Journal of Physics: Conference Series

PAPER • OPEN ACCESS

Research on SAR Image Target Recognition Based on Convolutional Neural Network

To cite this article: Fan Xinyan and Zhu Weigang 2019 *J. Phys.: Conf. Ser.* **1213** 042019

Research on SAR Image Target Recognition Based on Convolutional Neural Network

Fan Xinyan[1,a] **Zhu Weigang**[2]

[1]Space Engineering University, Beijing
No.1, Bayi Road, Yanqi Town, Huairou District, Beijing

[2]Space Engineering University, Beijing
No.1, Bayi Road, Yanqi Town, Huairou District, Beijing

[a]492773427@qq.com

Abstract: A synthetic aperture radar (SAR) automatic target recognition can effectively improve the utilization efficiency of SAR image data. In order to improve the generalization and accuracy of SAR image target recognition, a convolutional neural network model is proposed to be applied to SAR image target recognition. Firstly, the network structure is designed for the characteristics of SAR image. Secondly, the introduction of dense block network structure improves the generalization performance of the model. Finally, Dropout reduced the computational complexity of the network and improved the generalization performance of the model. Experimental data were obtained from the United States Moving and Stationary Target Acquisition and recognition database. Experimental results of 10 types of target recognition showed that the overall recognition rate of the improved convolutional neural network is 99.18%. The convolutional neural network model proposed in this study improves the accuracy and generalization of the network, and is an effective method for target recognition of SAR images.

1. Introduction

SAR is extremely valuable in both military and civilian applications considering its all-day, all-weather and high-resolution features, and can effectively identify camouflage and penetrating cover. SAR target recognition, as an important part of SAR image interpretation, is a long-term research difficult problem for scholars all over the world. With the application of high-resolution SAR, the imaging area has been expanding, and different imaging modes have appeared one after another. The traditional interpretation of human interpretation faces enormous challenges. There are problems such as slow speed, heavy workload, and misjudgment. It is urgent to develop intelligent interpretation technology.

The Convolutional Neural Network (CNN) uses the original image as input, and can effectively learn the corresponding features from a large number of samples, which can avoid complex feature extraction process. Since the convolutional neural network can directly process two-dimensional images, it has been widely used in image processing and has obtained many research results. The network extracts more abstract features from the original image through a simple nonlinear model, and requires only a small amount of human involvement throughout the process[1]. At present, the target recognition algorithm based on convolutional neural network has been widely applied to SAR image target recognition, and has achieved a good Probability of Correct Cognition (PCC). Literature [2] uses CNN for SAR image target recognition and tests on the MSTAR dataset. The results show that CNN can greatly improve the recognition rate. For the problem of small sample and over-fitting of MSTAR

IOP Conf. Series: Journal of Physics: Conf. Series **1213** (2019) 042019 doi:10.1088/1742-6596/1213/4/042019

datasets, Li[3] uses autoencoder to pre-train the network, and Ding et al.[4] increase the number of samples by image transformation of SAR images. In order to further improve the recognition performance of CNN, some scholars improved the network structure. Z.Z.Tian [5] introduced the class separability measure into the error cost function, improved the class distinguishing ability of CNN, and then used Support Vector Machines(SVM) to realize feature classification. The experimental results show that the method is effective. In [6], the convolutional layer is used to replace the fully connected layer in CNN, which effectively suppresses the over-fitting problem, reduces the number of parameters, and further improves the recognition rate. However, convolutional neural networks require a large number of samples for training, and in the actual application process, the acquisition of a large number of tagged samples is a time-consuming and laborious task. Therefore, reducing the demand for CNN on the number of samples has gradually become a research hotspot.

In order to further improve the recognition accuracy and generalization of CNN for SAR target recognition, this paper proposes a new network structure based on CNN and dense block. Firstly, the network structure is designed for the characteristics of SAR image. Then, the dense block network structure is introduced to make different levels of features can be reused; finally, Dropout is used to reduce the amount of computation of the network and improve generalization. Experiments verify the effectiveness of the network model.

