A SAR Target Image Simulation Method With DNN Embedded to Calculate Electromagnetic Reflection

Shengren Niu ⓘ, Xiaolan Qiu ⓘ, *Senior Member, IEEE*, Bin Lei, and Kun Fu ⓘ

Abstract—Electromagnetic (EM) scattering calculation is a very important part of most synthetic aperture radar (SAR) target image simulation methods. It affects the intensity of the radar echo signal to a great extent, thus affecting the quality of the final simulation image. EM reflection models are usually approximate formulas derived under certain assumptions. The errors between these models and the actual situation can cause significant differences between simulated images and real images. To solve this problem, we propose a novel modified SAR target image simulation framework, in which the deep neural network (DNN) is embedded to calculate the EM reflection, so that the DNN can directly learn and fit the EM reflection models from real SAR images. First, the intensity calculation of radar signal in a single reflection is separated from the cumulative calculation of multiple radar reflection signals intensity in each pixel. Thus, the approximate calculation formulas of EM reflection can be replaced with the DNN models. Next, the DNN model is trained with the backpropagation algorithm to learn the actual EM reflection model from real SAR images. Finally, the fitted EM reflection models and an image post-processing model are applied to simulate images of the target under different imaging angles. In the simulation framework, the functions of the neural network models are limited to calculating the reflection coefficient and adding sidelobe and speckle noise. The imaging model is still the original simulation method based on ray tracing, which ensures the correctness and generalization of the simulation method. Experiments show that the proposed simulation method can significantly improve the quality of the simulation image. When the image is normalized to [0, 1], the minimum mean square error between the simulated SAR images and the real images of the Sandia laboratory implementation of cylinders target can reach 0.003. The visualization results of the DNN models show that the fitted reflection coefficient calculation curve and the convolution kernel used for image post-processing are consistent with the laws in the theoretical model. In addition, when the proposed method is used to simulate complex targets, the similarity of simulation images can also be significantly improved.

Index Terms—Deep neural network (DNN), differentiable rendering, electromagnetic (EM) scattering modeling, ray tracing, synthetic aperture radar (SAR) image simulation.

I. INTRODUCTION

SYNTHETIC aperture radar (SAR) image interpretation and application are key problems in SAR research. SAR target image simulation is of great significance to both of them. Specifically, the research on the principle of SAR image simulation is helpful to improve the ability of SAR image interpretation [1]. SAR target image simulation can quickly provide a large number of images of the target under different imaging conditions at a lower cost. These simulation images can be further applied to a wide range of fields such as target feature extraction [2], automatic target detection and recognition [3], [4], and building height inversion [5] and other image-based applications.

Fast simulation speed and convenient setting of simulation parameters are the prerequisites for a method to be used to simulate a large number of images. The quality of simulation image is the key factor affecting its application scope and effect. The simulation speed and quality of SAR image simulation are closely related to the electromagnetic (EM) scattering calculation method and imaging model. According to these two points, as far as we know, the current mainstream SAR image simulation methods can be divided into two categories.

The first simulation method originates from the radar cross section (RCS) simulation and takes the turntable model as the imaging model. This simulation method first calculates the RCS of the target at multiple azimuths and multiple frequency points to simulate the echo signal received by the radar antenna, and then uses the 2D IFFT imaging algorithm for imaging [6]. In this method, the local geometry and material information can only be reflected as the RCS of the whole target. Therefore, it is difficult to explore how the EM scattering occurred on many local surface elements affects the size and distribution of the scattering points of the final simulation image, and it is also difficult to use the result of comparison between simulation image and real image to adjust the simulation system.

The improvement of this simulation method is mainly focused on improving the modeling accuracy of target geometry and material and improving the precision and speed of numerical calculation of target's RCS [7]–[9]. However, accurate target modeling usually takes a lot of manpower, and accurate numerical calculation of complex target RCS also requires a lot of time and computing power. It often takes hours to days to simulate one SAR image of a complex target, but the final simulation image may still have some differences with the real image, and is not easy to be adjusted.

The second kind of simulation method originates from computer graphics. The work of Franceschetti *et al*. [10], [11]

provides the theoretical basis for using the small panel model to simulate SAR images of complex ground scenes. The theoretical derivation of SAR imaging formulas ensures that the imaging model based on ray tracing method is consistent with the real SAR system [12]. Generally, the reflection law of radar signals on the local surface is approximated by the specular reflection and diffuse reflection models in the optical field, and the radar signal echo is accumulated on range-azimuth plane by ray tracing method for imaging [1]–[13]. Compared with the simulation method based on RCS simulation, the relationship between the models and the simulation image in this method is more concise and intuitive. Although there are a lot of rays involved in the simulation process that need to be traced, the rendering program library based on CPU or GPU parallel acceleration technology greatly reduces the calculation time. The simulation of an image can be completed in a few minutes or even less than one second. Such fast simulation speed lays a good foundation for its wide application.

However, this method also has some shortcomings. First, the EM wave in microwave band and visible light band are different. In visible light band, since the wavelength of light is often far less than the roughness of the object surface, the reflection type of EM wave is mainly diffuse reflection. In microwave band, the wavelength is often larger than the surface roughness of the object, so the reflection type is mainly specular reflection. Therefore, the parameters of the optical approximation models used in the SAR image simulation have no corresponding actual values, and can only be adjusted according to the simulation experience and simulation results. Second, ray-tracing method often ignores the diffuse multiple reflections [14]. Therefore, the approximate EM reflection models used in this simulation method is different from the actual case.

The key approaches to improve the performance of this simulation method is to automatically set the simulation parameters, and to obtain more accurate EM reflection models that are compatible with the imaging models based on ray-tracing technology. Deep learning technology, which has achieved good results in many optical image applications, is a powerful tool to realize these two points. In recent years, some scholars have made a lot of meaningful explorations on how to apply deep neural network (DNN) model in SAR image simulation. Referring to the application of the conditional generative adversarial networks (C-GAN) model in image generation, Guo *et al.* [15] train the C-GAN model with SAR images to generate SAR images of targets under different azimuth and depression. Liu *et al.* [16] further learn from the successful application of GAN in optical image style conversion, and input simulation SAR images into GAN model for further modification, so as to improve the application effect in classification problems.

Although the DNN model has very strong "end-to-end" fitting and learning capabilities, it may not be a good approach to directly transfer its application in the field of optical images to the field of SAR image. The imaging principle of the SAR system is different from that of optical system. SAR images are also very different from optical images. For example, targets in SAR images are not color blocks with clear boundary, but scattering points with different intensity and sizes. Since the reflection

of radar signal is mainly specular reflection, these scattering points are very sensitive to the imaging angle. A small change in imaging angle can result in a huge change of the scattered point.

In the problem of how to effectively and correctly apply DNN to SAR target image simulation, since SAR images are very sensitive to imaging angles, the correlation between SAR images from different angles is much lower than that of optical images, it is very important to ensure the generalization ability of the simulation method. Too complex DNN or directly using DNN to generate images often cannot achieve good simulation results, and the interpretability of the DNN models are also in doubt. However, most of these attempts which apply deep learning to SAR image simulation are only based on the successful DNN model, without in-depth analysis of the existing simulation methods or clear explanation of the function of deep network model. Guo pointed out in his article that the SAR images generated by GAN are seriously degraded under some azimuth, and are very different from real images. Liu only used the modified simulation SAR image as the augmented data of the real SAR image.

The core idea of this article is to use the neural network with simple structure and fewer parameters to complete specific and simple functions in simulation, so as to increase the interpretability of the network and the generalization ability of the simulation method. It is also one of the goals and very important innovations of the framework proposed in this article. We believe that a good framework combining SAR image simulation with DNN model should follow the following design principles.

1) Keep the original simulation principle as much as possible.
2) Clearly define the specific function of DNN model in simulation, so as to improve the interpretability of DNN model.
3) Limit the function and input data of the DNN, thus avoiding the DNN model to directly generate images based on the features of target category and azimuth, instead of the usual simulation calculation data.

In [17], we proposed and implemented a method which automatically extract simulation parameters from a single real SAR image using DNN model. In this method, the function of the DNN model is limited to extracting simulation parameters based on the brightness of target and background in SAR real images, rather than directly modifying the simulation image. In this way, the original simulation method and formula are kept unchanged and the generalization ability of the simulation method is guaranteed.

After solving the problem of automatic setting of simulation parameters, we think that the adjustability of the approximate EM models is limited, and even if the simulation parameters are accurately adjusted, the neglected part of the scattering may still not be compensated. Therefore, as mentioned earlier, it is imperative to further adjust the original EM reflection models. We believe that this direction of applying DNN models to SAR image simulation can achieve better results. In this way, the adjustability of the reflection models is increased. Even when there is only material label, the calculation model can be directly fitted from the real images without the need to measure the simulation parameters in the field.

Xu *et al.* made a beneficial preliminary attempt in this direction [18]. They trained the DNN model to fit the scattered field of an inhomogeneous region by taking the permittivity map in a circular area as input and the scattered field calculated by a 2-D finite element-boundary integral model as output. Their experiments show that DNN can predict the scattering field at an acceptable level on a certain number of datasets, but the DNN model will no longer converge when the datasets become more complex. Therefore, the permittivity distribution data alone is not sufficient for the DNN model to learn an effective EM scattering model with strong generalization ability for complex targets. Li *et al.* used DNN to solve the nonlinear EM near-field inverse scattering [19].

SAR image simulation can be thought as a complex combination of local operations. "Simple operation" refers to the EM reflection occurring on the local surface of an object. The EM reflection model is a multivariate function, with less than six input variables and one or two output variables. Hence, this function performs dimensionality reduction operation on the low-dimensional data, that is, from 6-D to 1-D. "Complex combination" means imaging the scattered signal echoes. If imaging process is also considered as a function, the number of its input variables is the number of scattered signal echoes, and the number of output variables is the number of pixels of the simulation image. Therefore, the mapping relationship between the scattering signal echoes and the SAR image represented by this function is the mapping relationship between the data in high dimensional space. In this high-dimensional space, data is sparse, so it is difficult for DNN models to learn complex and general models. However, for the EM reflection calculation on local surfaces, which is a mapping problem in low dimensional space, the data is enough to train and learn a DNN model with strong generalization ability.

On the basis of the above analysis, we propose the idea of using DNN models to fit EM reflection model from real images, and design and implement a simulation framework which embeds DNN model into the simulation process. In this framework, the DNN model is used to replace the original EM approximate reflection models to calculate each EM reflection of radar signal on the local surface of the object. The input data of the DNN model is limited to the material and geometric information on the local surface, without any overall characteristics of the target. Thus, it is avoided that the DNN model generate specific images based only on the overall structure of the target. The output data of the DNN model is limited to reflection intensity or reflection coefficient, which emphasizes the specific function of the DNN model in simulation and enhances the interpretability of network. Meanwhile, the function that DNN needs to learn is a function curve fitting task on a low-dimensional and dense dataset, which ensures the generalization performance of the network.

It may be easy to come up with the idea of using DNN model to learn and fit EM reflection model from real images, but there are many difficulties in the process of realizing this idea.

1) How to use the reflection intensity data accumulated in a single pixel to train the single reflection model of the signal.

