Basis Pursuit Denoising via Recurrent Neural Network Applied to Super-Resolving SAR Tomography

Kun Qian, Yuanyuan Wang, *Member, IEEE*, Peter Jung, *Member, IEEE*, Yilei Shi, *Member, IEEE*, and Xiao Xiang Zhu, *Fellow, IEEE*

***Abstract*—Finding sparse solutions of underdetermined linear systems commonly requires the solving of L_1 regularized least-squares minimization problem, which is also known as the basis pursuit denoising (BPDN). They are computationally expensive since they cannot be solved analytically. An emerging technique known as *deep unrolling* provided a good combination of the descriptive ability of neural networks, explainable, and computational efficiency for BPDN. Many unrolled neural networks for BPDN, e.g., learned iterative shrinkage thresholding algorithm and its variants, employ shrinkage functions to prune elements with small magnitude. Through experiments on synthetic aperture radar tomography (TomoSAR), we discover the shrinkage step leads to unavoidable information loss in the dynamics of networks and degrades the performance of the model. We propose a recurrent neural network (RNN) with novel sparse minimal gated units (SMGUs) to solve the information loss issue. The proposed RNN architecture with SMGUs benefits from incorporating historical information into optimization and, thus, effectively preserves full information in the final output. Taking TomoSAR inversion as an example, extensive simulations demonstrated that the proposed RNN outperforms the state-of-the-art deep learning-based algorithm in terms of super-resolution power and generalization ability. It achieved 10%–20% higher double-scatterer detection rate and is less sensitive to phase and amplitude ratio difference between scatterers. Test on real TerraSAR-X spotlight images also shows the high-quality 3-D reconstruction of the test site.**

***Index Terms*—Basis pursuit denoising (BPDN), recurrent neural network (RNN), sparse reconstruction, synthetic aperture radar tomography (TomoSAR).**

I. Introduction

A. Motivation

SPARSE solutions are ordinarily desired in a multitude of fields, such as radar imaging, medical imaging, and acoustics signal processing. Compressive sensing (CS) theory tells that the exact solution in the absence of noise is the signal with the minimum L_0-norm while still fulfilling the forward model. As the L_0-norm minimization is NP-hard, this is often solved by L_1-norm minimization. The unconstrained form of a linear system can be formulated as follows:

$$\min_{x} ||\mathbf{A}\mathbf{x} - \mathbf{b}||_2^2 + \lambda||x||_1 \tag{1}$$

where $\mathbf{A}$, $\mathbf{x}$, and $\mathbf{b}$ are the sensing matrix, the signal to be retrieved, and the measurements. Solving (1) is an unconstrained convex optimization problem, whose objective function is nondifferentiable. It is also known as basis pursuit denoising (BPDN) [1]. In the field of remote sensing, sparse signals are widely expected. Therefore, BPDN is broadly employed to exploit sparsity prior in various remote sensing applications, including, but not limited to, pan-sharpening [2], spectral unmixing [3], microwave imaging [4], and synthetic aperture radar tomography (TomoSAR) [5]. In this work, we focus on addressing BPDN in TomoSAR inversion, but our findings are applicable to general sparse reconstruction problems in other fields as well.

Generic solvers for BPDN are either first- or second-order CS-based methods [6], [7], [8]. First-order methods are typically based on a linear approximation of gradient, e.g., the iterative shrinkage thresholding algorithm (ISTA) [9], coordinate descent (CD) [10], and alternating direction method of multipliers (ADMM) [11]. Second-order methods usually have much better performance than first-order methods. An example of the second-order method is the prime dual inferior point method (PDIPM) [12]. It was demonstrated in [5] and [13] that CS-based methods are able to achieve unprecedented super-resolution ability and location accuracy compared to conventional linear algorithm [14], [15]. In spite of the good performance of CS-based methods, they often suffer from heavy computational burdens due to their iterative properties and are hard to extend to practical use.

In the past years, the advent of deep neural networks has attracted the interest of many researchers and triggered extensive studies due to their excellent learning and expression power. Deep neural networks have demonstrated their availability and advanced the state-of-the-art for many problems. More recently, an emerging deep learning algorithm coined ***deep unfolding*** [16] was proposed to provide a concrete and systematic connection between iterative physical model-based algorithms and deep neural networks. Inspired by this concept,

Manuscript received 31 May 2022; revised 9 September 2022; accepted 22 October 2022. Date of publication 2 December 2022; date of current version 20 December 2022. This work was supported in part by the European Research Council under Grant Agreement ERC-2016-StG-714087 (So2Sat), in part by the Helmholtz Association through the Framework of the Helmholtz Excellent Professorship “Data Science in Earth Observation—Big Data Fusion for Urban Research” under Grant W2-W3-100, and in part by the German Federal Ministry of Education and Research in the framework of the International Future AI Lab “AI4EO—Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond” under Grant 01DD20001. *(Corresponding author: Xiao Xiang Zhu.)*

Kun Qian, Yuanyuan Wang, Peter Jung, and Xiao Xiang Zhu are with the Chair of Data Science in Earth Observation, Technical University of Munich, 80333 Munich, Germany (e-mail: kun.qian@tum.de; y.wang@tum.de; peter.jung@tum.de; xiaoxiang.zhu@tum.de).

Yilei Shi is with the Chair of Remote Sensing Technology, Technical University of Munich, 80333 Munich, Germany (e-mail: yilei.shi@tum.de).

Digital Object Identifier 10.1109/TGRS.2022.3221185

various neural networks were proposed to solve BPDN in CS problems by unrolling iterative CS solvers. The first work of deep unfolding dates back to learned ISTA (LISTA) [17], which was designed for solving sparse recovery. LISTA unrolls ISTA, one of the most popular algorithms, and substantially improves computational efficiency and parameter tuning. Yang et al. [18] proposed ADMM-CSnet by unrolling the ADMM algorithm to deep hierarchical network architecture and applied ADMM-CSnet to magnetic resonance imaging (MRI) and natural image CS. Results in [18] indicate the favorable performance of ADMM-CSnet in high computational speed. For remote sensing applications, CSR-net [19] was proposed by combining deep unfolding structures and convolutional neural network modules, and achieved fast and accurate 3-D microwave imaging. In addition, Wei et al. [20] proposed AF-AMPNet by unrolling approximate message passing with phase error estimation (AF-AMP) to a deep neural network. AF-AMPNet was employed in sparse aperture (SA) inverse SAR (ISAR) imaging and accelerated the imaging process. Inspired by the encouraging achievements made by deep unfolding, the TomoSAR community started to design deep neural networks by unrolling iterative optimization solvers for solving BPDN in TomoSAR inversion. Gao et al. [21] unrolled and mapped vector AMP (VAMP) [22] into a neural network for line spectral estimation and applied it to tackle TomoSAR inversion. Results in [21] show that L-VAMP is able to separate overlaid scatterers. $\boldsymbol{\gamma}$-Net was proposed in [23] by tailoring the complex-valued (CV) LISTA network. $\boldsymbol{\gamma}$-Net introduced weight coupling structure [24] and support selection scheme [24] to each iteration block in LISTA and improved the conventional soft-thresholding function by the piecewise linear function. It was demonstrated in [23] that $\boldsymbol{\gamma}$-Net improves the computational efficiency by two to three orders of magnitude compared to the state-of-the-art second-order TomoSAR solver SL1MMER [13] while showing no degradation in super-resolution ability and location accuracy.

However, unrolled neural networks do not consider historical information in the updating rules. To be exact, the output is generated exclusively based on the output of its previous layer. This kind of learning architecture leads to an error propagation phenomenon, where the error in the first few layers will be propagated and even amplified in the upcoming layers. Moreover, when the unrolled neural networks are designed for sparse reconstruction, shrinkage steps are usually required to promote sparsity. The shrinkage step utilizes thresholding functions to prune elements with a small magnitude to zero, and such pruning causes information loss in the dynamics of the neural network. Once useful information is discarded in the previous layers, the upcoming layers have no longer a chance to utilize the discarded information, thus degrading the performance of the neural network and sometimes leading to a large error in the final output.

