

# Enhanced Radar Imaging Using a Complex-Valued Convolutional Neural Network

Jingkun Gao<sup>ID</sup>, Student Member, IEEE, Bin Deng<sup>ID</sup>, Yuliang Qin, Hongqiang Wang<sup>ID</sup>, and Xiang Li

**Abstract**—Convolutional neural networks (CNN) have successfully been employed to tackle several remote sensing tasks such as image classification and show better performance than previous techniques. For the radar imaging community, a natural question is: Can CNN be introduced to radar imaging and enhance its performance? This letter gives an affirmative answer to this question. We first propose a processing framework by which a complex-valued CNN (CV-CNN) is used to enhance radar imaging. Then we introduce two modifications to the CV-CNN to adapt it to radar imaging tasks. Subsequently, the method to generate training data is shown and some implementation details are presented. Finally, simulations and experiments are carried out, and both results show the superiority of the proposed method on imaging quality and computational efficiency.

**Index Terms**—Complex-valued convolutional neural network (CV-CNN), radar imaging, sidelobe reduction, superresolution.

## I. INTRODUCTION

IMPROVING radar imaging quality using signal processing techniques under given hardware platforms is an appealing topic and has attracted much attention [1], [2]. In the recent decade, compressive sensing [3] has greatly inspired the research on sparsity-driven radar imaging techniques [4], [5]. Typically, the following linear model is adopted:

$$\mathbf{y} = \mathbf{Ax} + \mathbf{n}, \mathbf{y} \in \mathbb{C}^{M \times 1}, \mathbf{n} \in \mathbb{C}^{M \times 1}, \mathbf{x} \in \mathbb{C}^{N \times 1}, \mathbf{A} \in \mathbb{C}^{M \times N} \quad (1)$$

where  $\mathbf{y}$  is the echo signal,  $\mathbf{n}$  is the additive noise,  $\mathbf{x}$  represents the image to be estimated, and  $\mathbf{A}$  is called the sensing matrix or the dictionary. According to (1), conventional imaging can be expressed as  $\hat{\mathbf{x}} = \mathbf{A}^H \mathbf{y}$ , which can be achieved by back projection algorithm (BPA) or by Fast Fourier Transform (FFT). However, either methods suffer from limited resolution, high sidelobes, and strong speckle. Sparsity-driven methods impose prior constraints to  $\mathbf{x}$  and the imaging problem is transformed into an optimization problem

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \psi(\mathbf{x}) \text{ s.t. } \|\mathbf{y} - \mathbf{Ax}\|_2^2 < \sigma^2 \quad (2)$$

where  $\psi(\mathbf{x})$  contains the prior knowledge on  $\mathbf{x}$  and  $\sigma^2$  represents the energy of  $\mathbf{n}$  which is always unknown. The

Manuscript received May 3, 2018; revised May 29, 2018 and July 7, 2018; accepted August 19, 2018. Date of publication September 10, 2018; date of current version December 26, 2018. This work was supported by the National Science Foundation of China under Grant 61871386 and Grant 61571011. (Corresponding author: Bin Deng.)

The authors are with the College of Electronic Science, National University of Defense Technology, Changsha 410073, China (e-mail: oscar92923@163.com; dengbin@nudt.edu.cn).

Color versions of one or more of the figures in this letter are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/LGRS.2018.2866567

most widely used prior is the  $l_1$ -norm constraint in the image domain, i.e.,  $\psi(\mathbf{x}) = \|\mathbf{x}\|_1$ . The constraint is also usually imposed to the transformed domain, i.e.,  $\psi(\mathbf{x}) = \|\mathbf{T}\mathbf{x}\|_1$ , where  $\mathbf{T}$  can be the wavelet transform, and the derivative operator. More complicated  $\psi(\mathbf{x})$ s which takes the inner coherence of  $\mathbf{x}$  into consideration are also proposed [6]. It can be expected that one can obtain better  $\hat{\mathbf{x}}$  by designing more sophisticated  $\psi(\mathbf{x})$  and developing more effective optimization algorithms.