2) How to efficiently and coherently record and organize the large amount of data related to EM reflection computing. These data are generated by the multiple reflections of radar signals on the local surface.

3) How to design an elaborate network structure to efficiently calculate the EM reflection on the local surface element, and to adopt an appropriate model connection mode to ensure the correct intensity relationship between reflection and incident signals in multiple reflections.

4) How to ensure that all steps in the simulation framework from the calculation of EM reflection intensity to the final output of the simulation image are differentiable, so as to reverse transfer the simulation error to the DNN model.

5) How to quickly and effectively calculate the error between the simulation image and the real image which are not registered.

Aiming at the above problems and difficulties, we redesigned and adjusted the whole simulation framework on the basis of keeping the original simulation method and imaging principle unchanged. We design a multichannel "image" model data structure to store simulation information, and an ingenious DNN model and a connection architecture between models in a new perspective. Finally, we complete the function of each part of the simulation framework, and carry out complete and effective experimental verification.

The innovation of this article can be summarized as follows.

1) Under the condition of keeping the imaging model of the original SAR target image simulation method unchanged, we propose a simulation framework that embed DNN model into the simulation process for EM reflection calculation. This method concours the above five difficulties.

2) The simulation results are verified by moving and stationary target acquisition and recognition (MSTAR) Sandia laboratory implementation of cylinders (SLICY) and T-72 tank data. Compared with other applications of the DNN model in the SAR image simulation, the method proposed in this article emphasizes more on the specific functions of the deep network model in simulation and enhances the interpretability of the network and so get better generality. Compared with the simulation parameter adjustment using DNN model, the DNN model in this method is more closely combined with the simulation method and get better learning ability.

The rest of this article is organized as follows. Section II mainly describes the basic simulation method of the proposed simulation framework, namely the simulation method based on ray tracing. The imaging model and EM reflection calculation model of the simulation method are introduced. Section III describes the SAR image simulation framework which embedded the DNN models for EM reflection calculation, as well as the specific implementation methods of each part. Section IV verifies the design of each part, and visualizes the function of DNN models embedded in the simulation framework. Finally Section V summarizes the content of the full article, and gives the further research and application direction.
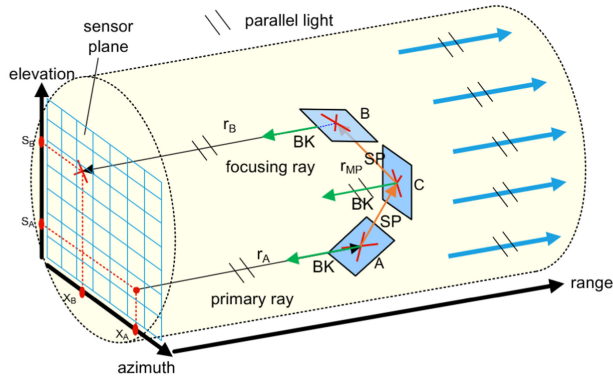
Fig. 1. Design of virtual SAR sensor: orthographic projection in azimuth and elevation for local scenes; parallel light for representing the radar signal.

II. MODEL AND PROBLEM

In this section, we mainly introduce the basic simulation method in the proposed simulation framework. It is a modified and ported version of RaySAR [13]. RaySAR is a state-of-art ray-tracing SAR image simulator [15] which is developed and open source by Dr. Stefan Auer [1]–[13] for analyzing multiple reflection in SAR images.

In [17], we proposed that the SAR image simulation can be abstracted as

$$X' = f(M, Y') \qquad (1)$$

where M is the geometry of an object, Y' is the surface parameters, f is the simulation method, and X' is the simulation image. Generally, the simulation function f needs to complete two main functions: calculate the intensity of radar signal, and calculate the focused position of the signal echo in the range-azimuth plane. These two functions correspond to the EM scattering model and the imaging model in the simulation method respectively.

In the imaging model, RaySAR uses parallel light for representing the radar signal and orthographic projection in azimuth and elevation for local scenes. In the EM scattering model, it uses the specular reflection and diffuse reflection models in the optical field to approximately calculate the intensities of radar signal signals as they are reflected between objects and as they are reflected from objects to the radar antenna. The details of the imaging model and EM reflection model of this method are described in the following content. In addition, some problems in applying this simulation method are explained.

A. Imaging Model

Fig. 1 shows the imaging model of RaySAR. In this model, a plane of the same size as the scene being imaged is used as the virtual SAR sensor. It is called the sensor plane and is used for transmitting and receiving radar signals. A set of parallel rays perpendicular to the sensor plane is used as the radar signal.

The simulation imaging process can be divided into three steps: transmitting radar signals, calculating the interaction between signals and the target scene, and accumulating the signal echo to the range-azimuth plane for imaging. First, a
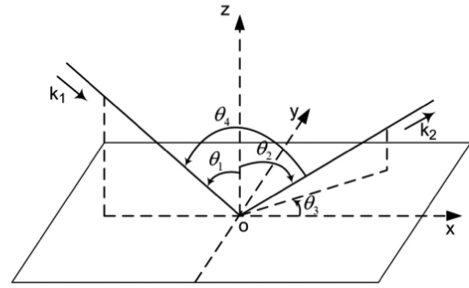

Fig. 2. Geometry of scattering problem.

set of parallel rays are emitted from the pixels on the sensor plane to the scene. Each pixel cell can emit multiple rays. These rays are called primary rays. Second, when these rays intersect the target or background, they are reflected and generate new signal rays. RaySAR simulates the reflected signal rays of two directions: the direction of specular reflection signal (SP) and the direction of backward echo (BK). Third, the rays in the BK direction, called focusing ray, will return to the sensor plane and be accumulated into the receiving pixel unit according to their focusing positions in the range-azimuth plane. The rays in the SP direction, called the secondary, tertiary rays, etc., will continue to intersect with the target or background and generate new rays of the two directions.

The cumulative position of the focusing rays should be consistent with the position of the real echo signal. Although RaySAR does not use the Range-Doppler coherent imaging algorithm, its imaging geometry model can guarantee the above relationship. In RaySAR, the position of the echo is calculated by projecting the starting point of the focusing ray into azimuth direction and elevation direction respectively. According to the derivation in [12], we further change the calculation model to

$$x = \frac{x_A + x_B}{2} \qquad (2)$$

$$r = \frac{1}{2}(r_A + r_B + r_{MP}) \qquad (3)$$

$$y = \frac{r - r_0}{2} \qquad (4)$$

where x represents the azimuth coordinate, while x_A and x_B are respectively the coordinates of the incidence point of primary ray and the exit point of focusing ray projected on the azimuth direction, as shown in the Fig. 1. The focusing position of a ray in the azimuth is the middle position between the ray's incident point and the exit point. y is the range coordinate, and r is the sum of lengths of the ray which emitted from the sensor plane r_A and returned to the sensor plane r_B after multiple reflections r_{MP}. r_0 is the total length of the ray corresponding to the signal echo in the center of the scene.

B. Approximate EM Reflection Model

In RaySAR, the intensity of the signal echo is only related to the multiple reflections of the ray on the local surface. Fig. 2 illustrates the geometric data involved in ray reflection. θ_1 represents the angle between the incident ray and the surface normal. θ_2 represents the angle between the reflection direction

and the surface normal. θ_3 represents the azimuth of the direction of reflection.

The reflection of radar signal on the local surface is divided into two parts: specular reflection and diffuse reflection. The Fresnel model, e.g., reported in [20], is common for approximating the specular reflection of radar signals. The Fresnel coefficient in horizontal polarization is

$$R_h = \frac{\cos\theta_1 - \left(\varepsilon_r - \sin^2\theta_1\right)^{1/2}}{\cos\theta_1 + \left(\varepsilon_r - \sin^2\theta_1\right)^{1/2}}. \tag{5}$$

The Fresnel coefficient in vertical polarization is

$$R_v = \frac{\varepsilon_r\cos\theta_1 - \left(\varepsilon_r - \sin^2\theta_1\right)^{1/2}}{\varepsilon_r\cos\theta_1 + \left(\varepsilon_r - \sin^2\theta_1\right)^{1/2}} \tag{6}$$

where ε_r is the relative dielectric constant.

In case of diffuse reflection of radar signals, the small perturbation method (SPM) [20] is applicable for surfaces whose roughness is small compared to the signal wavelength. The calculation formula of SPM is

$$|E_{pq}|^2 = \frac{4a\left|k^2\cos\theta_1\cos\theta_2\beta_{pq}\right|^2 W\left(\eta_{xy}\right)}{(2\pi R)^2} \tag{7}$$

$$\beta_{hh} = \frac{(\varepsilon_r - 1)\cos(\theta_3)}{\left(\cos\theta_2 + \sqrt{\varepsilon_r - \sin^2\theta_2}\right)\cdot\left(\cos\theta_1 + \sqrt{\varepsilon_r - \sin^2\theta_1}\right)} \tag{8}$$

$$\beta_{vv} = \frac{\sqrt{\varepsilon_r - \sin^2(\theta_2)}\cdot\sqrt{\varepsilon_r - \sin^2(\theta_1)}\cdot\cos\theta_3\cdot(\varepsilon_r - 1)}{\left(\varepsilon_r\cos\theta_2 + \sqrt{\varepsilon_r - \sin^2\theta_2}\right)}$$

$$\frac{-\varepsilon_r\sin(\theta_1)\sin(\theta_2)(\varepsilon_r - 1)}{\cdot\left(\varepsilon_r\cos\theta_1 + \sqrt{\varepsilon_r - \sin^2\theta_1}\right)} \tag{9}$$

where a is the area of the surface, $k = 2\pi/\lambda$ is the signal wavenumber, and R is the spatial distance between SAR and object. β_{pq} is a coefficient associated with the mode of polarization. $W(\eta_{xy})$ is the surface power spectrum, i.e., the Fourier transform of the surface autocorrelation function. It may be defined by means of standard surface models such as gaussian or exponential shapes or by using fractals [21].