2. Convolutional Neural Networks

The deep convolutional network is a special structure of deep neural network. The first few layers are composed of a convolution layer and a pooling layer. The latter layers are fully connected layers. The working principle is that the convolutional layer learns different features, and the pooling layer aggregates the spatial domain shape into the high-dimensional feature space, and the multi-layer alternate convolution + pooling can learn the hierarchical feature representation. The role of the final fully connected layer is to learn a classifier in the high dimensional feature space.

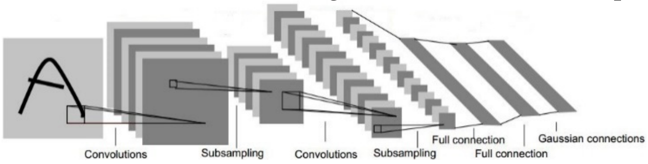

Figure1. Convolutional neural network model

As shown in Figure 1, the standard CNN is a feedforward neural network model, usually composed of convolutional layer, pooling layer (subsampling layer), fully connected layer, and output layer. The convolution layer uses a sparse connection to the CNN of the input image. There are many convolution kernels (convolution filters) in the convolutional layer, and each convolution kernel shares the same parameters. The convolution kernel acts as a sliding convolution window on the whole receptive field, and convolves the input image to obtain the feature map, which completes the extraction of the local features of the input image. The specific idea of the pooling operation is to replace the overall output of the network at that location with the overall statistical characteristics of the adjacent outputs at a certain location. The pooling operation can reduce the sensitivity of the neural network model to the displacement and deformation of the image, increase the robustness of the feature, and also aggregate the low-level texture features of the image into high-level semantic features to some extent.

After the convolutional layer is alternately connected to the pooled layer, the fully connected layer is accessed. The fully connected layer is essentially a perceptron that classifies or returns input data. In CNN, the parameters of the convolution kernel and the fully connected layer are trained by a BP algorithm with stochastic gradient descent (SGD). During the training process, dropout[7] is widely used as a regularization scheme to effectively avoid over-fitting. Dropout refers to temporarily invalidating the weights of some hidden layer nodes in a training process, thereby reducing the joint adaptability between network nodes and achieving better results.

3. Densely Connected Convolutional Networks

Liu Z et al.[8] proposed Densely Connected Convolutional Networks(DenseNet), whose core structure is dense block, as shown in Figure 2.

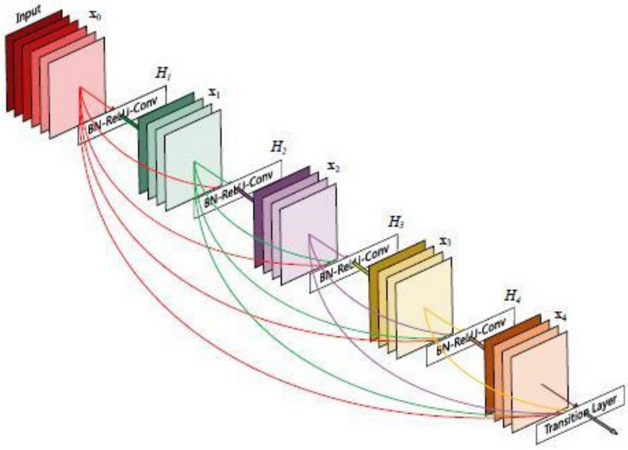

Figure2． Dense block structure

In the dense block, the output of all layers before each layer is used as input. For the traditional convolutional network of the L layer, the number of connections is L, and for DenseNet, the number of connections is L(L+1)/2. Suppose the image input is x_0, the L layer output is x_l, the network has L layer, and each layer contains a nonlinear transformation $H_l(\cdot)$, which can be Batch Normalization (BN), Rectified Linear Units(ReLU), pooling or convolutional layer, between them relationships can be expressed as:

$$x_l = H_l([x_0, x_1, \cdots x_{l-1}]) \qquad (1)$$

DenseNet has several compelling advantages: they can alleviate the vanishing gradient problem, enhance feature propagation, encourage feature reuse, and dramatically reduce the number of parameters.