B. Contribution of This Article

In this article, we aim to address the problem of information loss caused by shrinkage steps in unrolled neural networks designed for sparse reconstruction. To this end, we propose a novel architecture, termed the sparse minimal gated unit (SMGU), to incorporate historical information into optimization so that we can promote sparsity using thresholding functions and preserve full information simultaneously. In addition, we extend SMGU to the CV domain as CV-SMGU and use it to build a gated recurrent neural network (RNN) for solving TomoSAR inversion. The main contribution of this article is listed in the following.

1) We addressed the problem of information loss in unrolled neural networks for sparse reconstruction by a novel gated RNN. The gated RNN is built using SMGUs, which incorporate historical information into optimization. The proposed gated RNN is able to promote sparsity by employing shrinkage thresholding functions. Simultaneously, the pruned information will be reserved in the cell state of SMGUs; thus, full information can be preserved in the dynamics of the network.
2) We extend the SMGU to the CV domain, called CV-SMGU, and apply the gated RNN built with CV-SMGUs to solve TomoSAR inversion. To the best of our knowledge, it is the first attempt to bridge the gated RNN and TomoSAR inversion. We may provide novel insights and open a new prospect for future deep learning-based TomoSAR inversion.
3) We carry out a systematic evaluation to demonstrate that the proposed gated RNN outperforms the state-of-the-art deep learning-based TomoSAR algorithm $\boldsymbol{\gamma}$-Net in terms of super-resolution power and generalization ability for TomoSAR inversion.

The remainder of this article is outlined as follows. The TomoSAR imaging model and $\boldsymbol{\gamma}$-Net is briefly reviewed in Section II. Section III provides an overview of the formulation of SMGUs and CV-SMGUs with application to TomoSAR inversion. Results of systematic evaluation, using simulated and real data, are presented in Section IV. Section V discussed the generalization ability w.r.t. baseline discrepancy and analyzed the model convergence. Finally, the conclusion of this article is drawn in Section VI.

II. Background

A. TomoSAR Imaging Model

In this section, we briefly introduce the TomoSAR imaging model. Fig. 1 demonstrates the SAR imaging model at a fixed azimuth position. A stack of CV SAR acquisitions over the illuminated area is obtained at slightly different orbit positions (the elevation aperture). The CV measurement g_n of the nth acquisition is the integral of the reflectivity profiles $\gamma(s)$ along the elevation direction s. The discrete TomoSAR imaging model can be written as

$$\mathbf{g} = \mathbf{R}\boldsymbol{\gamma} + \boldsymbol{\varepsilon} \tag{2}$$

where $\mathbf{g} \in \mathbb{C}^{N\times 1}$ is the CV SAR measurement vector and $\boldsymbol{\gamma} \in \mathbb{C}^{L\times 1}$ denotes the discrete reflectivity profile uniformly sampled at elevation position $s_l (l = 1, 2, \ldots, L)$ along the elevation direction. N is the number of measurements, and L is the number of discrete elevation indices. $\mathbf{R} \in \mathbb{C}^{N\times L}$ is the irregularly sampled discrete Fourier transformation mapping matrix with $R_{nl} = \exp(-j2\pi\xi_n s_l)$, where ξ_n is the frequency

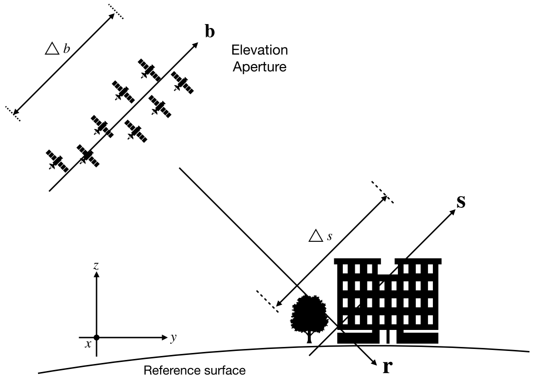


Fig. 1. SAR imaging geometry at a fixed azimuth position. The elevation synthetic aperture is built up by acquisition from slightly different incidence angles. The flight direction is orthogonal to the plane.

proportional to the perpendicular baseline of the nth acquisition. The readers can refer to [14] for more details of the SAR imaging model.

Since the reflectivity profile γ is sufficiently sparse in urban areas [5], retrieving γ is a sparse reconstruction problem. Accordingly, γ in the presence of measurement noise $\boldsymbol{\varepsilon}$ can be estimated by BPDN optimization, which is formulated as follows:

$$\hat{\boldsymbol{\gamma}} = \arg\min_{\boldsymbol{\gamma}} \left\{ \|\mathbf{g} - \mathbf{R}\boldsymbol{\gamma}\|_2^2 + \lambda \|\boldsymbol{\gamma}\|_1 \right\} \tag{3}$$

where λ is a regularization parameter balancing the sparsity and data-fitting terms. It should be adjusted according to the noise level and the desired sparsity level. The choice of a proper λ is described in great detail in [1].

B. Review of $\boldsymbol{\gamma}$-Net

As shortly mentioned previously, conventional CS-based BPDN solvers for (3) are extremely computationally expensive. To overcome the heavy computational burden and make super-resolving TomoSAR inversion for large-scale processing feasible, the authors proposed $\boldsymbol{\gamma}$-Net in [23], which tailors the first unrolling ISTA network, to mimic a CS-based BPDN solver. To be specific, $\boldsymbol{\gamma}$-Net introduces the weight coupling structure and support selection scheme, and improves the conventional soft-thresholding function by the piecewise linear function. Fig. 2 illustrates us the architecture of the ith layer in $\boldsymbol{\gamma}$-Net. ***SS*** in $\boldsymbol{\gamma}$-Net indicates a special thresholding scheme called support selection, which will select ρ^i percentage of entries with the largest magnitude and trust them as "true support." The "true support" will be directly fed to the next layer, bypassing the shrinkage step. η_{pwl} is a novel thresholding function, called piecewise linear function, to execute shrinkage in the $\boldsymbol{\gamma}$-Net. It contributes to improving the convergence rate and reducing reconstruction error. More details about $\boldsymbol{\gamma}$-Net formulation and the full model structure can be found in the Appendix.

However, as one can see in Fig. 2, $\boldsymbol{\gamma}$-Net inherits the learning architecture of LISTA despite modifications made by the authors to improve the performance. Therefore, it can be imagined that $\boldsymbol{\gamma}$-Net will suffer from the same problem as LISTA. Specifically speaking, in the learning architecture of $\boldsymbol{\gamma}$-Net, the output at the current layer is generated exclusively from the previous output. As a natural consequence, the final output can only utilize the information from the second last layer. When useful or important information is pruned by shrinkage steps in the intermediate layers, the discarded information is no longer possible to contribute to the final output. Consequently, a large reconstruction error in the final output can be expected. Fig. 3 demonstrates an unsuccessful detection of double scatterers in our experiments. In this experiment, the double scatterers were assumed to have identical phase and amplitude and were spaced by 0.6 Rayleigh resolution, i.e., in a super-resolution regime, and the SNR level was set as 6 dB. In general, if we cannot resolve the overlaid double scatterers, the reflectivity profile should have a dominant amplitude peak between the true elevation position of the double scatterers, as it is shown by the estimate of a nonsuper-resolving algorithm SVD-Wiener [14] in Fig. 3. However, $\boldsymbol{\gamma}$-Net was able to detect one of the double scatterers with very high localization accuracy but failed to find the other one. From our perspective, it was abnormal, and we supposed that this unsuccessful double-scatterer separation should attribute to the information loss caused by shrinkage steps in $\boldsymbol{\gamma}$-Net. Inspecting the intermediate layers in $\boldsymbol{\gamma}$-Net, we discovered that the information of the second scatterer gradually diminished after each shrinkage step in the intermediate layers. Until the second last layer, the information of the second scatterer fell out completely. As a result, the final output of $\boldsymbol{\gamma}$-Net, i.e., the estimate of γ, did not contain the information of the second scatterer. Hence, we cannot detect the second scatterer.