The Fresnel model and SPM are derived from Maxwell's equations under certain assumptions. However, RaySAR does not actually use the above EM reflection models, because the ray-tracing library on which RaySAR was developed, POV-ray, is originally intended for optical simulation, and does not provide such models. Hence, RaySAR adopts the rendering models used in optical simulation field to approximates the reflection of radar signals. In RaySAR, the model for specular reflection model is

$$I_s = F_s \cdot (\vec{N} \cdot \vec{H})^{\frac{1}{F_r}} \tag{10}$$

where F_s is a specular reflection coefficient [0, 1] and F_r is a roughness factor defining the sharpness of the specular highlight. $\vec{N}$ is the surface normal vector. $\vec{H}$ is a bisection vector. The model for diffuse reflection model is

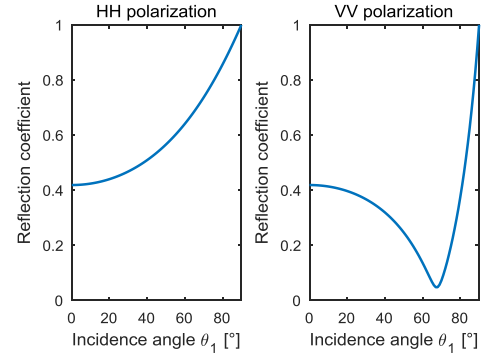$$I_d = F_d \cdot I_{sig} \cdot (\vec{N} \cdot \vec{L})^{F_b} \tag{11}$$

Fig. 3. Curves of Fresnel model which is for approximating the specular reflections of radar signals. Left: result for HH polarization. Right: result for VV polarization.
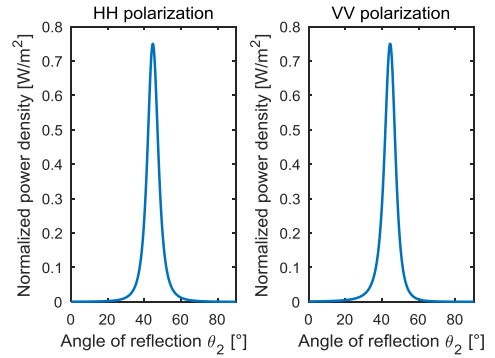

Fig. 4. Curves of SPM which is for approximating the diffuse reflections of radar signals on surfaces whose roughness is small compared to the signal wavelength. Left: result for HH polarization. Right: result for VV polarization.
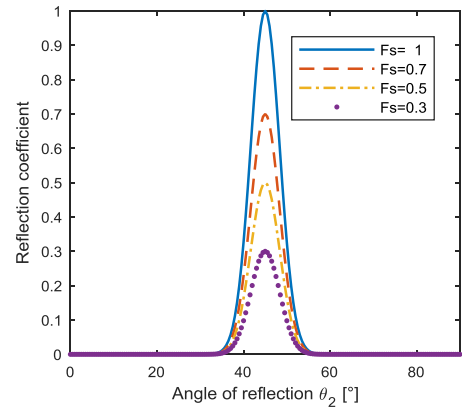

Fig. 5. Curves of RaySAR model for the specular reflections of radar signals.

where F_d is the diffuse reflection coefficient [0, 1], I_{sig} is the intensity of the incoming signal, $\vec{L}$ is the normalized signal vector pointing from the surface point to the SAR, and F_b is a surface brilliance factor [0, 1].

Figs. 3–6 show the reflection coefficient curves plotted according to the theoretical models and the approximate models. From the formulas and corresponding curves, we can draw the following conclusions. For specular reflection, the reflection coefficient
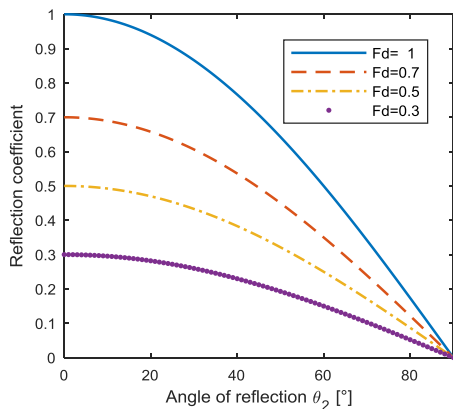
Fig. 6. Curves of RaySAR model for the diffuse reflections of radar signals.

of the specular model used in RaySAR is always F_s in the specular direction (see Fig. 5), rather than changing with the incident angle as the Fresnel model does (see Fig. 3). For diffuse reflection, the diffuse model used in RaySAR assumes that most scatterers have Lambertian reflection characteristic, that is, their reflection coefficients are only related to the incident angle but not to the reflection direction (see Fig. 6). This is not consistent with the SPM model (see Fig. 4) and the practice, because in the EM wave band of SAR operation, diffuse reflection is also related to the reflection direction. Hence, there are some differences between the theoretical models and the approximate models.

In the actual simulation, Dr. Stefan Auer does not directly use the approximate model provided by POV-ray in RaySAR, but skillfully uses the approximate specular reflection model to simultaneously simulate the specular reflection and diffuse reflection between objects. In this way, he makes the diffuse coefficient related to the reflection direction. In addition, he introduces diffuse and specular coefficients to combine the direct backscattering of radar signals with angle-dependent multiple reflections.

However, even with the above approaches, any definition of reflection parameters can only provide a rough approximation of real signal reflections due to the limitations of the simulation geometry and the reflection models. Especially, appropriate setting of the reflection coefficients and, hence, the absorption of radar signals is hard to be realized [14]. Since the original development of RaySAR is focused on simulating the geometrical distribution of signal responses, it is necessary to modify its EM reflection models in order to enable it simulate better SAR images.

III. METHOD

In this section, we describe in detail how to embed the deep neural network model into the simulation process while keeping the original simulation imaging model unchanged, so as to realize the learning of the reflection model suitable for this simulation model from the real SAR images. This content is divided into three parts, namely the modified SAR target image simulation framework, the construction of the input and output datasets of neural networks, and the design of DNN models.

A. Simulation Framework and Process

In the real SAR image, the number and intensity of the accumulated signals per pixel are uncertain, so it is difficult to separate the single reflection on the local surface element to train the DNN model. In order to avoid the above difficulties, a feasible idea is to embed it into the simulation method. It can be seen from Section II-A that the calculation of signal reflection intensity and the accumulation of signal intensity in the range-azimuth plane are alternately carried out in the SAR image simulation process based on ray tracing. In addition, the new generated reflection signal will also affect the next reflection as the incident signal, and each ray's reflection is independent of each other. Therefore, it is not a simple thing to embed the deep neural network model into SAR image simulation. It is necessary to split and recombine the original simulation methods reasonably and correctly to retain the original simulation imaging model.

Fig. 7 shows the proposed SAR image simulation framework in which the DNN model can be embedded to calculate the signal intensity. In this framework, the ray tracing of the radar signal rays (step 2), the calculation of the echo intensity (step 3) and the accumulation of the focusing rays in the range-azimuth plane (step4) are divided into three relatively separated steps and are combined in sequence. The calculation of the intensity of each single reflection is centralized in step 3 so that this calculation can be replaced by a DNN model. Next, the specific functions and requirements of each step in the simulation framework are introduced according to the simulation process.

First, the preliminary step is to prepare the target's 3-D model and material parameters, and set the SAR imaging parameters. If the target is composed of multiple materials and the parameters of these materials are continuous physical parameters, the DNN model can learn and fit a reflection model related to material parameters. If the target consists of only a few materials with few parameters, the DNN model can learn to fit the reflection model corresponding to each material category. In other words, whether the final fitted reflection model can deal with the changing material parameters is related to whether the material parameters forming the training dataset are continuous. SAR imaging parameters, such as resolution, pixel spacing, depression, azimuth, should be consistent with the real SAR images.

The second step is to illuminate the target and scene with the rays representing the radar signal, trace the reflection path of the rays according the RaySAR imaging model, and record the data generated when the ray interacts with the scene. The recorded data are the material and geometric information when the ray intersects with the local surface of the object, such as the material parameters of local surface, the incident angle and reflection angle of the ray, and are used to calculate the reflection intensity in step 3. In addition, the coordinates of the accumulated pixels of the focusing ray in the range-azimuth plane also need to be calculated and recorded for imaging in step 4.

The third step is to use the DNN model to calculate the intensities of reflected radar signal in the specified directions (SP and BK). In the following content, we call this DNN model NetEM. Since the input and output of NetEM are consistent with the
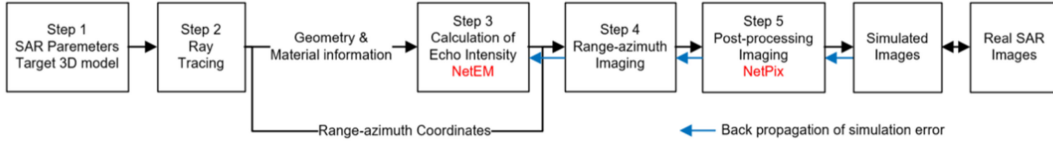
Fig. 7. Framework of SAR image simulation embedded with the DNN model.

original approximate reflection model, the function of NetEM is also limited to calculating the reflection coefficient. EM reflection model is a multi-input, single-output model. Therefore, theoretically from the perspective of hypothesis space, a fully connected network can complete the fitting task. However, there are some difficulties in the structure and connection mode. The structure of NetEM needs to be suitable for handling the complex reflection of a large number of rays during simulation. The reflected ray in the SP direction affects the next reflected ray as the incident ray. This means that the current output of NetEM should be affected by the previous output of the same model. The NetEM for calculating the intensity of rays corresponding to the same primary ray should be connected.

The fourth step is to accumulate the intensity of the focusing rays calculated by NetEM into the pixels of range-azimuth plane recorded in the step 2. It is important that all steps from this step must be differentiable so that the error between the resulting simulation image and the real image can be propagated back to NetEM for training. To illustrate the differentiability, this step can be considered as a two-layer partially connected neural network and its weights are fixed to one. The weight of one indicates that the radar signal represented by the first layer node is accumulated into the pixel represented by the second layer node. For different SAR images, the path of the rays in the scene are different, thus the pixel's positions of focusing rays are different, so the connection between the front and back layers of this network model should also change. Hence, the network model should be a dynamic model, although the "add operation" it does is mathematically easy to differentiate. Fortunately, the PyTorch library [22], which supports dynamic neural networks, provides a very effective tool for implementing this model and makes our work a lot easier.

The fifth step is the post-processing of the simulation SAR image output in the step 4. From the previous steps, it can be seen that this SAR image simulation method based on ray tracing can only simulate the position and intensity of the focusing radar signal, but cannot simulate the sidelobe and speckle noise caused by the coherent imaging of the SAR system. In addition, in order to simulate the fact that a single pixel emits multiple signals, the resolution of the sensor plane is often set to be greater than that of the real SAR image. Therefore, the image postprocessing step is required to add sidelobe and speckle noise to the simulated images, and reduce the resolution of the simulated image to the same resolution as the real image. The network model used in this step is called NetPix.

The above is a description of the functions of each simulation step and the problems to be solved. Next, we will explain in detail from the aspects of dataset construction, DNN model design and training.

B. Data Composition and Structure

1) Data Composition: The dataset can be divided into two parts according to the data source and the role of data in DNN models training. One part is the real SAR images of the target at multiple imaging angles. They are taken as the ground truth of NetPix, which are used to compare with the simulated images to calculate the simulation error. The other part is the simulation data recorded by the ray-tracing program in step 2, including the geometric and material information on the local surface and the cumulative position coordinates of the focusing rays in the range-azimuth plane. The former is the input data of NetEM, which is used to calculate the intensity of reflected ray in SP and BK direction. The latter is part of the input data of the imaging system in step 4, which is used for determining the cumulative position of the focusing rays.