4. SAR Target Recognition Based on Convolutional Neural Network

In view of the fact that SAR image data is relatively small and sensitive to observation conditions, it is easy to overfitting by training CNN with SAR data directly. Because CNN has too many free parameters, the training samples are not sufficient which leads to a serious overfitting. Therefore, this paper optimizes the convolutional neural network structure by means of dense connection. By using the dense block structure, the feature map of each convolution block is superimposed, so that the feature maps of each layer inside the dense block can be fully applied. Thereby reducing the number of feature maps in each convolutional layer and reducing network parameters. The convolution kernel size of the first and second convolutional layers in each dense block is set to 5 × 5 and 3 × 3, because the convolution kernel combination of different sizes is more favorable for the extraction of different scale features. At the same time, it is beneficial to improve the diversity of subsequent dense block network merging features.

The overall architecture of the proposed in this study is depicted in Fig. 3, which is composed of five convolution layers, three max pooling layers and 1 fully connected layer. The pooling layer adopts the max pooling, with a pooling size of 2 × 2 and a stride of 2 pixels. The ReLU nonlinearity is applied to every hidden convolution layer. The Softmax nonlinear function acts on the output node of the fully connected layer.

The 128×128 input image was filtered by 32 convolution filters of size 7×7 in the first convolution layer, resulting in 32 feature maps of size 62×62. After the first pooling layer, their sizes become 31×31. The first pooling layer's outputs are sent into the second convolution layer, which has a convolution filter size of 5×5 and zero pad of 2, leading to 64 feature maps of size 31×31. The third convolution layer, which has a convolution filter size of 3×3 and zero pad of 1, leading to 64 feature maps of size 31×31.After the second pooling layer, their sizes become 16×16. The filter size of the fourth convolution layer is 5×5 and zero pad of 2, The filter size of the fifth convolution layer is 3×3 and zero pad of 1, producing 128 feature maps of size 16 × 16, which becomes 8×8 after pooling. The dropout regularization technique is used in the third pooling layer. Then, after passing through a fully connected layer, it is output to 10 nodes, and each node is processed by Softmax to output the probability that the target belongs to each category.

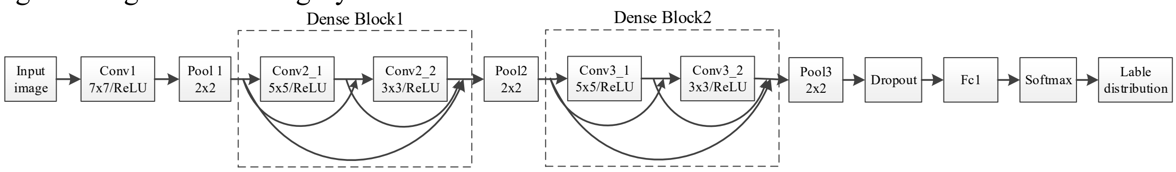

Figure3． network structure diagram

4.1 ReLU Activation Function

The ReLU activation function can effectively alleviate the problem of vanishing gradient or exploding gradient, and accelerate the Rate of convergence of the network to some extent. Therefore, the ReLU activation function is added after each convolutional layer.

Backward propagation error information in neural network learning, its gradient is prone to instability with the increase of network depth, especially the traditional nonlinear activation function such as sigmoid or tanh function, the gradient presents a spike at the origin, leaving the origin not far away is saturated immediately. When the multi-layer network returns, the gradient is multiplied, causing the gradients of multiple sigmoid/tanh functions to multiply and then explode at the origin and disappear away from the origin.

In order to effectively alleviate this phenomenon, a ReLU activation function similar to biological nerves is proposed, whose expression is:

$$y = max(0, x) \tag{2}$$

ReLU activation function Compared with the traditional Sigmoid function and Tanh function, ReLU is only saturated when x<0, linear and unsaturated when x>0, and its derivative is always equal to 1, the gradient is no longer attenuated. The effective mitigation of vanishing gradient or exploding gradient, avoids unsupervised layer-by-layer pre-training, making it possible to train deep neural networks in a supervised manner.

4.2 Dropout Regularization

The Dropout regularization function helps the network avoid overfitting and improve the generalization of the network. Therefore, this article adds the Dropout layer before the fully connected layer.