Although sparsity-driven methods can improve imaging quality remarkably, it faces great challenges. First, they are too time consuming to achieve real-time imaging. Different from conventional linear imaging process, the problem in (2) is nonlinear. Usually, a large amount of iterations are needed to converge to a reasonable solution. Second, their stability and robustness cannot easily be guaranteed. The imaging quality promotion obtained by solving (2) is based on accurate modeling. If  $\mathbf{A}$  is inaccurate or the constraint  $\psi(\mathbf{x})$  is unsuitable, the results can be greatly degraded. Although several methods have been proposed for these problems [7], it is at the cost of heavier computational burdens.

In fact, the model in (1) and (2) is a general problem solving framework. In many signal processing fields, e.g., image processing or computer vision [8], this model is also widely used. Convolutional neural network (CNN) is famous mostly for its cutting-edge performance in image classification tasks [9]. Recently, more and more researchers have extended and applied CNN to regression-type problems such as superresolution or denoising [10], [11]. Dong *et al.* [10] showed the close relationship between regression-type CNN and sparsity-based methods. Jin *et al.* [12] pointed out the similarity between the iterative shrinkage methods and the feedforward process of CNN. Moreover, CNN can learn from and be adaptive to the training data, and its structure is highly parallel and free of iterations. As a result, CNN can outperform sparsity-based methods in both accuracy and efficiency.

On top of the above review, a reasonable question is whether CNN can be applied to radar imaging. In our opinion, several issues need to be addressed when applying CNN to radar imaging. For example, what is the overall processing framework? What is the input and output of the CNN? How can one obtain the training data? Is CNN more effective than recent imaging methods? In Sections II and III, we will introduce our solutions to these issues.

## II. METHODS

### A. Overall Framework of CV-CNN-Enhanced Radar Imaging

The proposed processing framework is shown in Fig. 1, the dashed and solid lines show the data flow in training

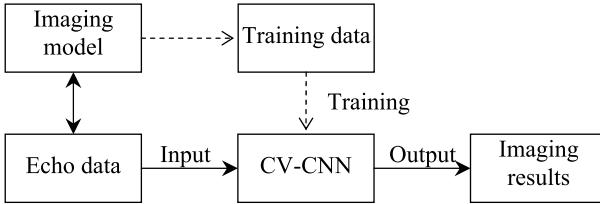


Fig. 1. Framework of CV-CNN-enhanced radar imaging.

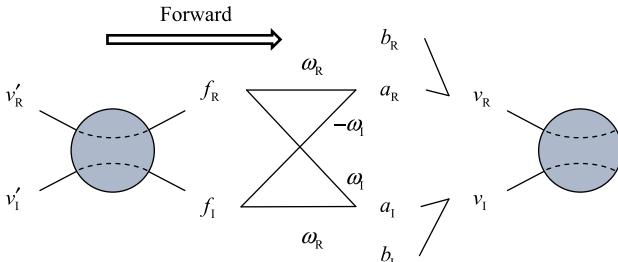


Fig. 2. Neuron and basic connection in CV-CNN.

and imaging/testing period, respectively. The imaging model contains all configurations and parameters that imaging needs. The proposed framework is actually an intuitive application of the idea of supervised learning to radar imaging tasks. If we regard the imaging processor as a black box, the input and output of this block are the echo data and the formed images, respectively. In Fig. 1, we replace the conventional imaging processor with a complex-valued CNN (CV-CNN). For conventional imaging processor, the algorithms are defined according to the developers' knowledge on imaging. Differently here, the imaging processor, i.e., the CV-CNN, is self-developed by training with given input and output examples. It can be seen that both training and imaging rely on the imaging model. Therefore, the premise of CV-CNN-enhanced radar imaging is to confirm the imaging model. Detailed discussions on the modifications on the CV-CNN and the method of training data generation will be given in Sections II-B and II-C.