In real SAR images, the distribution of pixel values is very uneven. Most of the pixel values are concentrated in $[0, 0.4]$, while some pixel values can be very large. The uneven distribution of data affects the training and generalization performance of DNN model. In order to solve this problem, we first use tanh function (12) to constrain the pixel value to $[0, 1]$, and then stretched it linearly to $[0, 255]$ to save it as a picture

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)}. \tag{12}$$

The geometric data involved in the EM reflection includes the θ_1, θ_2, θ_3 as illustrated in Fig. 2. The reflection intensity of EM wave is symmetrical distribution, that is, when the values of θ_1 and θ_2 are constant, the reflection coefficient at the θ_3 is equal to that at $2\pi - \theta_3$. In order to reduce the redundancy of the data, the angle θ_4 between the reflection direction and the incident direction is used instead of θ_3, so that the prior knowledge about symmetry is directly contained in the data. This helps to reduce the complexity of the reflection problem and accelerate the convergence speed of NetEM. The signal phase associated with the total length of the ray is also recorded. Considering the different value ranges of data, they are all normalized to $[-1, 1]$. Since there are few kinds of simulation materials in the scene, discrete values $(0, 1, 2)$ are directly used to represent the material categories.

2) Data Structure: The data structure often directly determines the form of the input and output data of the DNN model and further influences the design of the network architecture.
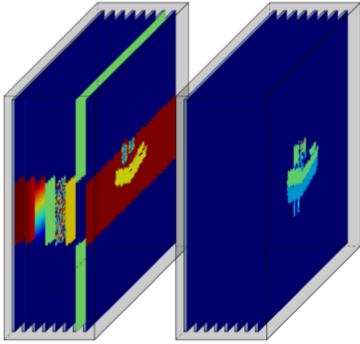
Fig. 8.   Data structure for storing reflection calculation data. Left: Data of the first reflection. Right: Data of the second reflection.

The data types involved in SAR image simulation are different from traditional DNN applications. Therefore, it is necessary to design the data structure well to effectively use the DNN to calculate EM reflection.

In the RaySAR, a 2-D array is used to organize the data recorded by the ray tracing. Each row of the array records the data of a reflection, such as signal intensity, echo position, number of reflections, whether it is specular or not. If this data structure is used, each training of the DNN model needs to retrieve all reflections associate with the pixels of the region of interest. This retrieval is very complex, retrieving not only the reflection of the focusing rays accumulated in the pixels, but also the reflection of the previous rays associated with each focusing ray. Hence, such data structure is not suitable as the input of DNN model.

To deal with the complexity of retrieval, a simple and straightforward idea is to store the data of each reflection directly in the accumulation position of the focusing rays in the range-azimuth plane. However, this method can only reduce the retrieval steps of the reflection data of focusing rays, but since the number of radar signals accumulated in each pixel is uncertain, it also brings the problem that the storage size of each pixel cannot be determined.

To solve the above problems, we propose a novel data structure which uses a 3-D array to store the reflection data. This 3-D array can be viewed as an image with multiple channels, shown in Fig. 8. Each channel of the image stores one type of reflection data (for example, incident angle). Each reflection of the ray is stored in a different "image" according to the number of reflections. If we set the maximum number of reflections of the ray to 5, we only need five such "images" to store all the reflection data. In addition to these contents, one of the most important innovations is that we use the emission position of the primary ray as the pixel coordinate of the stored data, rather than the cumulative position of the focusing ray. To solve the problem of data retrieval and imaging, it is sufficient to record the cumulative position of the focusing ray.

This data structure has the following advantages. First, it solves the problem of storage size uncertainty caused by the change of the number of rays in a single pixel in the range-azimuth plane. Second, in this data structure, the data of reflected rays corresponding to the same primary ray is stored in the same

pixel position. This feature brings a new perspective to the design of DNN, which greatly simplifies the model structure, reduces redundant calculation, and improves the speed of training. At the same time, the image postprocessing can be carried out smoothly. These contents will be covered in more detail in the next section.

C. Design and Training of DNN Models

In this section, we mainly introduce the design ideas and core structures of network models in the simulation framework, as well as their training methods. The specific super parameters of these models such as the number of convolutional layers and the size of the convolution kernel of the network model are compared and discussed in the following experiment section.

1) Architecture of the DNN Model for Calculating the Reflectivity: The architecture described in this part is the architecture of the DNN model in step 3, which is used to calculate the reflectivity of each reflection. Its function is exactly the same as the approximate model of EM calculation in the original simulation method. To simplify the presentation, it is then referred to as NetEM.

As mentioned earlier, the EM reflection model is essentially a multivariate function that takes incident angle, reflection angle, material, the intensity of the incident signal and other information as input, and outputs the intensities of reflected signal in the specific directions. More specifically, they correspond to a set of functional curves, as shown in Figs. 3–6. Hence, from the perspective of hypothesis space, the most basic fully connected neural network can fit the reflection models. In this network model, the input node is the geometry and material data, and the output node is the intensity of the reflected signal or reflection coefficients of specific directions.

However, from the perspective of the SAR image simulation, NetEM also needs to have the following functions. The first function is the ability to efficiently input simulation data and quickly complete reflection calculations. The simulation of a SAR image involves a lot of intensity calculation of a large number of reflected rays. Each pixel in the sensor plane emits multiple rays. Each time a ray is reflected, two new rays are generated. The second function is to output data in an organized data structure. As mentioned earlier, there is not only position correlation but also intensity calculation correlation among multiple reflected rays. The reflected ray in the SP direction affects the next reflected ray as the incident ray. Hence, the structure of the data output from the NetEM should be easily connected to the same model. In addition, each output of the network should facilitate the subsequent imaging processing.

Fortunately, the data structure described in Section III-B brings us a new perspective that enable us to design and use a better DNN structure to realize the above two functions. As shown in Fig. 9, when we use different channels of an "image" to store different types of simulation data, the calculation completed by the fully connection structure is changed to be performed on the channel dimension of the image. This structure is similar to that of the fully convolution network (FCN) [23], which converts the fully connected layers to convolutional layers. The difference is
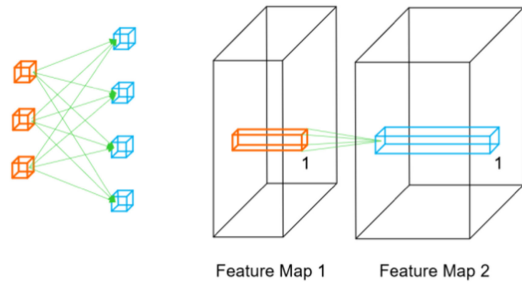
Fig. 9. Basic structure of NetEM model. In the image, the convolution operation with kernel size one is equivalent to the full connection operation in the channel direction.

that the size of the convolution kernel in NetEM is set to 1. Because the rays from adjacent emission positions do not affect each other when calculating reflection, this setup eliminates the spatial correlation between rays and ensures that the receptive field of each layer of neurons is always 1.

The input of this structure does not need to consider the size of the SAR image, retrieve the simulation data, or cut the original image into a small patch. It can directly input the whole image, thus greatly improving the efficiency of data input. In addition, the reflected intensity data output by this structure is naturally stored in the same position as the input data, that is, according to the emission position of the primary ray. This facilitates the connection between the NetEMs for each reflection calculation. Therefore, the structure of NetEM adopts the convolution layer with kernel size of one instead of the fully connected layer to complete the calculation related to single reflection.

According to the imaging model described in Section II, part A, the NetEM needs to calculate the intensity of the radar signal in the direction of SP and BK. There are two methods of dividing the reflection of EM signals on a surface. In RaySAR, according to the roughness of the reflecting surface, the reflection of EM signal in all directions is divided into specular reflection and diffuse reflection. According to the distance of EM wave propagation, EM reflection can be divided into near-field reflection between objects and far-field reflection between objects and antenna. In SAR image simulation, the influence of propagation distance on signal strength is greater than that of the roughness of the reflecting surface. Hence, we finally chose the latter method.

The specific structure of NetEM can be realized in three ways: one single network model with one single output node, one single network model with two output nodes, and two network models with one output node each. We compare these three implementations in the experiments section.

The type of output data of NetEM determines the connection mode between the models. If the output is the intensity of the reflected signal, then the output of current network needs to be connected to the next NetEM as the input data (shown in Fig. 10). If the NetEM's output is reflection coefficient, then the output of current network should be connected with the output of the next network, that is, multiplied together to calculate the intensity of the next reflected signal (shown in Fig. 11). Considering that the concept of reflection coefficient exists in the actual
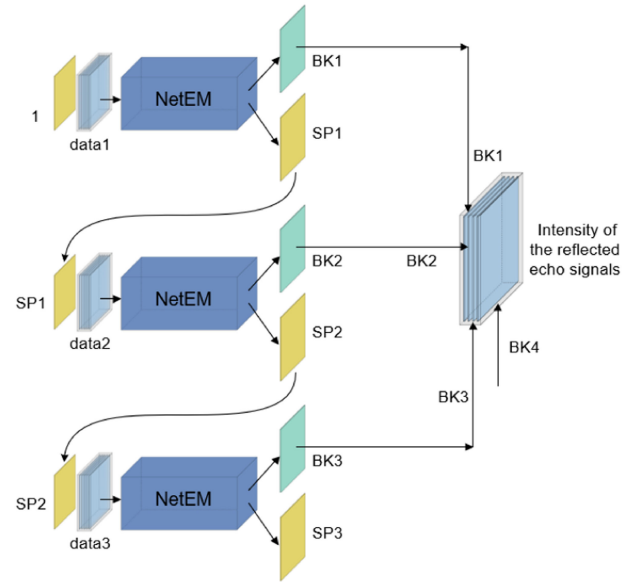

Fig. 10. Connection structure between NetEM model when the output is the intensity of the reflected signal. BK/SP: intensity of reflected signal in BK/SP direction.
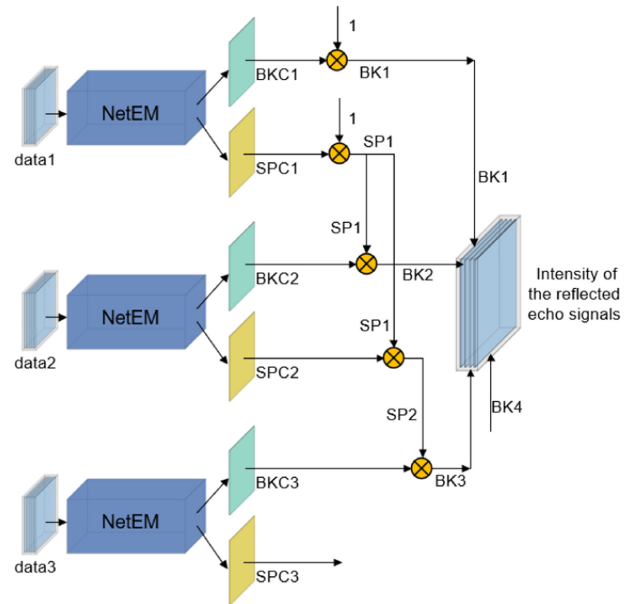

Fig. 11. Connection structure between NetEM model when the output is the reflection coefficient. BK/SP: intensity of reflected signal in BK/SP direction. BKC/SPC: reflection coefficient in BK/SP direction.

reflection of EM wave, and the output of both the theory and the approximate reflection model is also reflection coefficient, the latter NetEM network architecture is finally selected to increase the interpretability of the model and reduce the difficulty of the learning task.

2) Architecture of DNN Model for Image Postprocessing: The architecture described in this part is the architecture of the DNN model in step 5, which is used for image post-processing.

