding since multiple effects must be considered, such as interference, multipath reflections, reflective surfaces, discrete cells and multiple attenuation. Even if all the above problems are avoided, it inevitably relies on a mathematical model for simulation which may bring inaccuracies. To further promote the study of RTD with insufficient real-world radar data, the following ideas could be considered: 1) data augmentation; 2) some new generation of deep learning algorithms, such as generative adversarial networks (GANs) [36], are robust to the problem of insufficient training data; 3) some other learning-based methods like transfer learning [37, 38] and meta learning [39] could break through the limitation of data insufficiency.

In addition, despite a remarkable improvement in the application of deep learning in RTD, the available literature on ANNs and DNNs in RTD still remains relative sparse and lacks high-level maturity. For example, a

Table 5 Comparison of performance of the proposed method with other state-of-the-art methods

Method	Feature extraction approach	Input forms	D_a@10 dB
GO-ANN [29]	Handcrafted	Range–Doppler map	95.2%
DRD [32]	CNN	Range–Doppler map	96.7%
CA-CFAR	Handcrafted	Raw data	98.1%
Proposed method	CNN	Raw data	98.5%
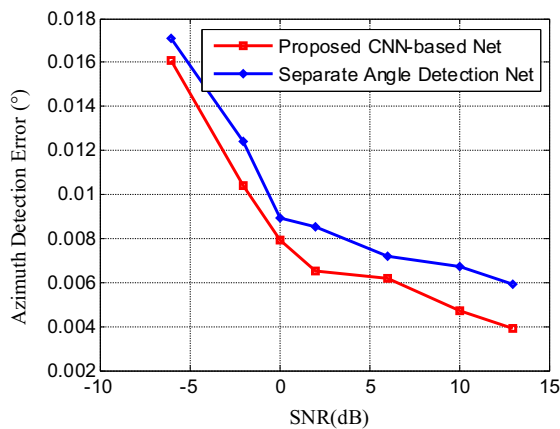
(a) Measuring error of azimuth angle
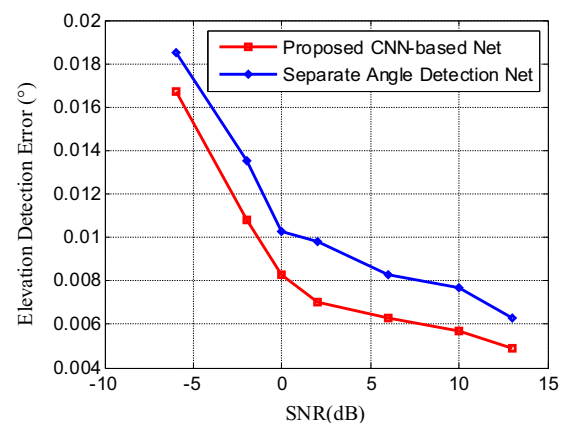
(b) Measuring error of elevation angle

Fig. 10 Accuracy comparison of the proposed net and the separate Angle Detection net

common aspect found in some of the literature is that the ANNs are in moderate size, the networks proposed in [26, 27], containing only one hidden layer. Only CNNs are widely used in RTD, even though various deep learning architectures have been proposed in other fields. Since radar processing system is facing more complex and challenging tasks, more varied but practical powerful learning-based training strategies are urgently required.

5 Conclusion

In this paper, we proposed a novel CNN-based model for multitask RTD, which works directly with raw radar echo data. Accordingly, the rich original information contained in the echo signal can be fully exploited. The proposed detection method is able to locate the target in multi-dimensional space of range, radial velocity, azimuth and elevation. The CNN-based detector shows a better measuring accuracy in each item than the conventional radar signal processing method when evaluated under different SNRs. In addition, the detection performance of the proposed method outperforms other state-of-the-art methods in the literature. Although this work adopts simulated data for training, we believe that it has promising potential to apply deep learning to RTD on radar complex data. In addition, we expect to get sufficient training data from radar system which will definitely prompt the detection model more convincing. In the future, we will try to extend and evaluate the CNN-based detector with real-world radar data.