### B. Modifications on the CV-CNN

There were articles talking about CV-CNN in 1990s [13]. More recently, some publications rediscovered CV-CNN and recognized some of its advantages [14], [15]. Fig. 2 shows the basic neuron structure and connections in CV-CNN.

$v'$  and  $v$  are the input of the neurons at lower and higher levels, respectively,  $f$  is the output of the neuron,  $\omega$  is the weight,  $b$  is the bias, and  $v'$ ,  $f$ ,  $\omega$ ,  $a$ ,  $b$ ,  $v \in \mathbb{C}$ . Subscript “R” and “I” represent the real and imaginary parts, respectively. Suppose the cost function of CV-CNN is  $E \in \mathbb{R}$  which will be defined latter. The training process, i.e., the complex-valued back propagation, can be summarized into following formulas [13], [15]:

$$\begin{cases} \partial E / \partial \omega_R = \partial E / \partial v_R \cdot \partial v_R / \partial \omega_R + \partial E / \partial v_I \cdot \partial v_I / \partial \omega_R \\ \partial E / \partial \omega_I = \partial E / \partial v_R \cdot \partial v_R / \partial \omega_I + \partial E / \partial v_I \cdot \partial v_I / \partial \omega_I \end{cases} \quad (3)$$

$$\begin{cases} \partial E / \partial v'_R = (\partial E / \partial v_R \cdot \omega_R + \partial E / \partial v_I \cdot \omega_I) \cdot \partial f_R / \partial v'_R \\ \partial E / \partial v'_I = (-\partial E / \partial v_R \cdot \omega_I + \partial E / \partial v_I \cdot \omega_R) \cdot \partial f_I / \partial v'_I \end{cases} \quad (4)$$

To make the CV-CNN more efficient and suitable for radar imaging, we introduce two modifications: a new activation function and a new type of neuron for the output layer.

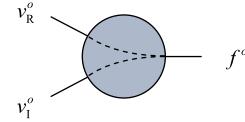


Fig. 3. Structure of the output layer neuron.

In [15], the conventional sigmoid function is extended to CV-CNN. However, many recent publications suggest that Rectified Linear Unit (ReLU) is a more effective activation. Therefore, we are extending ReLU to its complex-valued version in this letter. We define the complex-valued ReLU (cReLU) as follows:

$$f = \text{cReLU}(v') = \max(v'_R, 0) + j \max(v'_I, 0) \quad (5)$$

and we can obtain

$$\begin{aligned} \partial f_R / \partial v'_R &= \begin{cases} 1, & v'_R > 0 \\ 0, & v'_R \leq 0 \end{cases} \\ \partial f_I / \partial v'_I &= \begin{cases} 1, & v'_I > 0 \\ 0, & v'_I \leq 0 \end{cases} \end{aligned} \quad (6)$$

It can be seen that we simply apply the ReLU to the real and imaginary parts of the neuron input independently, which is effective and also critical to simplify the back propagation process. Similarly, a leaky version cReLU can also be defined according to (5).

In most radar imaging scenarios, people mainly care about the amplitude of the images. Also, focusing on the amplitude information of the image helps us to simplify some later issues such as training data generation. Consequently, we hope the outputs of our CV-CNN are real valued. Therefore, a new type of neuron for the output layer is specifically defined, as shown in Fig. 3.

According to Fig. 3, we define the corresponding output layer activation function as

$$f^o = |v_R^o + j v_I^o| = \sqrt{(v_R^o)^2 + (v_I^o)^2} \quad (7)$$

where the superscript “o” indicates the output layer and  $f^o \in \mathbb{R}$ . We can name this function as “Abs.” Then the definition of cost function can be given

$$E = \frac{1}{2} \sum (O - f^o)^2 \quad (8)$$

where  $\sum \cdot$  is applied to all the output layer neurons,  $O \in \mathbb{R}$  is the target function which will be defined in Section II-C. The error term of the output layer is

$$\partial f^o / \partial v_R^o = v_R^o / f^o, \quad \partial f^o / \partial v_I^o = v_I^o / f^o. \quad (9)$$

Given all the above definitions and modifications, we can train the proposed CV-CNN accordingly.