This model plays two main roles in the whole simulation framework. One is to add the same impulse response and window function as the real SAR system to the simulation image, so that the resolution and other parameters of the simulation image are consistent with the actual image. The second is to add speckle noise in the simulation image. Both impulse response and speckle noise are caused by the coherent imaging mechanism of the SAR system. This DNN model can also be regarded as an image degradation model, which does not generate new information, but loses part of the information in the image. To simplify the presentation, it is then referred to as NetPix.

It can be seen from the above that the functions of NetPix are much simpler than those of the models in most DNN application scenarios. From an operational perspective, NetPix just needs to convolve the input simulation image with two convolution kernels representing impulse response and speckle noise, respectively. Therefore, the most concise structure of NetPix is a convolutional neural network with only two convolutional kernels.

In the initial implementation, due to the need to register the simulated image with the real image before calculating the simulation error, the registration function is also assigned to NetPix. As a result, the implementation of this function requires that the structure of NetPix be at least five layers. The increase of layers means more adjustable parameters, wider hypothesis space, and the learning ability to perform complex functions. However, the improved learning capabilities of NetPix also means that the function of each step in the simulation framework is out of control. It is impossible to guarantee that the model in each step only completes its expected function, and does not make additional fitting. For example, NetPix may make too many additional adjustments to the simulation image obtained in the previous step, so that it can output high-quality simulation images only according to the fixed structural information such as the target contour. This reduces the information that would otherwise need to be learned by NetEM, resulting in a reduced generalization capability of the designed simulation method. In order to solve the above problems, we separate the image registration function from NetPix, and put it into the calculation of the loss function. This allows us to greatly simplify the structure of NetPix.

3) Loss Function: The loss function is used to measure the difference between the output simulation image and the real image during the training and testing process. Since our proposed simulation framework is differentiable and its output is directly an image, the loss function can be calculated directly using the simulation image and real image without any additional complex design.

Two commonly used methods to calculate the loss between images are mean absolute error (MAE) (13) and mean square error (MSE) (14). Among them, MSE is more sensitive to outliers. Because its penalty is squared, the loss of outliers can be very large. The calculation of MSE is easier. Since the size and distribution of the strong scattering points in the SAR simulation image are more important, the greater sensitivity of MSE to outlier or large values is not a disadvantage, but an advantage in measuring the simulation error of SAR image. Hence, the MSE

loss is used as the loss function of the whole framework

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \widehat{y_i}| \tag{13}$$

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \tag{14}$$

In the experimental dataset, each image of the target is manually cropped. Through comparison with the simulation image, we find that there is a position error caused by translation between the real image and the simulation image. Hence, the two images need to be registered before calculating the simulation error. According to [24], if the two images have the relationship of translation (15), the pixel shift of the two images can be calculated by the following fixed formulas. First, Fourier transform (FT) the two images (16). Second, calculate their cross-correlation matrix $\mathbf{P}$ (17) in the frequency domain. Then, the inverse FT (IFT) of $\mathbf{P}$ is a Dirac delta function centered on (x_0, y_0) [25], and it could be correctly captured by a simple IFT for $\mathbf{P}$. With this method, two images can be registered quickly. Although the registration accuracy of this method is only one pixel, it satisfies the requirement for calculating the loss value. The relevant formulas involved are as follows:

$$\mathbf{H}(x,y) = \mathbf{G}(x - x_0, y - y_0) \tag{15}$$

$$\widehat{\mathbf{H}}(u,v) = \widehat{\mathbf{G}}(u,v) \exp\{-i(ux_0 + vy_0)\} \tag{16}$$

$$\mathbf{P}(u,v) = \frac{\widehat{\mathbf{G}}(u,v)\widehat{\mathbf{H}}^*(u,v)}{\left|\widehat{\mathbf{G}}(u,v)\widehat{\mathbf{H}}^*(u,v)\right|} = \exp\{i(ux_0 + vy_0)\}. \tag{17}$$

For the rotation registration between the simulation image and the real image, we think that it can be regarded as the estimation problem of the target principal axis direction. At present, there have been some mature calculation methods to estimate the principal axis direction of the target. In our previous paper [17], a deep neural network was trained with simulation images to predict the principal axis direction of the target, and some results were obtained. However, the accuracy of these methods (5°–7°) still lags behind the interval (1°–2°) of azimuth labeled in dataset. Therefore, we think that the azimuth of the object labeled for each image is accurate, and there is no need to consider the registration problem caused by rotation.

4) Design and Training of DNN Models in Simulation Framework: The simulation framework contains two network models with different functions, and the two models can influence each other. Therefore, in order to ensure that each model can only complete its expected function without affecting the function of the other model, we design and adopt the following design process and training method.

Since the size of the convolution kernel used in NetEM is one, which naturally breaks the spatial correlation between the emission positions of rays, it is no longer necessary to divide the real image into small patches to increase the generalization ability of the model. This is the same as FCN, which also directly uses the whole image for training without considering the image size. Therefore, we only need to randomly divide the real image

and its corresponding "image" which stores multiple reflection data into batches, and then we can use them for model training.

Only when there is a translation relationship between the two images can the cross-correlation matrix be used to register them. At the beginning of NetEM training, because its parameters are randomly initialized, the image output of step 4 is almost random noise, so the final output cannot be registered with the real image. Therefore, in the first few epochs of NetEM training, we used the simulation image (just use the default values for simulation parameters) as the true value output of step 4, provide directional guidance on NetEM. (The simulation image obtained from the simulation does not need to be registered with the output image of step 4.) After the pretraining of the first five epochs, NetEM can basically build a rough reflection calculation model, and the image simulated by the network can be registered with the real image. In the subsequent epoch, it is no longer necessary to use the simulation image, but directly use the real SAR image as the final output true value to train NetEM for more precise adjustments.

Although the function of NetPix model is strictly limited by its structure, it can still differently adjust the signal intensity in SAR image. In the process of designing and training NetEM, only the fixed convolution kernel is used as NetPix for image post-processing. After training NetEM, keep the parameters of NetEM unchanged, and replace the NetPix model with the structure which has adjustable convolution kernels. If the impulse response convolution kernel is used to initialize the adjustable convolution kernel of the same function in NetEM, it is helpful to accelerate the convergence of the network. Finally, the well train NetEM and NetPix as well as the whole simulation framework can be obtained.

IV. EXPERIMENTS AND RESULTS

This section introduces experimental content, mainly including: the experimental data, the specific design of DNN structure, the visualization and interpretation of DNN models, the simulation results analysis and performance in the simulation of a complex target.

A. Experimental Data

The main simulation target of this article is the SLICY target. It is a precisely designed and machined engineering test target containing standard radar reflector primitive shapes, such as flat plates, dihedrals, trihedrals, and top hats. In another word, SLICY is a well-defined target for researchers to validate the functionality of their EM algorithm [26]. Fig. 12 shows the precise geometric model of SLICY built according to the description in [26].

There are many real SAR images of SLICY in the measured ground stationary SAR target dataset published by the MSTAR project [27]. Among them, there are 274 images with different azimuths at 15° depression angle and 288 images with different azimuths at 30° depression angle. The sensor collecting this dataset is a high-resolution spotlight SAR, which has a resolution of 0.3 × 0.3 m and works in the *X*-band. The image size is 54 pixels × 54 pixels, and the pixel spacing is 0.2 m.
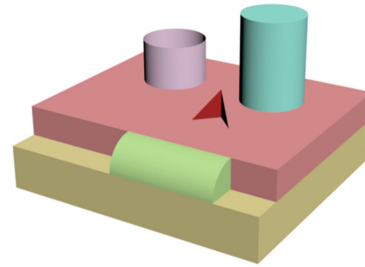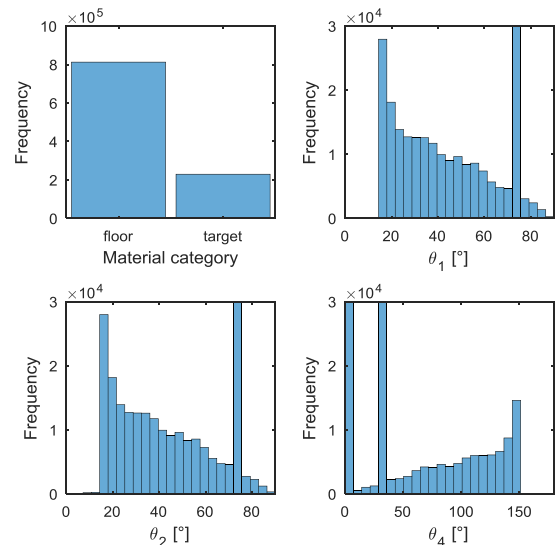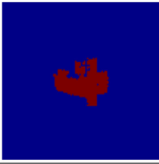

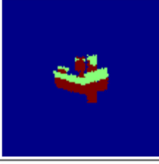Fig. 12.   Geometric model of SLICY target.
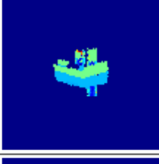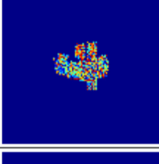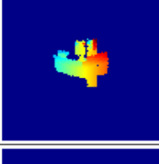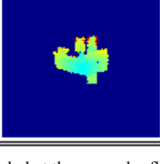

Fig. 13.   Distribution of various types of simulation data of SLICY at 15° depression angle.

Table I gives all kinds of simulation data recorded by the ray tracing program. The geometric and material data on the local surface are used to input into the NetEM model to calculate the EM reflectance in specific directions. Fig. 13 shows the distribution of these data within their respective values. Because the background is a large plane, the frequency of extremely individual angles is very high. In the value range of each angle, most of the angle samples exist. Hence, the training dataset is sufficient to ensure the generalization performance of the network to a certain extent. The position coordinates of the signal echo in the range-azimuth plane are used to accumulate the signal intensity into the range-azimuth imaging plane to form the SAR image. In order to match with the real SAR image, the azimuth and depression angle in the simulation are set to be the same as those of the real SAR images. In the last step of the simulation system, image postprocessing, the resolution and pixel spacing of the simulation image will be adjusted. Therefore, in the ray tracing system, the pixel spacing is set to 0.1 m which is smaller than that of the real images to increase the number of signal rays in each pixel.

Table II gives the division of the dataset. 20% of the images in 15° depression and 80% in 30° depression are used to train

TABLE I
SIMULATION DATA RECORDED BY THE RAY TRACING PROGRAM

Data Content	Pseudo Color Image of Simulation Data
whether the echo in antenna direction (SP) generated by radar signal reflection is blocked or not	
material category of the local surface element	
angle between the incident direction and the normal	
angle between the echo direction (BK) and the normal	
angle between the incident direction and the echo direction	
phase of the echo signal	
x-coordinate value of the signal echo in the range-azimuth plane	
y-coordinate value of the signal echo in the range-azimuth plane	

The table shows the simulation data recorded at the second reflection of radar signal.