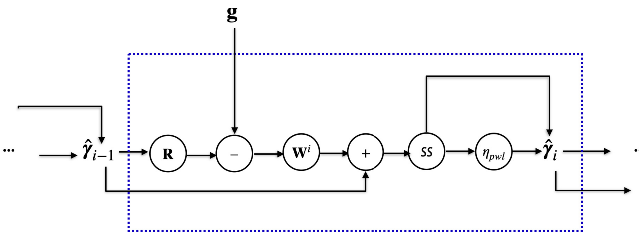


Fig. 2. Illustration of the ith layer in $\boldsymbol{\gamma}$-Net.

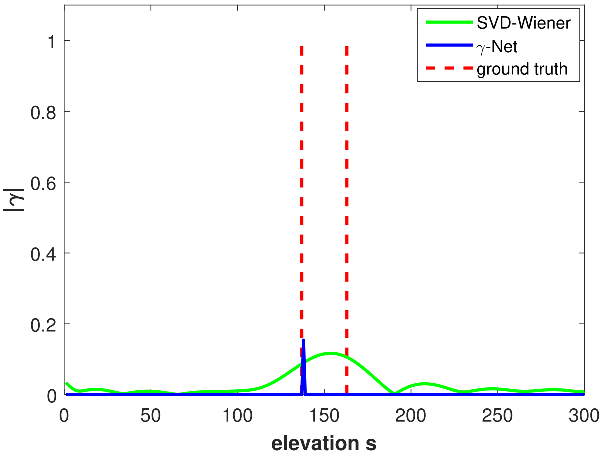


Fig. 3. Example of unsuccessful detection of double scatterers caused by information loss. $\boldsymbol{\gamma}$-Net detects one of the double scatterers with very high localization accuracy but fails to find the other one.

III. Methodology

A. Adaptive ISTA and sc2net

In the optimization community, it has been extensively studied and proved [25], [26], [27] that incorporating historical

TABLE I

FORMAL DEFINITION OF THE tTH LAYER IN DIFFERENT MODELS AND COMPARISON OF THEIR DIFFERENCE. γ-NET HAS NO GATED EXPRESSION. SLSTM UNIT INTRODUCES FORGET AND INPUT GATES TO INCORPORATE HISTORICAL INFORMATION. SMGU HAS THE MINIMAL NUMBER OF GATES WHILE MAINTAINING THE PERFORMANCE COMPARED TO THE SLSTM UNIT. CV-SMGU EXTENDS SMGU TO THE CV DOMAIN. THE FORGET GATE IS ACTIVATED ON THE MAGNITUDE USING ***tanh*** FUNCTION INSTEAD OF THE SIGMOID FUNCTION TO GUARANTEE THE ACTIVATION VALUE RANGING FROM 0 TO 1

	γ-Net layer	SLSTM unit	SMGU	CV-SMGU
Input gate	-	$\mathbf{i}^{(t)} = \sigma\{\mathbf{W}_{i2}^{(t)}\hat{\gamma}^{(t-1)} + \mathbf{W}_{i1}^{(t)}\mathbf{g}\}$	-	-
Forget gate	-	$\mathbf{f}^{(t)} = \sigma\{\mathbf{W}_{f2}^{(t)}\hat{\gamma}^{(t-1)} + \mathbf{W}_{f1}^{(t)}\mathbf{g}\}$	$\mathbf{f}^{(t)} = \sigma\{\mathbf{W}_{f2}^{(t)}\hat{\gamma}^{(t-1)} + \mathbf{W}_{f1}^{(t)}\mathbf{g}\}$	$\mathbf{f}^{(t)} = \tanh\{\lvert\tilde{\mathbf{W}}_{f2}^{(t)}\tilde{\gamma}^{(t-1)} + \tilde{\mathbf{W}}_{f1}^{(t)}\tilde{\mathbf{g}}\rvert\}$
Cell state	-	$\bar{\mathbf{c}}^{(t)} = \mathbf{W}_2\hat{\gamma}^{(t-1)} + \mathbf{W}_1\mathbf{g}$ $\mathbf{c}^{(t)} = \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + \mathbf{i}^t \odot \bar{\mathbf{c}}^t$	$\bar{\mathbf{c}}^{(t)} = \mathbf{W}_2(\mathbf{f}^{(t)} \odot \hat{\gamma}^{(t-1)}) + \mathbf{W}_1\mathbf{g}$ $\mathbf{c}^{(t)} = (1 - \mathbf{f}^{(t)}) \odot \hat{\gamma}^{(t-1)} + \mathbf{f}^{(t)} \odot \bar{\mathbf{c}}^{(t)}$	$\bar{\mathbf{c}}^{(t)} = \tilde{\mathbf{W}}_2(\mathbf{f}^{(t)} \odot \tilde{\gamma}^{(t-1)}) + \tilde{\mathbf{W}}_1\tilde{\mathbf{g}}$ $\mathbf{c}^{(t)} = (1 - \mathbf{f}^{(t)}) \odot \tilde{\gamma}^{(t-1)} + \mathbf{f}^{(t)} \odot \bar{\mathbf{c}}^{(t)}$
Output	$\tilde{\gamma}^{(t)} = \eta_{ss}{}_{\theta^t}^{\rho^t}\{\tilde{\gamma}^{(t-1)} + \tilde{\mathbf{W}}^t(\tilde{\mathbf{g}} - \tilde{\mathbf{R}}\tilde{\gamma}^{(t-1)}), \boldsymbol{\theta_t}\}$	$\hat{\gamma}^{(t)} = \eta_{dt}\left(\mathbf{c}^{(t)}\right)$	$\hat{\gamma}^{(t)} = \eta_{dt}\left(\mathbf{c}^{(t)}\right)$	$\tilde{\gamma}^{(t)} = \eta_{cv-dt}\{\mathbf{c}^{(t)}\}$

information contributes to improving the algorithm performance. Inspired by the high-level ideas from the previous research, researchers proposed adaptive ISTA in [28] to integrate and make use of historical information by introducing two adaptive momentum vectors $\mathbf{f}$ and $\mathbf{i}$ into ISTA in each iteration, which is formulated as follows:

$$\begin{aligned}\bar{\mathbf{c}}^{(t)} &= \mathbf{W}_2\hat{\boldsymbol{\gamma}}^{(t-1)} + \mathbf{W}_1\mathbf{g}\\ \mathbf{c}^{(t)} &= \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \odot \bar{\mathbf{c}}^{(t)}\\ \hat{\boldsymbol{\gamma}}^{(t)} &= \eta_{\text{st}}\left(\mathbf{c}^{(t)}\right)\end{aligned} \tag{4}$$

where η_{st} indicates the conventional soft-thresholding function and its CV version reads

$$\eta_{\text{st}}\left(\hat{\gamma}_i, \theta_i\right) = \begin{cases} \dfrac{\hat{\gamma}_i}{|\hat{\gamma}_i|}\max\left(|\hat{\gamma}_i| - \theta_i, 0\right), & |\hat{\gamma}_i| \neq 0\\ 0, & \text{else.}\end{cases} \tag{5}$$

Compared to ISTA, whose update rule can be equivalently expressed as $\hat{\boldsymbol{\gamma}}^{(t)} = \eta_{\text{st}}(\bar{\mathbf{c}}^{(t)})$ using the same notation, the adaptive ISTA takes not only the current information but also the previous information into consideration. To be exact, at the tth iteration of the adaptive ISTA, the estimate is generated by linear combining the historical information $\mathbf{c}^{(t-1)}$ at the previous iteration and the current information $\bar{\mathbf{c}}^{(t)}$ at the current iteration. The historical information $\mathbf{c}^{(t-1)}$ and the current information $\bar{\mathbf{c}}^{(t)}$ are weighted by the adaptive momentum vectors $\mathbf{f}^{(t)}$ and $\mathbf{i}^{(t)}$, respectively. By this means, the final estimate of the adaptive ISTA will accumulate historical information weighted by different $\mathbf{f}^{(t)}$ and $\mathbf{i}^{(t)}$ for different iterations.