### C. Generation of the Training Data

As aforementioned, the premise of training data generation is the knowledge of the imaging model. In this letter, we will take the classical turntable model shown in Fig. 4 as an example. For other typical imaging models such as SAR/ISAR, the ideas to generate training data are similar.

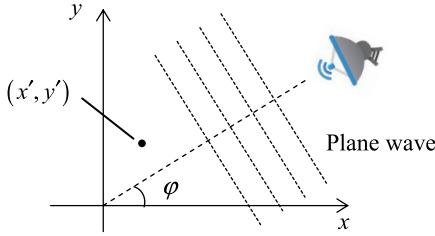


Fig. 4. Turntable imaging model.

According to Fig. 1, the input and output of the CV-CNN are radar echoes and expected images. For real targets, the echoes are determined by the electromagnetic scattering equation

$$\begin{aligned} \mathbf{E}^s(\mathbf{r}) = & \frac{jk \exp(jkR_0)}{4\pi R_0} \hat{k}_s \\ & \times \int_S (\hat{n} \times \mathbf{E}(\mathbf{r}') - \eta_0 \hat{k}_s \times (\hat{n} \times \mathbf{H}(\mathbf{r}')) \\ & \times \exp(-jk(\hat{k}_s - \hat{k}_i) \cdot \mathbf{r}') d\mathbf{r}' \end{aligned} \quad (10)$$

where  $\mathbf{E}^s$ ,  $\mathbf{E}$ , and  $\mathbf{H}$  represent the scattered wave, the total electric, and magnetic fields on target's surface, respectively.  $k$  is the wavenumber,  $R_0$  is the distance from target to the observation point,  $\hat{k}_s$ ,  $\hat{k}_i$  are the unit vectors of the scattering and incidence direction,  $S$  stands for the target's surface,  $\hat{n}$  is the unit normal, and  $\eta_0$  is the free-space wave impedance. The echo is a complicated function of observation angle, polarization, frequency, and there is no explicit expression of the expected image function. To train CV-CNN, input and output must be exactly defined. Consequently, necessary simplification is applied to (10). Concretely, the ideal point scattering model is used and the scattering is confined within 2-D space. Neglecting the attenuation and the propagation phase term, we can obtain the echo under turntable model as

$$E^s(k, \varphi) = \sum_{x', y'} o(x', y') \cdot \exp(-2jk(x' \cos \varphi + y' \sin \varphi)) \quad (11)$$

where  $o(x', y') \in \mathbb{C}$  represents the target's scattering coefficients. Then we can explicitly defined the expected images as

$$O(x, y) = p(x, y) * |o(x, y)| \quad (12)$$

where  $*$  stands for convolution,  $O(x, y) \in \mathbb{R}$  is the same function in (8), and  $p(x, y)$  is the ideal point spread function. We define  $p(x, y)$  as

$$p(x, y) = \exp(-x^2/\sigma_x^2 - y^2/\sigma_y^2) \quad (13)$$

where  $\sigma_x$ ,  $\sigma_y$  control the width of  $p(x, y)$ . According to the  $-3$  dB definition, we can deduce the resolution in these two directions is  $1.18\sigma_x$  and  $1.18\sigma_y$ , respectively.

While generating training data, we first randomly generate hundreds of coordinates  $(x', y')$  within the given area using uniform distribution, and the scattering coefficients are generated using standard complex Gaussian distribution, i.e.,  $o(x', y') \sim N(0, 1) + jN(0, 1)$ . Then training data at the input and output ports of the CV-CNN are obtained by (11) and (12), respectively. We want to clarify that the randomly generated coordinates  $(x', y')$  are not discretized or confined

to integer image cells. They are made arbitrary continuous real numbers so as to imitate the real-world cases upmost.