TABLE II
DIVISION OF THE DATASET

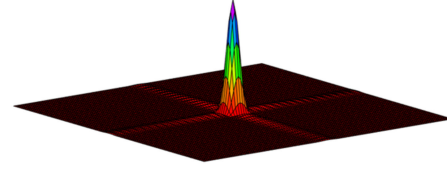| Depression Angel	15-degree	30-degree
training set	54	230
validation set	220	58

Fig. 14. Impulse response function and window function (spatial domain) for adjusting image resolution and adding sidelobe and speckle noise.
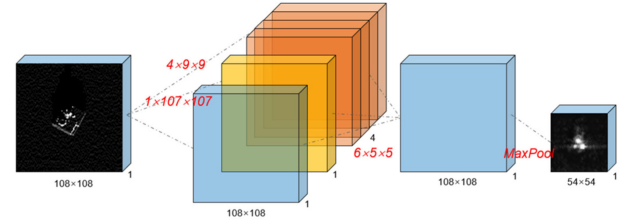

Fig. 15. Structures of NetPix. The convolution kernel with the size of $1 \times 107 \times 107$ can be the convolution kernel with fixed value shown in Fig. 14 or a convolution kernel with adjustable value.

DNN model. The remaining images are used to verify the DNN model.

B. Design of the Neural Network Structures

1) Network Structure of NetPix Model: The function of Net-Pix model in the simulation framework is to post-process the simulated SAR image. Specifically, its function is to adjust the resolution and add sidelobe and speckle noise to the simulation image. As mentioned in the method section, excessive image modification by NetPix may reduce the learning content of NetEM and hinders the back propagation of simulation error. Therefore, the important principle of NetPix structure design is that on the basis of guaranteeing the basic functions of the model, reduce its adjustable parameters as far as possible and limit its learning fitting ability.

The adjustment of image resolution and sidelobe effect caused by the coherent imaging principle of SAR system can be simulated by convoluting the image with impulse response function and window function. Hence, a convolution kernel with fixed value can accomplish these two main functions of NetPix. The speckle noise in SAR image is a multiplicative noise, so the convolution structure can also be used to add speckle noise to some extent. We construct the corresponding fixed convolution kernel according to the real SAR image parameters of SLICY target, as show in Fig. 14. The resolution of the impulse function is 0.3 m. The window function is Taylor window.

Fig. 15 shows the specific structures of NetPix. The whole model contains only two convolutional layers and corresponding activation functions. The first convolution layer consists of one convolution kernel with the size of $1 \times 107 \times 107$ and four small convolution kernels with the size of $1 \times 7 \times 7$, and uses the ReLU activation function to accelerate the convergence of the model. The large convolution kernel can be the convolution kernel with fixed value shown in Fig. 14 or a convolution kernel

TABLE III
STRUCTURES OF NETEM MODELS WITH DIFFERENT ACTIVATION FUNCTIONS

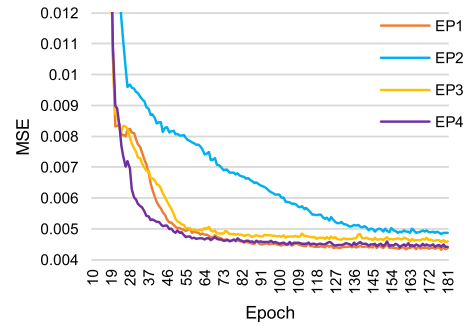| Layer No.	Kernel Size	Kernel Num	Stride	Activation Function	
1	1×1	16	1	ReLU	
2	1×1	32	1	ReLU	
3	1×1	64	1	ReLU	
4	1×1	32	1	ReLU	
5	1×1	2	1	None	EP1
				Sigmoid	EP2
				Tanh	EP3
				ReLU	EP4

Fig. 16. Loss curves of the network structures in Table III on the validation set.
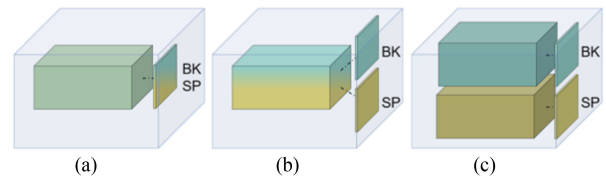

Fig. 17. Three different architectures for the NetEM model. (a) One single network model with one single output node (b) One single network model with two output nodes. (c) Two network models with one output node each.
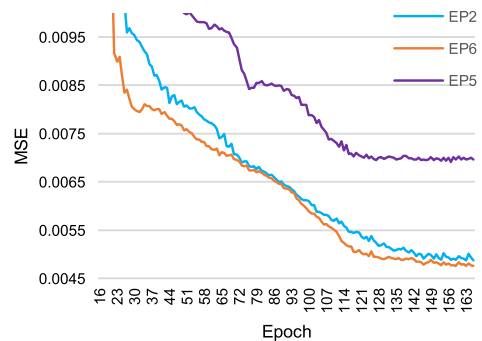

Fig. 18. Loss curves of the network architectures in Fig. 17 on the validation set.

with adjustable value. The second convolutional layer contains one convolution kernel with the size of $6 \times 5 \times 5$. Since tanh function is used to normalize the real SAR image, it is also used as the activation function in the last layer of NetPix to ensure that the output of the previous layer is the same as the signal intensity in the real SAR image. Excluding the large convolution kernel with the size of 107×107, the NetPix network model has only 478 trainable parameters, far less than the usual DNN model, and can achieve the expected function. The concise network structure and few trainable parameters help to ensure the generalization ability of the entire simulation framework.

2) Network Structure of NetEM Model: The function of NetEM is to learn the reflection model of EM scattering of radar signal on local surface element from real SAR images. As described in the methods section, the reflection model of radar signal on the local surface is actually a curve composed of reflection coefficients at all angles. This curve can be fitted by a fully convolutional network with convolution kernel size of 1×1. Therefore, in order to determine the optimal NetEM structure, we only need to experiment and compare the following contents: The activation function of the network, the number of models and the number of layers of the network. It should be mentioned again that in the following experiments comparing various structures of NetEM, only the fixed convolution kernel shown in Fig. 14 is used to replace NetPix for image post-processing. This ensures that the training and performance of NetEM are not affected by NetPix.

The activation function between layers has little influence on the curve fitting ability of the model. Therefore, ReLU activation function, which is faster in calculation, is directly selected as the interlayer activation function. The activation function of the last layer of the NetEM model determines the output range of the model. Therefore, we compare four networks with different activation functions given in Table III.

Fig. 16 shows the loss curves of the four networks on the validation set. It can be seen that all the four network structures can converge effectively. Even though EP2 which uses Sigmoid as activation function has the slowest convergence speed and the highest loss value, only the Sigmoid function can effectively limit the reflection coefficient calculated by the model to the range of $[0, 1]$, so as to make it conform to the actual physical law. Therefore, the activation function of NetEM's last layer is set to the Sigmoid function.

As shown in Figs. 1 and 10, NetEM needs to calculate the reflection coefficient in two directions: the reflection coefficient in the SP direction which is used to calculate the reflection between the target and the background, and the reflectivity in the antenna echo direction (BK) which is used to calculate the intensity of the signal echo. According to the output of NetEM, there are three different ways to construct NetEM as shown in Fig. 17. These three modes are: one single network model with one single output node, one single network model with two output nodes, and two network models with one output node each.

Fig. 18 shows the loss curves of these three network structures on the validation set. It can be seen that the final loss value of EP2 and EP6 is much smaller than that of EP5, and that of EP6 is slightly smaller than that of EP2. Therefore, EP6 structure is selected finally. With the increase of network output nodes and network models, the computational coupling degree of the whole

TABLE IV
STRUCTURES OF NETEM MODELS WITH DIFFERENT ACTIVATION FUNCTIONS

EP No.	Layer No.	Kernel Size	Kernel Num	Stride	Activation Function
EP6 BK/SP	1	1×1	16	1	ReLU
	2	1×1	32	1	ReLU
	3	1×1	64	1	ReLU
	4	1×1	32	1	ReLU
	5	1×1	1	1	Sigmoid
EP7 BK/SP	1	1×1	32	1	ReLU
	2	1×1	64	1	ReLU
	3	1×1	32	1	ReLU
	4	1×1	1	1	Sigmoid
EP8 BK/SP	1	1×1	16	1	ReLU
	2	1×1	32	1	ReLU
	3	1×1	64	1	ReLU
	4	1×1	32	1	ReLU
	5	1×1	32	1	ReLU
	6	1×1	1	1	Sigmoid
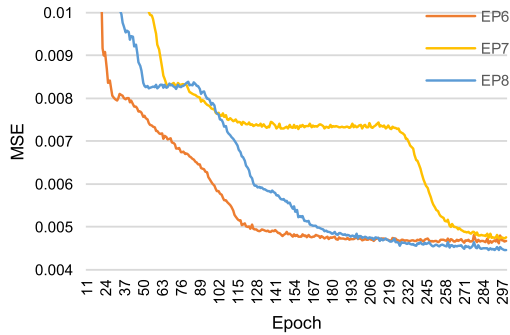
Fig. 19. Loss curves of NetEM models with different number of layers on the validation set.

Fig. 20. Curves of the reflection coefficient in the SP direction.

model in two directions decreases. Hence, it can be concluded that radar signal reflection models are much different in SP direction and BK direction. This conclusion is consistent with the fact that the near field scattering and far field scattering of EM waves follow different rules.

Finally, we compare the NetEM models with four layers (EP7), five layers (EP6), and six layers (EP8) respectively. Their structures are given in Table IV. Fig. 19 shows the loss curves of the three models on the verification set. EP7 has the slowest convergence rate and the highest loss. EP6 has the fastest convergence rate and smaller loss value than EP7. EP8 has a moderate convergence rate and the minimum loss of the three structures. Therefore, the fully connected network with a six-layer structure can fully fit the reflection model, and the number of layers of NetEM is set to six layers.

To sum up, the NetEM model and NetPix model that we finally designed are EP8 and Fig. 15 respectively.

C. Visualization of the Well-Trained Neural Network Models

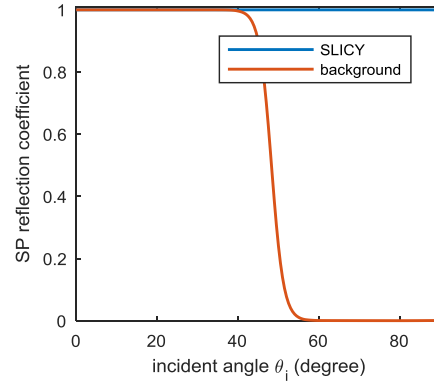In order to illustrate that the NetEM and NetPix in the simulation framework complete and only complete the designed functions, we visualized these two neural networks to increase the interpretability of their functions.

1) EM Reflection Models Fitted by NetEM: The expected function of NetEM is to fit the EM scattering model suitable for the SAR image simulation method from real SAR image. The specific function of the EM scattering model is to calculate the reflection coefficient in the specific directions according to the geometric and material information. Therefore, the reflection model fitted by the NetEM model can be simply illustrated by the curve drawn according to the input and output of NetEM. In order to draw the complete reflection curve, we set the data of various angles and materials, input them into the NetEM model, calculate the corresponding reflection coefficients, and then draw the corresponding curve. Although this method cannot indicate which parts of the drawn curve are supported by training data, it can easily and quickly draw the complete curve fitted by NetEM.