However, one problem of the adaptive ISTA is that the two momentum vectors in each adaptive ISTA iteration are difficult to determine. So far, there has been no analytical way to determine the values of the adaptive momentum vectors $\mathbf{f}^{(t)}$ and $\mathbf{i}^{(t)}$. Usually, they are selected by tediously handcraft tuning, which takes a lot of time and cannot guarantee optimal performance. To address this issue, the authors proposed ***sc2net*** in [28] by recasting the adaptive ISTA as an RNN to parameterize the two momentum vectors and learn them from data. The ***sc2net*** is built by sparse long short-term memory (SLSTM) [28] units, as it is demonstrated in Fig. 4. Each SLSTM unit represents an individual layer of ***sc2net***. At the tth layer of ***sc2net***, the input gate and the forget gate correspond to the momentum vectors $\mathbf{i}^{(t)}$ and $\mathbf{f}^{(t)}$ in each adaptive ISTA iteration, respectively. Hence, we use the same notation in SLSTM units to describe the input and forget gates. The two gates in each SLSTM unit are parameterized with the input data $\mathbf{g}$ and the output $\hat{\boldsymbol{\gamma}}^{(t-1)}$ at the previous layer as follows:

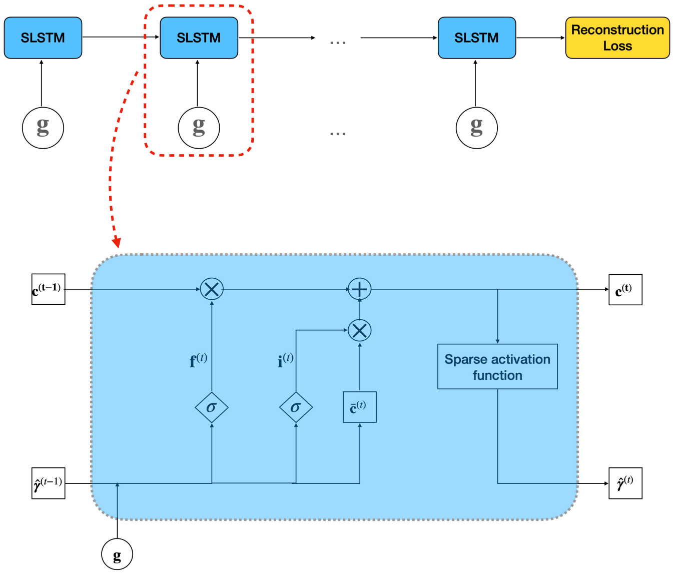


Fig. 4. Sc2net and detailed learning architecture of SLSTM unit. Each SLSTM unit builds an individual layer of sc2net.

$$\begin{aligned}\mathbf{i}^{(t)} &= \sigma\left(\mathbf{W}_{i2}^{(t)}\hat{\boldsymbol{\gamma}}^{(t-1)} + \mathbf{W}_{i1}^{(t)}\mathbf{g}\right)\\ \mathbf{f}^{(t)} &= \sigma\left(\mathbf{W}_{f2}^{(t)}\hat{\boldsymbol{\gamma}}^{(t-1)} + \mathbf{W}_{f1}^{(t)}\mathbf{g}\right).\end{aligned} \tag{6}$$

To clarify, the SLSTM unit does not have an output gate like conventional LSTM units. By substituting (6) into (4), we have the formal definition of the SLSTM unit, as it is listed in Table I. $\mathbf{W}_{i1}, \mathbf{W}_{i2}, \mathbf{W}_{f1}$, and $\mathbf{W}_{f2}$ denote four trainable weight matrices to determine the input and forget gates in each SLSTM unit. It is worth mentioning that the weight matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ are also learned from data, while they are shared for all SLSTM units in an individual ***sc2net***. $\sigma(\cdot)$ indicates the conventional sigmoid function, which is expressed as

$$\sigma(x) = \frac{1}{1 + e^{-x}}. \tag{7}$$

The sparse activation function employed in the SLSTM to promote sparse codes is the double hyperbolic tangent function, which is abbreviated as $\eta_{dt}(\cdot)$ and defined as follows:

$$\eta_{dt}(\hat{\boldsymbol{\gamma}}, s, \theta) = s \cdot \left[\tanh(\hat{\boldsymbol{\gamma}} + \theta) + \tanh(\hat{\boldsymbol{\gamma}} - \theta)\right] \tag{8}$$

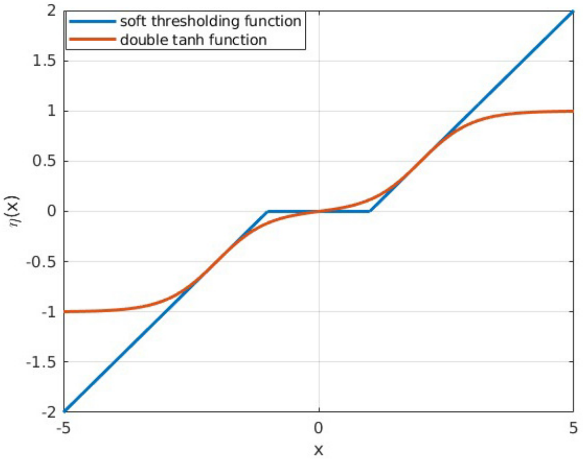


Fig. 5. Comparison of the double hyperbolic tangent function $\eta_{dt}(\cdot)$ and the soft-thresholding function. $\eta_{dt}(\cdot)$ effectively imitates the soft-thresholding function within the interval of $[-\theta, \theta]$.

where s and θ denote two trainable parameters. It is worth noting that the double hyperbolic tangent function can be viewed as a smooth and continuously differentiable alternative to the conventional soft-thresholding function. Its advantages are mainly twofold. On the one hand, its second derivative sustains for a long span, thus contributing to addressing the gradient vanishing problem caused by the cell recurrent connection [29]. On the other hand, it is able to effectively imitate the soft-thresholding function within the interval of $[-\theta, \theta]$. Fig. 5 demonstrates an example of the double hyperbolic tangent function and compares it to the soft-thresholding function.

To sum up, ***sc2net*** inherits the advantage of the adaptive ISTA, which incorporates historical information into optimization. The cell state $\mathbf{c}^{(t)}$ in each SLSTM unit of ***sc2net*** acts as an "eye" to supervise the optimization from two aspects. First, the long-term dependence on the previous outputs can be captured and maintained. Second, important information will be automatically accumulated, whereas useless or redundant information will be forgotten, in the dynamics of ***sc2net***.

However, when we tried to apply ***sc2net*** in TomoSAR inversion, we discovered that a drawback of ***sc2net*** impedes its application. As it is known, a complicated RNN model, on the one hand, hinders theoretical analysis and empirical understanding. On the other hand, it also implies that we have to learn more parameters and tune more components. As a natural result, more training sequences, which mean more training time, and (perhaps) larger training datasets are required. When ***sc2net*** is applied to solve TomoSAR inversion, we need to learn four weight matrices $\mathbf{W}_{f1}^{(t)}$, $\mathbf{W}_{f2}^{(t)}$, $\mathbf{W}_{i1}^{(t)}$, and $\mathbf{W}_{i2}^{(t)}$, which have the dimension $L \times L$, $L \times N$, $L \times L$, and $L \times N$, respectively, to determine the forget gate $\mathbf{f}^{(t)}$ and the input gate $\mathbf{i}^{(t)}$ in each individual SLSTM unit. Moreover, SAR data are CV. Hence, there weight matrices to be learned should be CV as well, thus duplicating the number of trainable components and parameters since two weight matrices need to be learned simultaneously as the real and imaginary parts of a CV weight matrix. Through our research and experiments, we found that such a large amount of high-dimensional weight matrices to be learned makes the training procedure time-consuming. More seriously, it is difficult for the model to converge in the training process.

B. Complex-Valued Sparse Minimal Gated Unit

To address the aforementioned issue and better leverage the power of incorporating historical information for solving TomoSAR inversion, it is necessary to reduce the components and simplify the model architecture. Recently, studies and evaluations in [30], [31], and [32] demonstrated that the gated unit contributes to significantly improving the performance of an RNN compared to that without any gated unit. However, it does not signify that the more the gates, the better the performance of an RNN. Based on this fact, the author proposed an RNN model with only one gate, termed the minimal gated unit (MGU), and revealed that fewer gated units reduce the complexity but not necessarily the performance.