Dropout regularization dropout implicit nodes by changing the structure of the neural network itself, so that when updating weights, it is not necessary to update the weights connected to this node, reducing the amount of calculation, and the update of weights no longer depends on the joint action of fixed implicit nodes, avoiding the problem that certain features are only effective under certain features, improving the generalization ability of the network, and also saving certain training time. The principle of Dropout is shown in Figure 4.

Consider a neural network with L hidden layers. Let $l \in \{1, \dots, L\}$index the hidden layers of the network. Let $\boldsymbol{y}^{(l)}$denote the vector of outputs from layer l. $W^{(l)}$and $b^{(l)}$ are the weights and biases at layer l. The feed-forward operation of a standard neural network can be described as (for $l \in \{1, \dots, L-1\}$ and any hidden unit i)

$$y_i^{(l+1)} = f(W_i^{(l+1)} X_i + b_i^{(l+1)}) \tag{3}$$

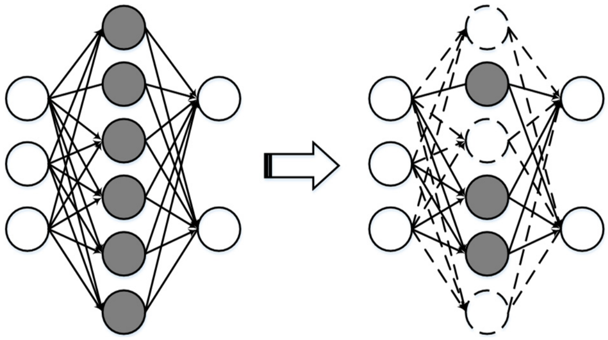

Fig4. Dropout schematic diagram

With dropout, the feed-forward operation becomes

$$\begin{aligned} & r_i^{(l)} \sim \text{Bernoulli}(p) \\ & \tilde{X}^{(l)} = r_i^{(l)} \times x^{(l)} \\ & y_i^{(l+1)} = f(W_i^{(l+1)} \tilde{X}^{(l)} + b_i^{(l+1)}) \end{aligned} \tag{4}$$

Here × denotes an element-wise product. For any layer l, $\boldsymbol{r}^{(l)}$ is a vector of independent Bernoulli random variables each of which has probability p of being 1. This vector is sampled and multiplied element-wise with the outputs of that layer, $\boldsymbol{x}^{(l)}$, to create the thinned outputs $\tilde{\boldsymbol{X}}^{(l)}$.

5. Experiments

5.1 MSTAR Public Data Set

The measured SAR image data used herein is the MSTAR data set. The MSTAR data set is acquired by the X-band SAR sensor. It adopts HH polarization mode and spotlight imaging mode. The radar azimuth and distance resolution are both 0.3m. The omnidirectional angular coverage of the ground target is 0° ~ 360° , but the available public data set is not the SAR image covered by all azimuths. The actual target sample azimuth interval is 1° ~5° . The image size mainly has 128×128 pixels, 158×158 pixels, etc. The MSTAR dataset includes a total of 10 types of ground tactical targets (armored vehicles: BMP-2, BRDM-2, BTR-60, BTR-70; tanks: T-62, T-72; rocket launcher: 2S1; air defense unit: ZSU-234; military truck: ZIL-131; bulldozer: D7). As shown in Figure 3.

The data set uses the image of the ten types target at the 17° depression angles as the training set, and the image at the 15° depression angles is the test set. The specific target model and number of images are shown in Table 1.