Fig. 20 shows the curves of the reflection coefficient in the SP direction when the materials are SLICY target and background. The reflection coefficient of radar signal on SLICY surface is close to one. This is the same as the original specular reflection simulation model in RaySAR (see Fig. 5). This means that there is almost no attenuation of radar signal intensity when reflected on SLICY surface, and the reflection of radar signal on SLICY is close to specular reflection. According to [28], the material of SLICY target is metal, so this conclusion is exactly the same as the reality. The reflection coefficient of the signal on the background decreases with the increase of the incident angle. This trend is consistent with the nature of the background material (growing vegetation) and is similar to that of VV polarization in Fresnel specular model, as shown in Fig. 3.

Fig. 21 shows the curves of the reflection coefficient when the radar signal is reflected back to the SAR antenna (BK). With the change of incident angle, the reflection coefficient curve also changes regularly. For the material of SLICY, the reflection coefficient reaches the minimum at the incident angle θ_1, increases gradually on both sides of the incident angle, and gradually increases to the maximum on the other side of the normal. For background material (only growing vegetation), its reflection coefficient decreases rapidly on the other side of the normal. When the reflection angle θ_2 is greater than 0° and less
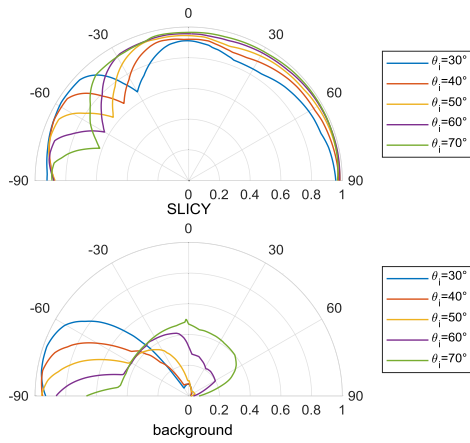
Fig. 21. Curves of the reflection coefficient in the BK direction. Top: The material of SLICY. Bottom: The material of background (only growing vegetation).
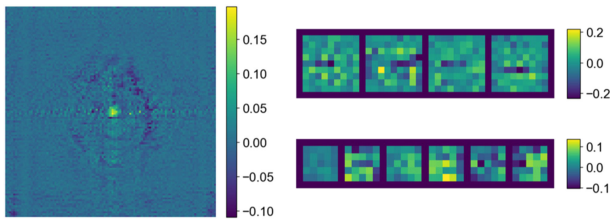

Fig. 22. Large and small convolution kernels of the NetPix model in Fig. 15.

than 90°, the reflection coefficient increases with the increase of incident angle for both of the materials. This law is the same as that of HH polarization of Fresnel model in Fig. 3. This indicates that the far field reflection calculated by NetEM is related to the incidence angle.

2) Convolution Kernels of NetPix: The expected function of NetPix is to add side lobe and speckle noise caused by the coherent imaging mechanism of SAR to the simulation image. As mentioned above, these two functions can be simulated simply by using convolution kernel. In addition, because the number of layers of NetPix model is very small, the function of NetPix can be interpreted just by looking at the content of convolution kernel. From the convolution kernel shown in Fig. 22, it can be seen that the small-sized convolution kernel of the two layers learns some local basic structures of SLICY to concentrate the reflection of these structures in one or two pixels, while the large convolution kernel fits the convolution kernel corresponding to the impulse response. Therefore, these two convolution kernels are in accordance with the expectation.

D. Comparison Between the Simulation Images, the Real Images and the SLICY Model

Fig. 23 shows the curve of the MSE between the simulation images and the real images on the verification set. In the figure, the convergence value of MSE is 0.003253. This loss value indicates that the shape size, intensity and position distribution of the scattered points in the simulation images are close to the real images, that is, the simulation images are very similar to the real images. Table V gives the comparison between the simulation
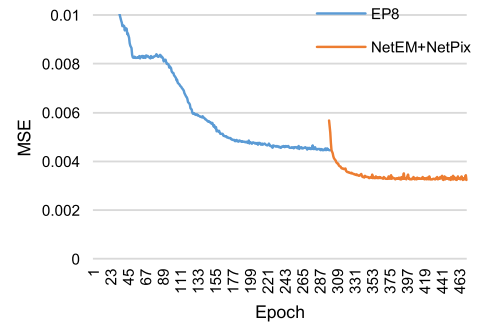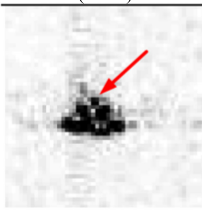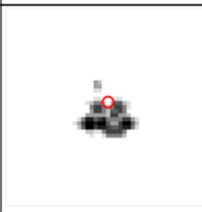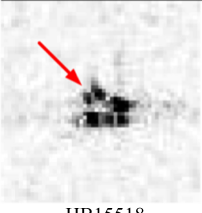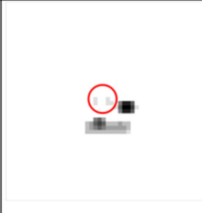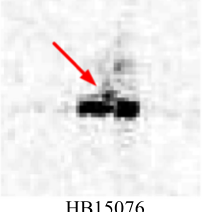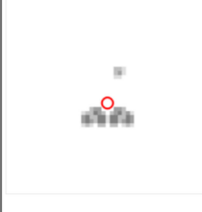

Fig. 23. Curve of the MSE between the simulation images and the real images on the verification set.

TABLE V
COMPARISON OF TWO KINDS OF SIMULATION IMAGES AND REAL IMAGES

images and the real images. In addition to the images simulated by the proposed method, the other set of simulated images are simulated by frequency asymptotic code for electromagnetic target scattering (FACETS) [26].

FACETS is a commercial high-frequency EM code developed by Thales U.K. Limited and its excellent numerical simulation capability and the quality of simulation images are verified by Defence R&D Canada [29]. Although the images simulated by FACETS are of high quality, it can be seen from the comparison of Table V that the images simulated by FACETS still lack some key scattering points compared with the real images. However, these scattering points are well simulated by the proposed method. This is because FACETS uses a modular structure to compute various scattering processes. The modular structure makes FACETS unable to calculate the EM reflection between the scattering primitives and results in the missing of

Fig. 24.   Geometric structure of T-72 main battle tank.

Fig. 25.   CW-SSIM index curve of the simulated image datasets obtained by the two simulation methods. (a) 15° depression. (b) 17° depression.

some scattering points in the simulation image. In addition, the locations of scattering primitives on the target must be manually identified and fed into the FACETS's EM code. For complex targets, the nominations of all the scattering primitives could be quite labor intensive. Since the proposed method directly uses radar signal rays to illuminate the entire imaging scene, the above problems do not exist.

We further draw Fig. 26 in appendix to show more comparisons between the simulated images and the real images and illustrate the assistance of image simulation in SAR image interpretation. In the figure, we give the simulation images before and after the image post-processing in step 5, in which the image before postprocessing has two quantized results: truncation quantization and peak quantization. The relevant scattered points are marked with red circles, so that the corresponding relationship between scattered points in the real image and geometric structure can be clearly obtained. By comparing the images, it is easy to identify which structure in the target causes each scattered point in the real image.

E. Performance in the Simulation of a Complex Target

In order to further verify the performance of the proposed method in simulating SAR images of targets with more complex structures and the improvement in the similarity of the simulated images compared with the original RaySAR simulation method, we used the two methods to simulate the SAR images of another main target in MSTAR dataset, T-72 main battle tank, for comparison and analysis. The geometric model of T-72 main battle tank is shown in Fig. 24. Compared with SLICY, it contains more geometric structures and combinations, which is complex enough to verify the performance of the proposed method in simulating complex targets. In addition, the complex-wavelet structural similarity (CW-SSIM) [30] index between the simulated images and the real images was calculated respectively to quantitative measure the improvement of the similarity of the simulation image of the whole T-72 dataset. Fig. 25 shows the CW-SSIM index curves of the simulation image datasets. It can be clearly seen that our method not only has good performance in simulating complex targets, but also has a great improvement in the similarity of simulation images compared with those obtained by RaySAR simulation method which directly uses the EM approximate calculation models.
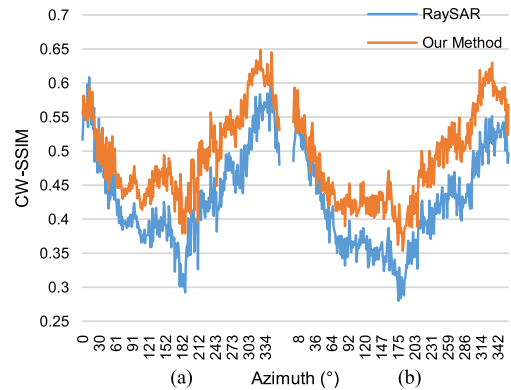
V. Conclusion

Aiming at the problem that the EM reflection approximate models adopted in the SAR image simulation method based on ray tracing is inconsistent with the actual situation and lack of adjustability, we propose a simulation framework that embeds DNN models into the simulation process to calculation EM reflection, which solves the problem of poor interpretability and weak generalization ability of DNN models. This method can directly learn and fit the EM reflection models from real images, and improve the similarity of the simulation image. The experimental results show that the proposed method is correct and effective. The EM reflection model fitted by the DNN model is consistent with the material characteristics of target and background. The convolution kernel of the post-processing model is also consistent with the expected form.

Compared with other methods of applying DNN model to the SAR image simulation, this method deeply analyzes and decomposition of the original simulation method, and decomposes the originally very complex simulation principle and process into several steps with clear functions. Therefore, the DNN model in each step has a very specific function, so as to enhances the interpretability and generalization ability of the model. During the implementation of this framework, the difficulties caused by complex computational data and complex model connection are overcome through the ingenious design of the data structure and network model architecture, thus effectively solving the problem of embedding DNN model into the simulation process while keeping the original basic imaging model and simulation principle unchanged.