Inspired by the valuable works in [33] and [34], we proposed SMGU, as illustrated in Fig. 6, by coupling the input gate to the forget gate, thus further the simplifying SLSTM unit. The detailed equations for defining the SMGU are listed in Table I.

In the tth layer of an RNN with SMGUs, we will first compute the forget gate $\mathbf{f}^{(t)}$. In addition, the short-term response $\bar{\mathbf{c}}^{(t)}$ is generated by combining the input data $\mathbf{g}$ and the "forgotten" portion $(\mathbf{f}^{(t)} \odot \hat{\boldsymbol{\gamma}}^{(t-1)})$ of the output from the previous layer. Hereafter, the new hidden state $\mathbf{c}^t$ of the current layer can be formulated by combining part of $\hat{\boldsymbol{\gamma}}^{(t-1)}$ and the short-term response $\bar{\mathbf{c}}^{(t)}$, which are determined by $(1-\mathbf{f}^{(t)})$ and $\mathbf{f}^{(t)}$, respectively. Eventually, the sparse activation function, i.e., the double hyperbolic function, will be applied to the current hidden state $\mathbf{c}^t$ for shrinkage and thresholding to promote sparsity of the output.

In this formulation, we can see that the SMGU is able to simultaneously execute a twofold task with only one forget gate. On the one hand, SMGU allows a compact representation by enabling the hidden state $\mathbf{c}^{(t)}$ to discard irrelevant or redundant information. On the other hand, SMGU is capable of controlling how much information from the previous layer takes over. In addition, comparing the formulation of SMGU to SLSTM in Table I, we can see that the parameter size of SMGU is only about half of that of SLSTM since the weight matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ are shared for different layers in a network. The main advantage brought by the significant elimination of trainable parameters is that we can reduce the requirement for training data, training time, and architecture tuning.

In addition to the improvements using SMGU, an extension of SMGU to the complex domain is required. CV-SMGU has essentially the same structure as SMGU despite the two differences. First, each neuron in CV-SMGU has two channels indicating the real and imaginary parts of a complex number, respectively. Often, the real and imaginary parts are not directly activated. Instead, the activation is performed on the magnitude of the complex number. Hence, it is no longer appropriate to use the sigmoid function for activation to generate the forget gate since the magnitude is always greater than

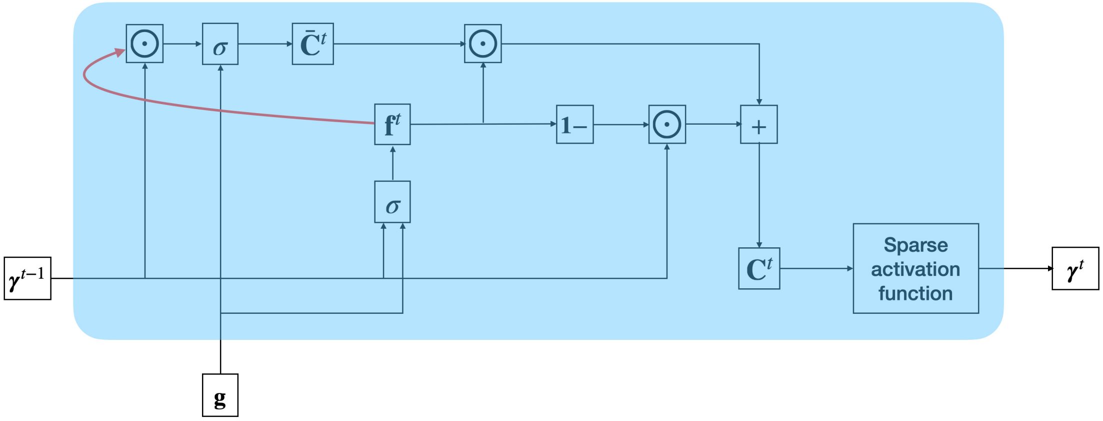


Fig. 6. Structure of the proposed SMGU. **f** indicated the only gate in each SMGU.

zero leading to the undesired result being always greater than 0.5 after activation. To tackle this problem, we employed the "tanh" function instead of sigmoid to guarantee that the value of the forget gate vector varies from 0 to 1 after activation, as it is originally designed. By applying the aforementioned adaptions, we have the formulations of CV-SMGU, as listed in Table I as well. The symbols $\tilde{\mathbf{W}}_*$, $\tilde{\mathbf{g}}$, and $\tilde{\boldsymbol{\gamma}}^*$ represent

$$\tilde{\mathbf{W}}_* = \begin{bmatrix} \mathrm{Re}(\mathbf{W}_*) & -\mathrm{Im}(\mathbf{W}_*) \\ \mathrm{Im}(\mathbf{W}_*) & \mathrm{Re}(\mathbf{W}_*) \end{bmatrix}$$

$$\tilde{\mathbf{g}} = \begin{bmatrix} \mathrm{Re}(\mathbf{g}) \\ \mathrm{Im}(\mathbf{g}) \end{bmatrix}$$

$$\tilde{\boldsymbol{\gamma}}^* = \begin{bmatrix} \mathrm{Re}(\hat{\boldsymbol{\gamma}}^*) \\ \mathrm{Im}(\hat{\boldsymbol{\gamma}}^*) \end{bmatrix}$$

where $\mathrm{Re}(\cdot)$ and $\mathrm{Im}(\cdot)$ denote the real and imaginary operators, respectively. $\eta_{cv-dt}(\cdot)$ is the CV version of the double hyperbolic function applied componentwise and expressed as follows:

$$\eta_{cv-dt}(\hat{\boldsymbol{\gamma}}, s, \theta) = \begin{cases} \dfrac{\hat{\boldsymbol{\gamma}}_i}{|\hat{\boldsymbol{\gamma}}_i|} s \cdot e^{j\cdot\angle(\hat{\gamma})}\left[\tanh(|\hat{\boldsymbol{\gamma}}|+\theta) + \tanh(|\hat{\boldsymbol{\gamma}}| - \theta)\right], & |\hat{\boldsymbol{\gamma}}_i| \neq 0 \\ 0, & \text{else.} \end{cases} \tag{9}$$

Table II summarizes and compares the features of different unrolled RNNs. Through experiments, we found that gated unrolled RNNs require significantly fewer layers to achieve comparable or even better performance. Moreover, the SMGU simplifies the model structure by coupling the two gates, thus significantly eliminating the number of free trainable parameters. Even if the CV-SMGU duplicates the number of parameters for determining the gate, it will not induce a serious memory burden or computational expense.

IV. Performance Evaluation

A. Simulation Setup and Model Training

In the simulation, we applied the same settings as [23], i.e., 25 regularly distributed spatial baselines in the range of −135 to 135 m were simulated. The corresponding inherent elevation resolution, i.e., Rayleigh resolution, amounts to about 42 m.

In the experiment, about four million training samples, half of which are single scatterer and the others are two-scatterer mixtures, were simulated to generate the training dataset. To make the training dataset adequate and the simulation more realistic, we randomized many parameters, i.e., SNR level, amplitude, phase, and elevation position of scatterers, when we simulated the training samples. The simulation details of single scatterer and double scatterers are listed as follows.

1) *Single Scatterer:* For single scatterer, the scattering phase ϕ is set to follow a uniform distribution, i.e., $\phi \sim U(-\pi, \pi)$. In addition, the amplitude A of the scatterer is simulated to be uniformly distributed in the range of $(1, 4)$. Hereafter, the CV scattering coefficient γ can be generated by $\gamma = A \cdot \exp(j\phi)$. The elevations of the simulated scatterers are regularly distributed on a 1-m grid between −20 and 300 m. Once the elevation is determined, the echo signal $\mathbf{g} \in \mathbb{C}^{25}$ is generated with different levels of SNR, which is regularly distributed between [0 dB, 10 dB] with 11 samples.
2) *Double Scatterers:* We simulated two single scatterers inside each resolution unit. The simulation of the two single scatterers is identical to the previous step. As a consequence, different amplitude ratios, different scattering phase offsets, and different elevation distances between the two scatterers are considered.