Compliance with Ethical Standards

Conflict of interest There is no conflict of interest from the authors.

References

1. Richards MA (2009) Fundamentals of Radar Signal Processing. IEEE Signal Process Mag 26(3):100–101. https://doi.org/10.1109/MSP.2009.932123
2. Long T, Liang Z, Liu Q (2019) Advanced Technology of High-Resolution Radar: Target Detection, Tracking, Imaging and Recognition, Science China. Inf Sci. https://doi.org/10.1007/s11432-018-9811-0
3. Melvin WL (2013). Principles of Modern Radar: Advanced Techniques. https://doi.org/10.1049/SBRA020E
4. LeCun Y, Bengio Y, Hinton G (2015) Deep Learning. Nature 521(7553):436–444
5. Bengio Y, Courville A, Vincent P (2013) Representation Learning: A Review and New Perspectives. IEEE Trans Pattern Anal Mach Intell 35(8):1798–1828
6. Goodfellow I, Bengio Y, Courville A (2016) Deep Learning, Cambridge, MA: MIT Press. Available: http://www.deeplearningbook.org/
7. Öztürk Ş (2020) Stacked Auto-encoder based Tagging with Deep Features for Content-based Medical Image Retrieval. Experts Systems with Applications. https://doi.org/10.1016/j.eswa.2020.113693
8. Huang G, Liu Z (2017) Weinberger KQ (2017) Densely Connected Convolutional Networks. IEEE Conference on Computer Vision and Pattern Recognition, Honolulu 21–26:4700–4708. https://doi.org/10.1109/CVPR.2017.243
9. Chen L, Papandreou G, Kokkinos I, Murphy K, A. Yuille L, (2015) Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFS. Proc. Int. Conf, Learn
10. Simonyan K, Zisserman A (2014) Very Deep Convolutional Networks for Large-scale Image Recognition.
11. Redmon J, Divvala S, Girshick R (2016) Farhadi A (2016) You Only Look Once: Unified. Real-time Object Detection, IEEE Conference of Computer Vision and Pattern Recognition 27–30:779–788. https://doi.org/10.1109/CVPR.2016.91
12. Wang S, Sun J, Phillips P, Zhao G, Zhang Y (2018) Polarimetric Synthetic Aperture Radar Image Segmentation by Convolutional Neural Network using Graphical Processing Units. Journal of Real Time Image Processing. 15:631–642. https://doi.org/10.1007/s11554-017-0717-0
13. Chen G, Li C, Wei W, Jing W, Woźniak M, Blažauskas T, Damaševičius R (2019) Fully Convolutional Neural Network with Augmented Atrous Spatial Pyramid Pool and Fully Connected Fusion Path for High Resolution Remote Sensing Image Segmentation. Applied Sciences. https://doi.org/10.3390/app9091816
14. Zhang Y, Dong Z, Chen X, Jia W, Du S, Muhammad K, Wang S (2017) Image based Fruit Category Classification by 13-layer Deep Convolutional Neural Network and Data Augmentation. Multimedia Tools and Applications. 78:3613–3632. https://doi.org/10.1007/s11042-017-5243-3
15. Öztürk Ş, Özkaya U (2020) Skin Lesion Segmentation with Improved Convolutional Neural Network. J Digit Imaging 33:958–970. https://doi.org/10.1007/s10278-020-00343-z
16. Öztürk Ş, Özkaya U (2020) Gastrointestinal Tract Classification using Improved LSTM based CNN. Multimedia Tools and Applications. 79:28825–28840. https://doi.org/10.1007/s11042-020-09468-3
17. Zhang M, Jing W, Lin J, Fang N, Wei W, Woźniak M, Damaševičius R (2020) NAS-HRIS: Automatic Design and Architecture Search of Neural Network for Semantic Segmentation in Remote Sensing Images. Sensors. https://doi.org/10.3390/s20185292
18. Zhang K, Zhang Z, Wang H, Li Z, Qiao Y, Liu W (2017) Detecting Faces Using inside Cascaded Contextual CNN, IEEE International Conference on Computer Vision
19. Zhu C, Zheng Y, Luu K, Savvides M (2017) Cms-rcnn: Contextual Multiscale Region-based CNN for Unconstrained Face Detection, Deep Learning for Biometrics
20. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet Classification with Deep Convolutional Neural Networks. International Conference on Neural Information Processing Systems. 93:1097–1105
21. Simonyan K, Zisserman A (2014) Very Deep Convolutional Networks for Large-scale Image Recognition.
22. Szegedy C, Liu W, Jia Y (2015) (2015) Going Deeper with Convolutions. IEEE Conference of Computer Vision and Pattern