### III. RESULTS

#### A. Network Structure and Implementation Details

The fully connected layer in our network is achieved by  $\hat{x} = \mathbf{A}^H \mathbf{y}$ . For large-scale problems, the number of elements in  $\mathbf{A}$  can approach the order of  $10^9$ . Therefore, FFT is employed to realize  $\hat{x} = \mathbf{A}^H \mathbf{y}$  fast and implicitly. For general imaging cases, implicit fast implementation of  $\mathbf{A}$  and  $\mathbf{A}^H$  also exist [16]. As a result, the weights of the fully connected layer do not exist explicitly, and they are also excluded in the training phase. For comparison, a similar RV-CNN is also designed in Fig. 5. The real and imaginary parts of the data are treated as two independent channels. The total degree of freedom (Dof) of the RV-CNN is designed to be higher than that of the CV-CNN. We can see that CNN-based imaging is actually a more general imaging approach and conventional algorithms can be treated as its special cases. For example, if we neglect the nonlinear activation function of the fully connected layer and all the following conv. layers and output layer, our network then degrade to the BPA which is a linear transformation.

Our recent implementation is based on MATLAB.<sup>1</sup> Momentum stochastic gradient descent is used and the momentum is 0.9. Weight decay technique is also employed and the coefficient is 0.001. The learning rate is  $3 \times 10^{-5}$  for the first three layers and  $1 \times 10^{-5}$  for the last layer. The batch size is 50 and 50 000 examples are generated for training. Training lasts for 5 epochs. One NVIDIA TITAN Xp card is used, and the training takes approximately 16 h.

For sparsity-driven imaging, we take SPGL1 [17] as a representative to compete with the proposed method. There are three reasons that make SPGL1 a good choice.

- 1) It supports implicit implementation of  $\mathbf{A}$ . For large-scale problems, explicit  $\mathbf{A}$  will lead to unacceptable memory cost and block the computation.
- 2) It is compatible with complex-valued problems. Complex-valued data must be transformed to real ones to comply with real-valued algorithms. This doubles the size of the matrices and vectors, or it sometimes may lead to a wrong solution.
- 3) High executive efficiency. We have tested several algorithms and found that SPGL1 is relatively fast, especially for large-scale problems.

#### B. Numerical Simulations

From Table I, it is easy to calculate that the range and azimuth resolutions are 1.17 and 1.15 cm, respectively. Both  $\sigma_x$ ,  $\sigma_y$  are set to be 0.4 cm which suggests a 0.47 cm resolution, so the expected superresolution ratio is about 2.5. According to [18], imaging can be done by 2-D IFFT for this small rotation angle. In fact, the linear operator  $\mathbf{A}$  and  $\mathbf{A}^H$  in this letter are implicitly implemented by 2-D IFFT and

<sup>1</sup><https://github.com/OscarWenGang/CV-CNN-enhanced-radar-imaging>

TABLE I  
IMAGING PARAMETERS

Probing frequency	213.6 GHz ~ 226.4 GHz
Rotation angle $\varphi$	-1.68°~1.67°
Number of frequency samplings	500
Number of angle samplings	300
Region of imaging	0.7×0.7 m <sup>2</sup>
Number of pixels in the image	236×236

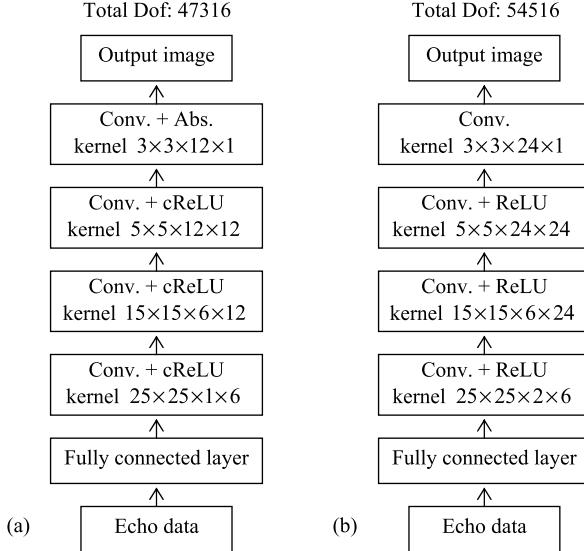


Fig. 5. Network structures of (a) proposed CV-CNN and (b) counterpart RV-CNN.