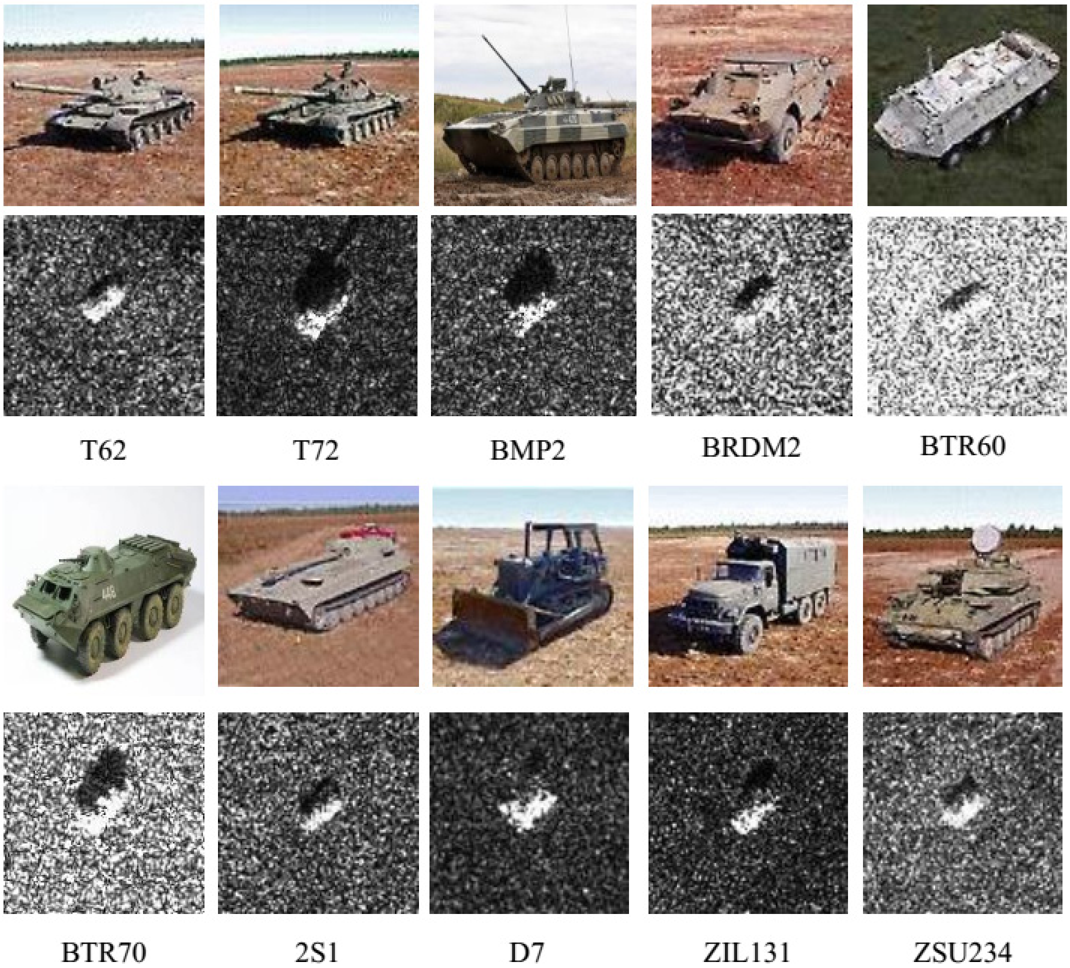

Fig5. Diagram of MSTAR data

Table 1. Target description of MSTAR database

Target model	2S1	BMP2	BRDM 2	BTR60	BTR70	D7	T62	T72	ZIL131	ZSU23 4
Training set17°	299	233	298	256	233	299	299	232	299	299
Test set15°	274	196	274	195	196	274	273	196	274	274

5.2 Experimental Environment Configuration

The experimental parameter settings are shown in Table 2. The parameter selection CNN identifies the common value of the network. The training set epoch is 100, and the test set is tested once for each 5 epoch, where iterating an epoch means traversing the training set once; the average recognition rate is the average of five tests of the network to the test set.

Table 2. Hyperparameters for network training

Experimental parameters	**Numerical value**
Batch size （N）	40
Base learning rate （η）	0.001
Momentum （β）	0.9
Weight decay （λ）	0.0005

The software platform of this experiment: the operating system is Ubuntu14.04 64-bit, using the open source deep learning framework Caffe as the main tool to build the network, using parallel computing based on CUDA8.0 and CUDNN5.1 programming framework to improve the network training speed. The hardware platform used in the experiment is CPU: Intel i7-7700K@4.20GHz, GPU: GTX TITAN X.

5.3 Experimental Results and Analysis

Through the verification and analysis of multiple experiments, after the adjustment of the network structure parameters and the optimization of the network structure, the obtained network structure has better recognition accuracy, timeliness and generalization performance. Table 3 shows the results of the design of the model for the 10 categories of targets.