The method proposed in this article can be further applied to the exploration of material properties and material classification, and can reduce the modeling cost of complex materials. More importantly, it provides a more in-depth and novel perspective for the combination of deep learning method and image simulation. In addition, the research ideas and techniques of this article provide a very novel and important entry point for the construction of a fully differentiable SAR image simulator, which will be further explored and improved in our next work.
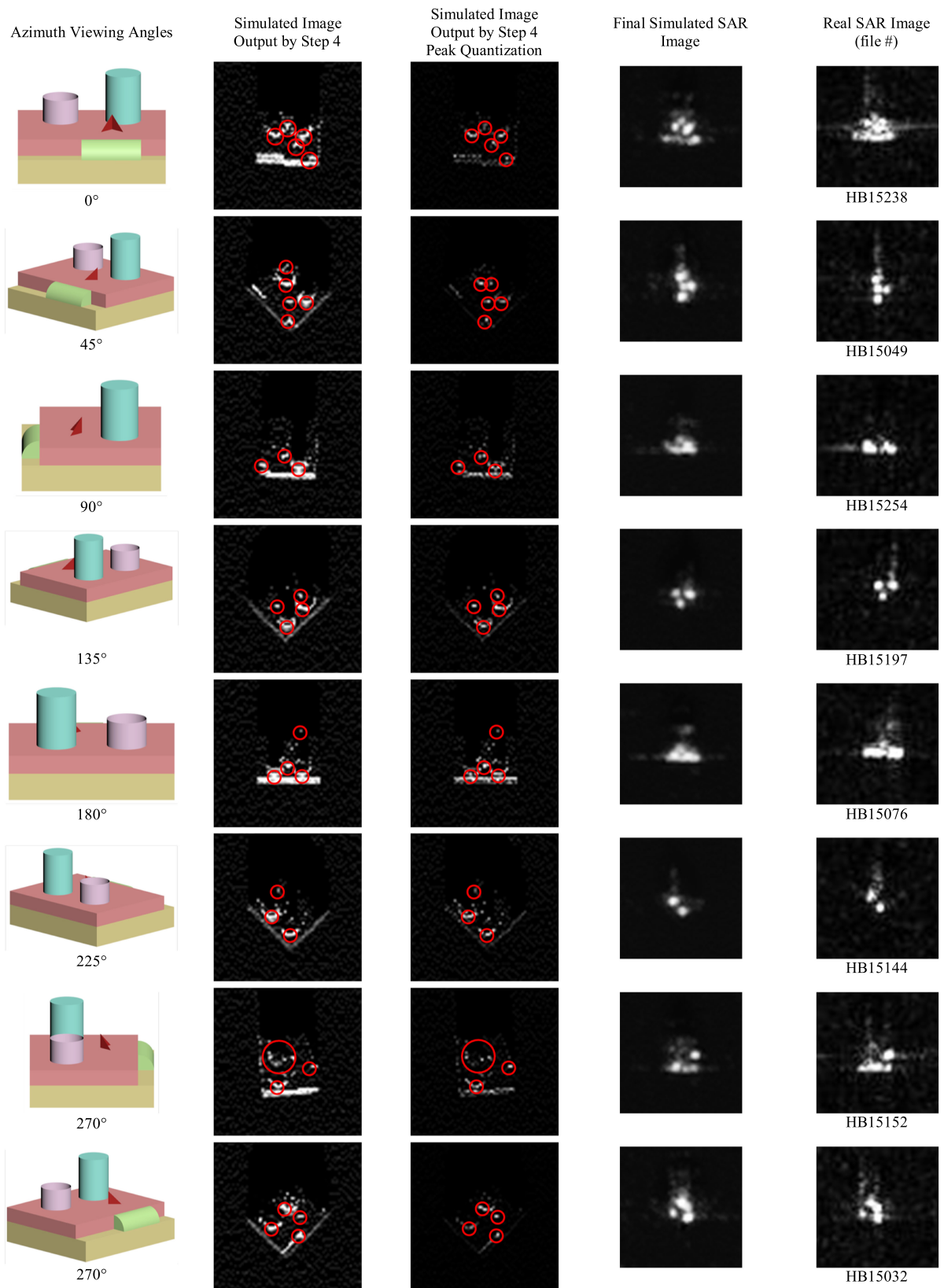
APPENDIX

Fig. 26.　Comparison between simulated and real SAR images at 15-degree elevation angle.

REFERENCES

[1] S. Auer, S. Hinz, and R. Bamler, "Ray-tracing simulation techniques for understanding high-resolution SAR images," *IEEE Trans. Geosci. Remote Sens.,* vol. 48, no. 3, pp. 1445–1456, Mar. 2009.

[2] J. C. Holtzman, V. S. Frost, J. L. Abbott, and V. H. Kaupp, "Radar image simulation," *IEEE Trans. Geosci. Electron.,* vol. 16, no. 4, pp. 296–303, Oct. 1978.

[3] Q. Song, H. Chen, F. Xu, and T. J. Cui, "EM simulation-aided zero-shot learning for SAR automatic target recognition," *IEEE Geosci. Remote Sens. Lett.,* vol. 17, no. 6, pp. 1092–1096, Jun. 2019.

[4] M. Wohlers, S. Hsiao, J. Mendelsohn, and G. Gardner, "Computer simulation of synthetic aperture radar images of three-dimensional objects," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 16, no. 3, pp. 258–271, May 1980.

[5] D. Brunner, "Advanced methods for building information extraction from very high resolution SAR data to support emergency response," Ph.D. dissertation, Dept. Inf. Eng. Comput. Sci., Univ. Trento, Trento TN, Italy, 2009.

[6] K. Ji, A. Zhang, H. Zou, and W. Sun, "Simulation of SAR images of ground vehicles." in *Proc. MIPPR: Automat. Target Recognit. Image Anal., Int. Soc. Opt. Photon.*, 2009, vol. 7495, p. 749535.

[7] Y. Chang, C. Chiang, and K. Chen, "SAR image simulation with application to target recognition," *Prog. Electromagn. Res.-Pier,* vol. 119, pp. 35–57, 2011.

[8] C. Z. Dong, L. P. Hu, G. Q. Zhu, and H. C. Yin, "Efficient simulation method for high quality SAR images of complex ground vehicles," *J. Radars,* vol. 4, 2015, pp. 351–360.

[9] F. Xu and P. Wang, "Analytical modeling of rough surface SAR images under small perturbation approximation," *IEEE Trans. Geosci. Remote Sens.,* vol. 55, no. 7, pp. 3694–3707, Jul. 2017.

[10] G. Franceschetti, M. Migliaccio, D. Riccio, and G. Schirinzi, "SARAS: A synthetic aperture radar (SAR) raw signal simulator," *IEEE Trans. Geosci. Remote Sens.,* vol. 30, no. 1, pp. 110–123, Jan. 1992.

[11] G. Franceschetti, M. Migliaccio, and D. Riccio, "SAR raw signal simulation of actual ground sites described in terms of sparse input data," *IEEE Trans. Geosci. Remote Sens.,* vol. 32, no. 6, pp. 1160–1169, Nov. 1994.

[12] Y. Zhang, C. Ding, X. Qiu, and F. Li, "The characteristics of the multipath scattering and the application for geometry extraction in high-resolution SAR images," *IEEE Trans. Geosci. Remote Sens.,* vol. 53, no. 8, pp. 4687–4699, Aug. 2015.

[13] S. J. Auer, R. Bamler, and P. Reinartz, "RaySAR-3D SAR simulator: Now open source," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2016, pp. 6730–6733

[14] S. J. Auer, "3D synthetic aperture radar simulation for interpreting complex urban reflection scenarios," Ph.D. dissertation, Dept. Remote Sens. Technol., Techn. Univ. Mnchen, Munich, Germany, 2011.

[15] J. Guo, B. Lei, C. Ding, and Y. Zhang, "Synthetic aperture radar image synthesis by using generative adversarial nets," *IEEE Geosci. Remote Sens. Lett.,* vol. 14, no. 7, pp. 1111–1115, Jul. 2017.

[16] L. Liu, Z. Pan, X. Qiu, and L. Peng, "SAR target classification with CycleGAN transferred simulated samples," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 4411–4414.

[17] S. Niu, X. Qiu, B. Lei, C. Ding, and K. Fu, "Parameter extraction based on deep neural network for SAR target simulation," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4901–4914, Jul. 2020.

[18] F. Xu and S. Fu, "Modeling EM problem with deep neural networks," in *Proc. IEEE Int. Conf. Comput. Electromagn.*, 2018, pp. 1–2.

[19] L. Li, L. G. Wang, F. L. Teixeira, C. Liu, A. Nehorai, and T. J. Cui, "DeepNIS: Deep neural network for nonlinear electromagnetic inverse scattering," *IEEE Trans. Antennas Propag.,* vol. 67, no. 3, pp. 1819–1825, Mar. 2018.

[20] L. Tsang, J. A. Kong, and K.-H. Ding, *Scattering of Electromagnetic Waves: Theories and Applications*. Hoboken, NJ, USA: Wiley, 2004.

[21] G. Franceschetti and D. Riccio, "Fractal models for scattering from natural surfaces," in *Scattering.* Amsterdam, The Netherlands: Elsevier, 2002, pp. 467–485.

[22] A. Paszke *et al.*, "Automatic differentiation in Pytorch," 2017.

[23] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[24] X. Geng and W. Yang, "Cyclic shift Matrix—A new tool for the translation matching problem," *IEEE Trans. Geosci. Remote Sens.,* vol. 57, no. 11, pp. 8904–8913, Nov. 2019.

[25] H. Foroosh, J. B. Zerubia, and M. Berthod, "Extension of phase correlation to subpixel registration," *IEEE Trans. Image Process.,* vol. 11, no. 3, pp. 188–200, Mar. 2002.

[26] S. Wong, "High fidelity synthetic SAR image generation of a canonical target," Defence Res. and Devel., Ottawa, Canada, 2010.

[27] T. D. Ross, S. W. Worrell, V. J. Velten, J. C. Mossing, and M. L. Bryant, "Standard SAR ATR evaluation experiments using the MSTAR public release data set," *Proc. SPIE Algorithms Synth. Aperture Radar Imag.*, vol. 3370, pp. 566–574, Sep. 1998.

[28] R. L. Moses, L. C. Potter, and I. J. Gupta, "Feature extraction using attributed scattering center models for model-based automatic target recognition (ATR)," Ohio State Univ., Columbus, OH, USA, 2005.

[29] S. Wong, "Validation of the electromagnetic code FACETS for numerical simulation of radar target images," Defence Res Develop., Ottawa,Canada, 2009.

[30] M. P. Sampat, Z. Wang, S. Gupta, A. C. Bovik, and M. K. Markey, "Complex wavelet structural similarity: A new image similarity index," *IEEE Trans. Image Process.,* vol. 18, no. 11, pp. 2385–2401, Nov. 2009.

Shengren Niu received the B.Eng. degree in electronic engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2016. He is currently working toward the Ph.D. degree at the Institute of Electronics, Chinese Academy of Sciences, Beijing, China.

His research interests include SAR image simulation, deep learning, and neural rendering.

Xiaolan Qiu (Senior Member, IEEE) received the B.S. degree in electronic engineering and information science from the University of Science and Technology of China, Hefei, China, in 2004, and the Ph.D degree in signal and information processing from the Graduate University of Chinese Academy of Sciences, Beijing, China, in 2009.

Since 2009, she has been with the Institute of Electronics, Chinese Academy of Sciences, Beijing, China. In 2011, she was a Visiting Scholar with the University of Siegen, Siegen, Germany. She is currently a Research Fellow with IECAS Beijing and also with IECAS Suzhou. Her research interests include SAR imaging, SAR simulation, SAR image interpretation, and SAR 3-D vision.

Bin Lei received the B.Sc. degree in electrical engineering from Tsinghua University, Beijing, China, in 2000, and the M.Sc. and Ph.D. degrees in signal and information processing from the Institute of Electronics, Chinese Academy of Sciences, Beijing, China, in 2003 and 2014, respectively.

He is currently a Professor with the Institute of Electronics, Chinese Academy of Sciences. His research interests include remote sensing data processing and system implementation.

Kun Fu received the B.Sc. and Ph.D. degrees from the National University of Defense Technology, Changsha, China, in 1995 and 2002, respectively.

He is currently the Head and a Professor with the Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, Institute of Electronics, Chinese Academy of Sciences, Beijing, China. He has authored two books and more than 60 refereed publications. His research interests include geospatial data organization and visualization, computer vision, and remote sensing image interpretation.