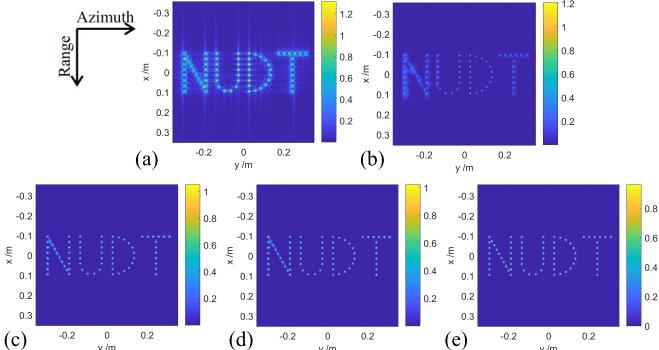


Fig. 6. Imaging results of “NUDT” by (a) FFT, (b) SPGL1, (c) RV-CNN, (d) CV-CNN, and (e) groundtruth image.

2-D FFT. It should be claimed that these fast implementations introduce some errors into  $A$  and  $A^H$  since FFT-based imaging actually approximates the fan-shaped spectral domain support area into a rectangular one. Fig. 6 shows the imaging results of the “NUDT” made up of several unit ideal point scatters. The groundtruth image generated by (12) is shown in Fig. 6(e). Also, we want to clarify that the coordinates of these point scatters are continuous and are not placed at the center of the image cells.

In Fig. 6(a), we can see that the image is of high sidelobes and relatively low resolution. We notice that the points located around the center are focused better than those close to the borders. In [19], this phenomenon is carefully analyzed, and it is caused by the errors introduced into  $A$  we mentioned above. In Fig. 6(b), similar phenomenon can be observed

TABLE II  
COMPARISON ON RMSES AND TIME NEEDS FOR DIFFERENT METHODS

Methods	RMSE, -10dB	RMSE, -5dB	RMSE, 0dB	RMSE, 5dB	RMSE, 10dB	Time needs
FFT	0.1987	0.1778	0.1705	0.1682	0.1675	<b>0.042 s</b>
SPGL1	0.0568	0.0560	0.0568	0.0574	0.0683	27.66 s
RV-CNN	0.0456	0.0308	0.0262	0.0251	0.0248	0.083 s
CV-CNN	<b>0.0434</b>	<b>0.0289</b>	<b>0.0255</b>	<b>0.0247</b>	<b>0.0245</b>	0.071 s

since the errors in  $A$  impact the solving of SPGL1. Compared with Fig. 6(a)–(d) achieve higher quality, and they are also more similar to the groundtruth image in Fig. 6(e). For RV-CNN and CV-CNN, their recent fully connected layers are implicitly implemented by 2-D IFFT. However, the networks can automatically fix these errors in  $A^H$  since the following convolutional layers are adaptive. As a result, RV-CNN and CV-CNN are more robust to modeling errors. Quantitatively, the root-mean-square errors (RMSE) are calculated as performance index. The RMSEs of different methods under different SNRs are listed in Table II. For each SNR level and each method, RMSE is the average of 100 times experiments’ results. The testing data are generated in the same way with the training data. The time needs of different methods are the mean value of all the experiments for each method. For the time consumption of CNN-based imaging, it contains two parts. First, the fully connected layer which is implicitly achieved by 2-D-FFT is implemented on an Intel i3-4150 CPU. Then the data are transferred to the GPU where the following convolutions are calculated.