The model was implemented and trained under the framework of Pytorch [35]. The employed optimizer was Adam [36]. The learning rate was set to be adaptive according to the number of training epochs with the initial value being 0.0001. The loss function over the training data $\{(\mathbf{g}_i, \boldsymbol{\gamma}_i)\}_{i=1}^T$ is mean square error (mse) loss, which is defined as follows:

$$\underset{\boldsymbol{\Psi}}{\text{minimize}}\ \mathcal{L}(\boldsymbol{\Psi}) = \frac{1}{T}\sum_{i=1}^{T} ||\hat{\boldsymbol{\gamma}}(\boldsymbol{\Psi}, \mathbf{g}_i) - \boldsymbol{\gamma}_i||_2^2 \tag{10}$$

where $\boldsymbol{\Psi}$ denotes the set of all parameters to be learned from data. To determine the optimal structure of the network, we validated the performance of the network with different numbers of CV-SMGUs in terms of normalized mse (NMSE) on a validation dataset. The validation dataset was composed of 50 000 noise-free samples simulated using the same settings

TABLE II
COMPARISON OF DIFFERENT UNROLLED RNNS FOR SPARSE RECONSTRUCTION

Features	**γ-net**	**sc2net**	**SMGU**	**CV-SMGU**
complex-value	Yes	No	No	Yes
gates expression	No	Yes	Yes	Yes
number of gates	0	2	1	1
number of parameters for gates	0	$2 \cdot (L^2 + NL)$	$L^2 + NL$	$2 \cdot (L^2 + NL)$
required number of layers	≈ 15	≈ 5	≈ 5	≈ 5

introduced in Section III, and the NMSE is defined as follows:

$$\text{NMSE} = \frac{1}{T}\sum_{i=1}^{T}\frac{\|\hat{\boldsymbol{\gamma}}_i - \boldsymbol{\gamma}_i\|_2^2}{\|\boldsymbol{\gamma}_i\|_2^2}. \tag{11}$$

As we can see from Table III, the NMSE gradually converges with increasing the number of SMGUs. Moreover, after six CV-SMGUs, a further increase in the number of CV-SMGUs leads to marginal performance improvement. Instead, a heavier computational burden will be brought about. Therefore, the network we designed is composed of six CV-SMGUs.

B. Performance Assessment and Comparison to γ-Net

In this section, we carry out experiments to systematically evaluate the performance of the proposed algorithm in terms of super-resolution power, estimation accuracy, and generalization ability against different amplitude ratios and phase differences of scatterers.

C. Super-Resolution Power and Estimation Accuracy

The first experiment sets out to study the super-resolution power and estimation accuracy of the proposed algorithm via a TomoSAR benchmark test [5], [14]. In the experiment, we mimicked a facade-ground interaction by simulating two-scatterer mixtures with increasing elevation distance between them. The double scatterers were simulated to have identical phase and amplitude, i.e., the worst case for TomoSAR processing [13]. The proposed algorithm and $\boldsymbol{\gamma}$-Net were employed to resolve overlaid double scatterers at two SNR levels, i.e., SNR$\in \{0, 6\}$ dB, which represents typical SNR levels of a high-resolution spaceborne SAR image. We use the effective detection rate defined in [23] to fairly evaluate the super-resolution power. An effective detection should satisfy the following three criteria.

1) The hypothesis test correctly decides two scatterers for a double-scatterers signal.
2) The estimated elevation of ***both*** detected double scatterers are within ±3 times CRLB w.r.t. their true elevation.
3) Both elevation estimates are also within $\pm 0.5\ d_s$ w.r.t. their true elevation.

Here, d_s indicates the distance between the double scatterers. Fig. 7 compares the effective detection rate P_d of the proposed algorithm and γ-Net. It is presented as a function of the normalized distance α, which is the ratio of the scatterers' distance and the Rayleigh resolution $\alpha = d_s/\rho_s$. For each combination of SNR and α, we simulated 0.2 million Monte Carlo trials. From Fig. 7, one can see that the proposed algorithm and sc2net with CV-SLSTMs (CV-sc2net) have quite similar performance in terms of effective detection rate. This is the same as we expected since the CV-SMGU is constructed by simplifying the CV-SLSTM. The purpose of CV-SMGU is to reduce network components while maintaining performance. The advantages of the proposed algorithm compared to CV-sc2net are analyzed and discussed in Section V. When we compare the proposed algorithm and CV-sc2net to γ-Net, we can see that both the proposed algorithm and CV-sc2net outperform γ-Net by a fair margin at both SNR levels. Specifically, they are able to deliver 10%–20% higher effective detection rate in moderate super-resolving cases at 6-dB SNR. In the noisy case at 0-dB SNR, the proposed algorithm and CV-sc2net gradually approach about 90% effective detection rate with the increase in the normalized distance, whereas γ-Net reaches only about 70% effective detection rate. The superior performance of the proposed algorithm and CV-sc2net attributes to that they overcome the information loss in the dynamics of the network by incorporating historic data and preserving full information. As we have mentioned in Section II, the detection of double scatterers is affected by information loss. We cannot detect the scatterers whose information is discarded.

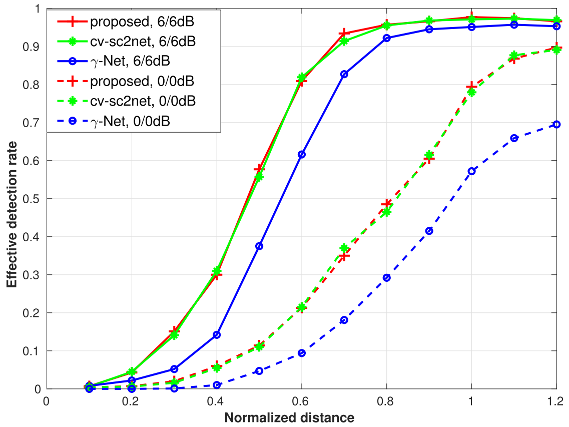


Fig. 7. Effective detection rate of the proposed algorithm, CV-sc2net and $\boldsymbol{\gamma}$-Net as a function of the normalized elevation distance between the simulated facade and ground with SNR = 0 and 6 dB under 0.2 million Monte Carlo trials.

To better manifest how the incorporation of historic information improves the performance, we simulated 2000 samples containing double scatterers with increasing scatterers distance at 6-dB SNR. We made a scatter plot of their elevation estimates and color-coded the points by the detector decision in Fig. 8. The x-axis refers to the true normalized elevation distance of the scatterers. The y-axis shows their normalized elevation estimates. The ideal reconstruction would be a horizontal and a diagonal straight line, which represents the ground

TABLE III

PERFORMANCE OF THE NETWORK WITH DIFFERENT NUMBER OF SMGUS. AFTER 6 SMGUS, THE PERFORMANCE IMPROVEMENT IS MARGINAL WITH INCREASING THE NUMBER OF SMGUS. INSTEAD, THE INCREASE IN SMGUS LEADS TO HEAVIER COMPUTATIONAL BURDEN

Number of CV-SMGUs	2	3	4	5	6	7	8	9
NMSE [dB]	-12.4	-20.7	-26.1	-29.6	-30.2	-30.6	-30.8	-30.9