It can be seen from Table 3 that the network model proposed in this paper has 2 recognition rate of 100% in the MSTAR dataset, and the recognition rate of the BTR60 with the lowest recognition rate is also 97.44%.

Table 3. Classification accuracies of the 10 classes of targets

Class	2S1	BRDM2	BTR60	D7	T62	ZIL131	ZSU234	BMP2	BTR70	T72	PCC
2S1	272	0	0	0	0	3	0	1	0	0	99.27%
BRDM2	0	269	0	0	0	0	2	0	0	3	98.18%
BTR60	0	0	190	0	0	3	1	0	1	0	97.44%
D7	0	2	0	272	0	0	0	0	0	0	99.27%
T62	0	0	0	0	272	1	0	0	0	0	99.63%
ZIL131	0	0	0	1	0	273	0	0	0	0	99.64%
ZSU234	0	0	0	0	0	0	274	0	0	0	100.0%
BMP2	0	0	0	0	0	0	1	192	0	2	98.46%
BTR70	0	0	0	0	0	0	0	0	195	1	99.49%
T72	0	0	0	0	0	0	0	0	0	196	100.0%
Total	99.18%										

In order to further verify the effectiveness of the algorithm, the common methods and deep learning methods are compared with the MSTAR ten-category target recognition results. The comparison results are shown in Table 4.

Table 4. Comparison of MSTAR target recognition results

method	PCC%
SVM [9]	88
CNN [5]	93.76
Gond Gauss [10]	97
Method of this paper	99.18

It can be seen that the network model adopted in this paper has a greater advantage than the traditional recognition method adopted in [10], and verify the effectiveness of CNN in target recognition. Similarly, using the CNN method, the network model proposed in this paper has been further improved compared with the literature [5] method, indicating the effectiveness of the designed network.

6. Conclusion

In order to improve the generalization and accuracy of SAR image target recognition, this paper combines the characteristics of SAR image to design a convolutional neural network model for target recognition. By introducing dense block network structure and using Dropout optimization, the generalization performance of the network is effectively enhanced, and the network model has good performance. The overall recognition accuracy of the 10 types of SAR images in the MSTAR database is 99.18%. Verify the validity of the proposed network. With the continuous development of deep learning and the increase of SAR image data, the method of deep learning combined with the characteristics of SAR images is optimized, which can effectively improve the accuracy of SAR image target recognition.

Reference

[1] LECUN Y, BENGIO Y, HINTON G. Deep learning[J]. Nature, 015,21(7553):436－444.

[2] Zhao J, Guo W, Cui S, et al. Convolutional neural network for SAR image classification at patch level[C]. International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, 2016: 945–948.
[3] Li X, Li C, Wang P, et al. SAR ATR based on dividing CNN into CAE and SNN[C]. 5th Asia-Pacific Conference on Synthetic Aperture Radar (APSAR), Singapore, 2015:676–679.
[4] Ding J, Chen B, Liu H, et al. Convolutional neural network with data augmentation for SAR target recognition[J].IEEE Geoscience and Remote Sensing Letters, 2016, 13(3):364–368.
[5] Z.Z.Tian, R.H.Zhan, J.M.Hu, et al. SAR ATR Based on Convolutional Neural Network[J].Journal of Radars, 2016,5(3):320–325
[6] Chen S, Wang H, Xu F, et al. Target classification using the deep convolutional networks for SAR images[J]. IEEE Transactions on Geoscience and Remote Sensing, 2016, 54(8): 4806-4817.
[7] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting[J]. Journal of Machine Learning Research, 2014, 15(1):1929-1958.
[8] Huang G, Liu Z, Laurens V D M, et al. Densely Connected Convolutional Networks[J]. 2016.
[9] Srinivas U, Monga V, Raj R G. SAR automatic target recognition using discriminative graphical models[J]. IEEE transactions on aerospace and electronic systems, 2014, 50(1): 591-606
[10] O'Sullivan J A, DeVore M D, Kedia V, et al. SAR ATR performance using a conditionally Gaussian model[J]. IEEE Transactions on Aerospace and Electronic Systems, 2001, 37(1): 91-108.