For all SNR levels, CV-CNN performs the best. For SPGL1, the SNR is imported into the algorithm as a known parameter which is not the case for real applications. Moreover, it can be seen that the RMSEs of SPGL1 is not monotonic. We think this is mainly caused by two reasons: 1) the algorithm is not quite stable since errors exist in  $A$  and 2) the groundtruth produced by (13) does not encourage image’s sparsity too much. For RV-CNN, its performance is weaker than CV-CNN although there are 7200 more parameters. We think this is because the structure shown in Fig. 2 is essentially more efficient for modeling complex-valued problems. For the time needs, CV-CNN is closely after the FFT-based method. As the fully connected layer of CV-CNN is achieved by FTT and its time need can no way better than FFT-based method recently. Compared with SPGL1, the imaging process of CV-CNN is just the feedforward process of the network which can be easily parallelized and greatly accelerated by GPU. Therefore, CV-CNN costs much less time than SPGL1 and can meet the requirements for most real time imaging scenarios. The speedup of the proposed algorithm is not solely due to the utilization of the GPU. More importantly, different from traditional iterative regularization algorithms, our CNN-based algorithm relies on much simpler convolution operations which can be easily parallelized. Therefore, we believe the data listed in Table II are valuable references for practical applications.

### C. Laboratory Results

The experiment scenario is shown in Fig. 7 and an airplane model is used as the target. Other imaging parameters are

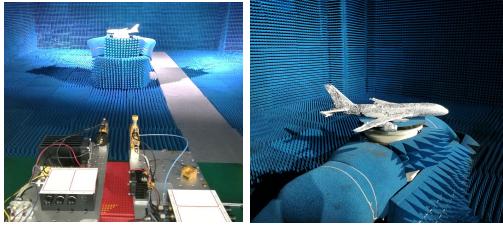


Fig. 7. Experiment scenario.

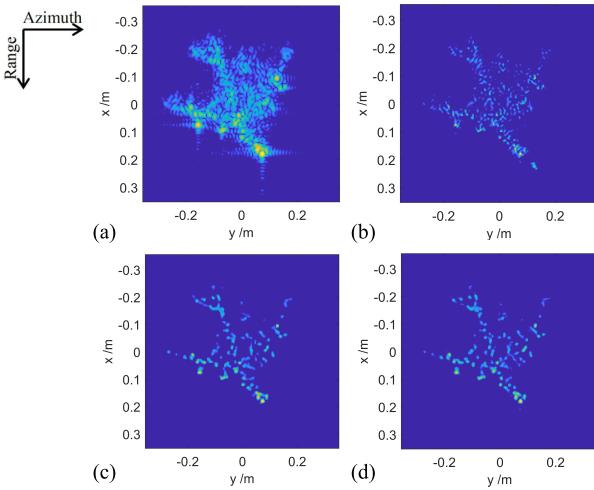


Fig. 8. Laboratory imaging results by different methods. (a) 2-D-FFT. (b) SPGL1. (c) RV-CNN. (d) CV-CNN.

the same with those listed in Table I. In Fig. 8, images are displayed in log magnitude and the dynamic range is 35 dB. The time needs are similar with those listed in Table II since parameters are the same. Unsurprisingly, the resolution of FFT-based image is relatively low, while the sidelobes are extremely high. All other methods can enhance the resolution and suppress the sidelobes, while their visual quality differs from each other apparently. Since it is hard to define groundtruth for real targets, quantitative comparisons can hardly be conducted. Moreover, the quality assessment of radar images is still an open question. Nevertheless, recent results have shown the superiority of the proposed method on robustness and efficiency.

#### D. Discussion

We recently assume that all needed imaging parameters are known in advance. In real-world scenarios, due to the uncertainty of movements, some parameters must be estimated. Whether deep networks can be designed to achieve automatic focusing is an interesting question. Also, under the recent processing framework, the imaging model should be fixed and the network is only valid for one specific imaging model after trained. Whether one network can be used to tackle various radar imaging regimes is also an appealing topic.