Recognition, Boston, MA 7–12:1–9. https://doi.org/10.1109/CVPR.2015.7298594

23. Targ S, Almeida D and Lyman K (2016) Resnet in Resnet: Generalizing Residual Architectures. arXiv eprints, https://ui.adsabs.harvard.edu/abs/2016arXiv160308029T

24. Gandhi P, Ramaurti V (1997) Neural Networks for Signal Detection in Non-gaussian Noise. IEEE Trans Signal Process 45(11):2846–2851

25. Jarabo-Amores P, Mata-Moya D, Zurera MR, Nieto-Borge JC, López-Ferreras F (2006) A Neural Network Approach to Improve Radar Detector Robustness. 14th European Signal Processing Conference. 8; 4–8

26. Rohman B, Kurniawan D, Miftahushudur MT (2015) Switching CA/OS CFAR using Neural Network for Radar Target Detection in Non-Homogeneous Environment. 17th International Electronics Symposium. doi: https://doi.org/10.1109/ELECSYM.2015.7380855

27. Akhtar J, Olsen KE (2018) A Neural Network Target Detector with Partial CA-CFAR Supervised Training, International Conference on Radar

28. Cheikh K, Soltani F (2006) Application of Neural Networks to Radar Signal Detection in K-distributed Clutter. IEEE Proceeding Radar Sonar Navigation 153(5):460–466

29. Akhtar J, Olsen KE (2019) GO-CFAR Trained Neural Network Target Detectors. IEEE Radar Conference. https://doi.org/10.1109/RADAR.2019.8835765

30. Pan M, Chen J, Wang S, Dong Z (2019) A Novel Approach for Marine Small Target Detection Based on Deep Learning, IEEE 4th International Conference on Signal and Image Processing, 978–1–7281–3660–8, pp.395–399, doi:https://doi.org/10.1145/3065386

31. Wang L, Tang J, Liao Q (2019) A Study on Radar Target Detection Based on Deep Neural Networks. IEEE Sensors Letters. https://doi.org/10.1109/LSENS.2019

32. Brodeski D, Bilik I, Giryes R (2019) Deep Radar Detector. IEEE Radar Conference. https://doi.org/10.1109/RADAR.2019.8835792

33. López-Risueño G, Grajal J, Haykin S, Díaz-Oliver R (2002) Convolutional Neural Networks for Radar Detection. International Conference on Artificial Neural Networks. https://doi.org/10.1007/3-540-46084-5_186

34. Su N, Chen X, Guan J, Li Y (2019) Deep CNN-Based Radar Detection for Real Maritime Target Under Different Sea States and Polarizations. Cognitive Systems and Signal Processing. https://doi.org/10.1007/978-981-13-7986-4-29

35. Kay SM (1998) Fundamentals of Statistical Signal Processing: Detection Theory, vol 2. Hoboken, NJ, USA

36. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B (2014) Generative Adversarial Networks. Adv Neural Inf Process Syst 76:2672–2680

37. Yosinski J, Clune J, Bengio Y, Lipson H (2014) How Transferable are Features in deep neural networks. Advances in Neural Information Processing Systems, Oral presentation

38. Oquab M, Bottou L, Laptev I (2014) Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. IEEE Conference on Computer Vision. https://doi.org/10.1109/CVPR.2014.222

39. Hospedales T, Antoniou A, Micaelli P, Storkey A (2004) Meta-Learning in Neural Networks: A Survey.