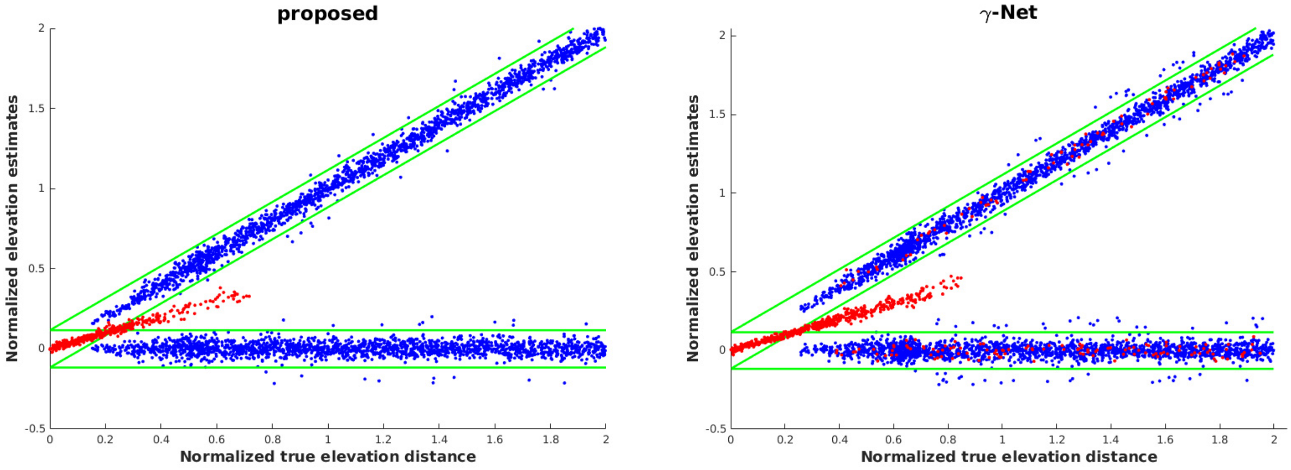


Fig. 8. Normalized estimated elevation of facade and ground of increasing elevation distance with SNR = 6 dB and N = 25. The double scatterers were simulated to have identical phase and amplitude. The true positions are a horizontal line referring to the ground and a diagonal line referring to the scatterers at variable elevations. The green lines depict true positions ± 3 times CRLB of elevation estimates for single scatterers. Red dots represent samples detected as single scatterers. Blue dots indicate detected overlaid double scatterers.

truth of the simulated ground and facade. The green lines refer to ground truth ±3 times CRLB of single scatterer elevation estimate. The blue dots indicate the detected double scatterers, whereas the red dots represent that the samples were detected as single scatterers, meaning that the second scatterer was lost in the network output. Fig. 8 clear shows the following.

1) γ-Net experiences many more red dots locate within ±3 times CRLB w.r.t. the ground truth, meaning that it occasionally can only detect one of the double scatterers but is able to estimate its elevation with high precision. We ascribe this problem to the information loss caused by the learning structure of γ-Net. On the contrary, the proposed algorithm utilizes CV-SMGUs to preserve full information, thus avoiding discarding any significant information.
2) The proposed algorithm is able to resolve double scatterers at much smaller scatterers' distances. Specifically, the proposed algorithm starts to separate double scatterers from about 0.15 Rayleigh resolution, whereas γ-Net can only detect double scatterers only after about 0.3 Rayleigh resolution.

The elevation estimates of the simulated facade and ground are plotted in Fig. 9 w.r.t. the normalized true elevation distance. The red horizontal and slant lines indicate the ground truth of the ground and façade, respectively. The black dashed curves represent the ground truth ±1× CRLB. The error bars indicate the standard deviation of the elevation estimates with the midpoint depicting the mean value of the elevation estimates at the given normalized true elevation distance. We discarded the points below an effective detection rate of 5% in the figures. Due to the strict criteria of the effective detection, both the proposed algorithm and γ-Net provide high elevation estimation accuracy, especially at 6-dB SNR, where the bias of the elevation estimates derived by both methods approaches 0. However, in the extremely noisy case, we can see that the proposed algorithm is able to estimate the elevation with a slightly lower bias compared to γ-Net.

D. Performance w.r.t. Amplitude Ratio

In this experiment, we propose to study how the proposed algorithm performs at different amplitude ratios of double scatterers. The double scatterers were set to have identical phases. The SNR level was set as 6 dB. Fig. 10 compares the effective detection rate of the proposed algorithm and γ-Net at different amplitude ratios. As can be seen, the effective detection rate of both algorithms degrades with the increase in the amplitude ratio. The reason for the degradation of the effective detection rate is twofold. First, dark scatterers suffer from larger and larger bias with the increase in the amplitude ratio since their elevation estimates tend to approach the other more prominent scatterer. Second, at a high amplitude ratio, the energy of the second scatterer is closer to the noise level. In real-world applications, we usually see dark scatterers at a high amplitude ratio (≥4) as noise. However, by comparing the two algorithms, we can see from Fig. 10 that the proposed algorithm performs much better with the increase in the amplitude ratio than $\boldsymbol{\gamma}$-Net despite the fact that the effective detection rate is seriously affected. From our perspective, the better performance of the proposed algorithm attributes to that the estimates derived by the proposed algorithm preserve the full information; thus, we have a higher chance to retrieve weak signals of dark scatterers.

E. Performance w.r.t. Phase Difference

As it was investigated in [13], the super-resolution power depends strongly on the phase difference when double

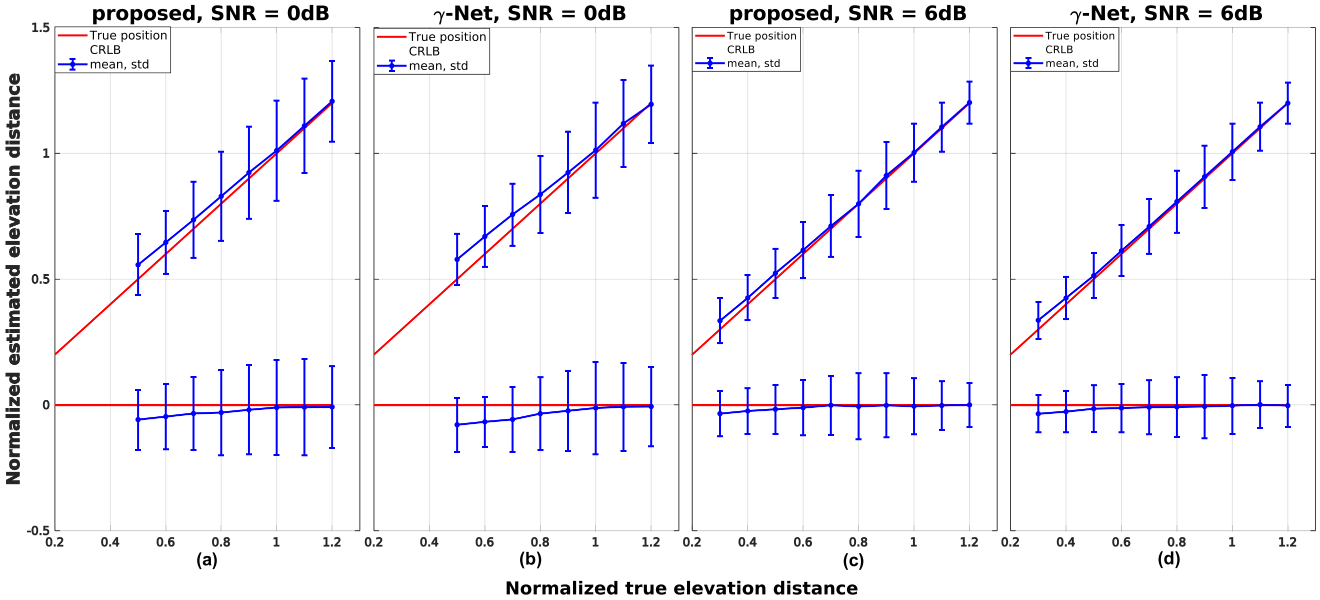


Fig. 9. Estimated elevation of simulated facade and ground: (a) SNR = 0 dB with the proposed algorithm, (b) SNR = 0 dB with $\boldsymbol{\gamma}$-Net, (c) SNR = 6 dB with the proposed algorithm, and (d) SNR = 6 dB with $\boldsymbol{\gamma}$-Net. Each dot has the sample mean of all estimates as its y value and the correspond standard deviation as the error bar. The red line segments represent the true elevation of the simulated facade and ground. The dashed curves denote the true elevation $\pm 1\times$ CRLB normalized w.r.t. the Rayleigh resolution.