#### IV. CONCLUSION

A novel CV-CNN-enhanced radar imaging method is proposed. The processing framework is first presented. The cReLU activation is proposed, and for the neurons in the

output layer, the “Abs” activation is used. Then the method for generating training data is introduced. Different from iterative-based imaging methods, the CNN-based imaging process is just the feedforward process of the network. This makes it much faster and possible to achieve real time enhanced imaging. CNN-based method is also more robust since it is adaptive to the training data. Compared with its real-valued counterpart, CV-CNN achieves better performance with less parameters. We believe radar imaging will benefit more from deep learning techniques such as CNN in the future.

#### REFERENCES

- [1] L. Zhao, L. Wang, L. Yang, A. M. Zoubir, and G. Bi, “The race to improve radar imagery: An overview of recent progress in statistical sparsity-based techniques,” *IEEE Signal Process. Mag.*, vol. 33, no. 6, pp. 85–102, Nov. 2016.
- [2] S. Zhang, Y. Liu, and X. Li, “Autofocusing for sparse aperture ISAR imaging based on joint constraint of sparsity and minimum entropy,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 3, pp. 998–1011, Mar. 2017.
- [3] D. L. Donoho, “Compressed sensing,” *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [4] G. Xu, M. Xing, L. Zhang, Y. Liu, and Y. Li, “Bayesian inverse synthetic aperture radar imaging,” *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 6, pp. 1150–1154, Nov. 2011.
- [5] M. Çetin *et al.*, “Sparsity-driven synthetic aperture radar imaging: Reconstruction, autofocusing, moving targets, and compressed sensing,” *IEEE Signal Process. Mag.*, vol. 31, no. 4, pp. 27–40, Jul. 2014.
- [6] L. Wang, L. Zhao, G. Bi, and C. Wan, “Sparse representation-based ISAR imaging using Markov random fields,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 8, pp. 3941–3953, Aug. 2015.
- [7] X. Zhou, H. Wang, Y. Cheng, and Y. Qin, “Off-grid radar coincidence imaging based on variational sparse Bayesian learning,” *Math. Problems Eng.*, vol. 2016, Mar. 2016, Art. no. 1782178.
- [8] M. Elad, M. A. T. Figueiredo, and Y. Ma, “On the role of sparse and redundant representations in image processing,” *Proc. IEEE*, vol. 98, no. 6, pp. 972–982, Jun. 2010.
- [9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [10] C. Dong, C. C. Loy, K. He, and X. Tang, “Image super-resolution using deep convolutional networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2015.
- [11] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising,” *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.
- [12] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, “Deep convolutional neural network for inverse problems in imaging,” *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4509–4522, Sep. 2017.
- [13] G. M. Georgiou and C. Koutsougeras, “Complex domain backpropagation,” *IEEE Trans. Circuits Syst. II. Analog Digit. Signal Process.*, vol. 39, no. 5, pp. 330–334, May 1992.
- [14] C. Trabelsi *et al.* (2017). “Deep complex networks.” [Online]. Available: <https://arxiv.org/abs/1705.09792>
- [15] Z. Zhang, H. Wang, F. Xu, and Y.-Q. Jin, “Complex-valued convolutional neural network and its application in polarimetric SAR image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7177–7188, Dec. 2017.
- [16] J. Fang, Z. Xu, B. Zhang, W. Hong, and Y. Wu, “Fast compressed sensing SAR imaging based on approximated observation,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 1, pp. 352–363, Jan. 2014.
- [17] E. van den Berg and M. P. Friedlander, “Probing the Pareto frontier for basis pursuit solutions,” *SIAM J. Sci. Comput.*, vol. 31, no. 2, pp. 890–912, 2008.
- [18] J. K. Gao, Y. L. Qin, B. Deng, H. Q. Wang, J. Li, and X. Li, “Terahertz wide-angle imaging and analysis on plane-wave criteria based on inverse synthetic aperture techniques,” *J. Infr. Millim. Terahertz Waves*, vol. 37, no. 4, pp. 373–393, 2016.
- [19] J. Gao *et al.*, “Fast three-dimensional image reconstruction of a stand-off screening system in the terahertz regime,” *IEEE Trans. THz Sci. Technol.*, vol. 8, no. 1, pp. 38–51, Jan. 2018.