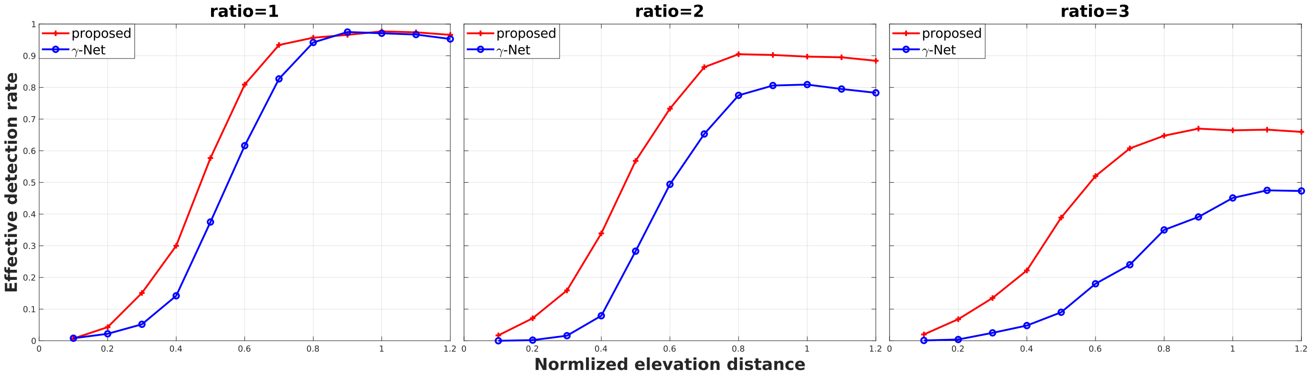


Fig. 10. Effective detection rate of the two algorithms w.r.t. the normalized elevation distance at different amplitude ratios.

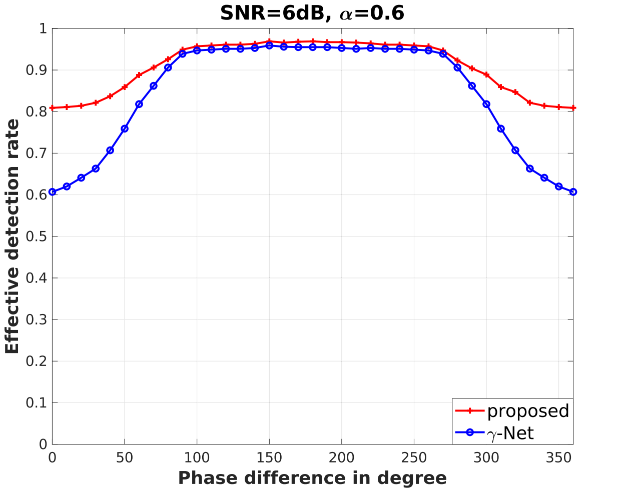


Fig. 11. Effective detection rate ρ_d of the two algorithms as a function of phase difference $\triangle\phi$ under the case: $N = 25$, SNR = 6 dB, and $\alpha = 0.6$.

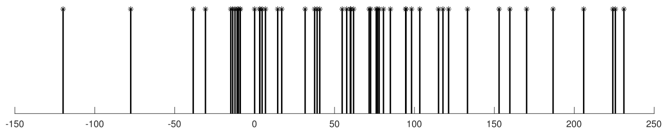


Fig. 12. Effective baselines of the 50 acquisitions.

scatterers were spaced within the Rayleigh resolution. To evaluate how the proposed algorithm performs w.r.t. phase difference of double scatterers in super-resolving cases, we vary the phase difference of simulated double scatterers in this experiment and test the effective detection rate. The double scatterers are simulated with identical amplitude. Fig. 11 illustrates the effective detection rate of the proposed algorithm and γ-Net for the case when $N = 25$ and SNR = 6 dB with $\alpha = 0.6$. As can be seen, both algorithms have the worst performance at $\triangle\phi = 0$ and perform better when $\triangle\phi$ approaches 180°. Compared to γ-Net, the proposed algorithm is less sensitive to the phase difference. When $\triangle\phi = 0$, the proposed algorithm delivers about 20% higher effective detection rate than $\boldsymbol{\gamma}$-Net.

F. Practical Demonstration

For the real data experiment, we used the test data stack over the city of Las Vegas covering Paris Hotel. The stack is composed of 50 TerraSAR-X high-resolution spotlight

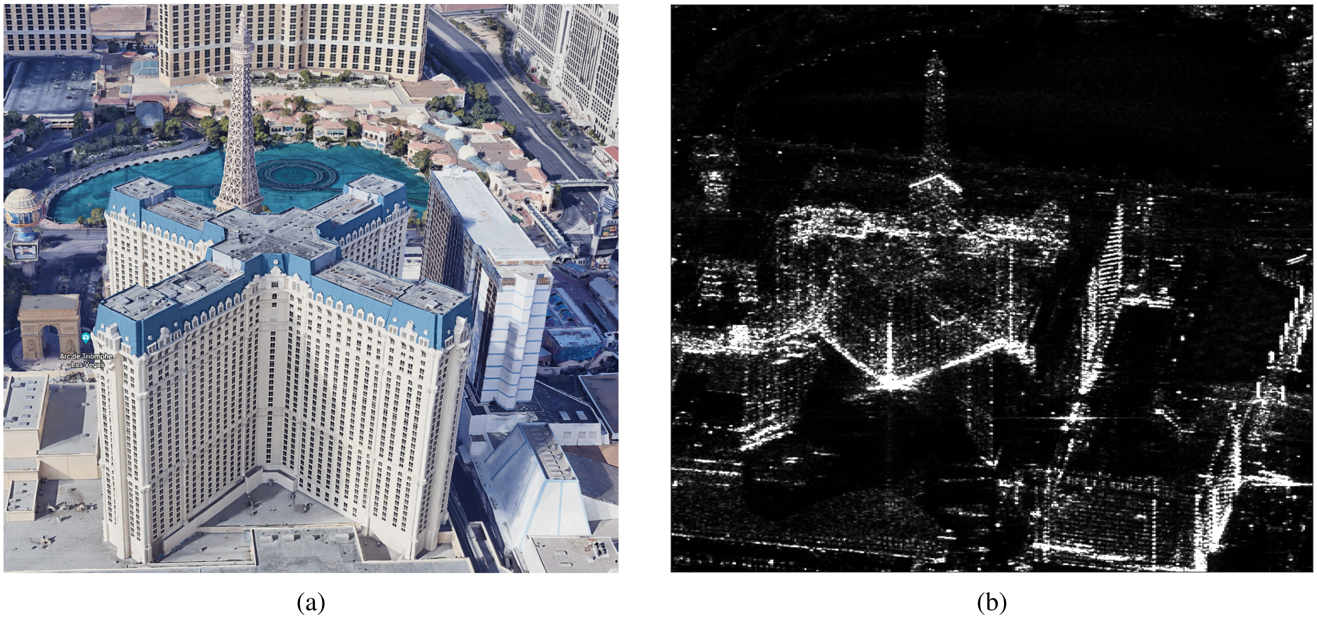


Fig. 13. Test site. (a) Optical image from Google Earth. (b) SAR mean intensity image.

TABLE IV
System Parameters of the TerraSAR-X High-Resolution Spotlight Image Stack

parameter	value
slant-range resolution	0.6m
azimuth resolution	1.1m
acquisition time	2008-2010
range distance	704km
incidence angle	31.8°

TABLE V
Percentage of Scatterers' Detection for the Two Algorithms

Algorithm	Percentage of detection as		
	0 scatterer	1 scatterer	2 scatterers
proposed	62.01 %	33.30 %	4.69 %
γ-Net	61.06 %	35.83 %	3.11 %

images with a slant-range resolution of 0.6 m and an azimuth resolution of 1.1 m, whose spatial baseline distribution is demonstrated in Fig. 12. In Fig. 13, an optical image from Google Earth and the SAR mean intensity image of the test site are demonstrated. The images were acquired between 2008 and 2010. More details of the data stack that we use are listed in Table IV.

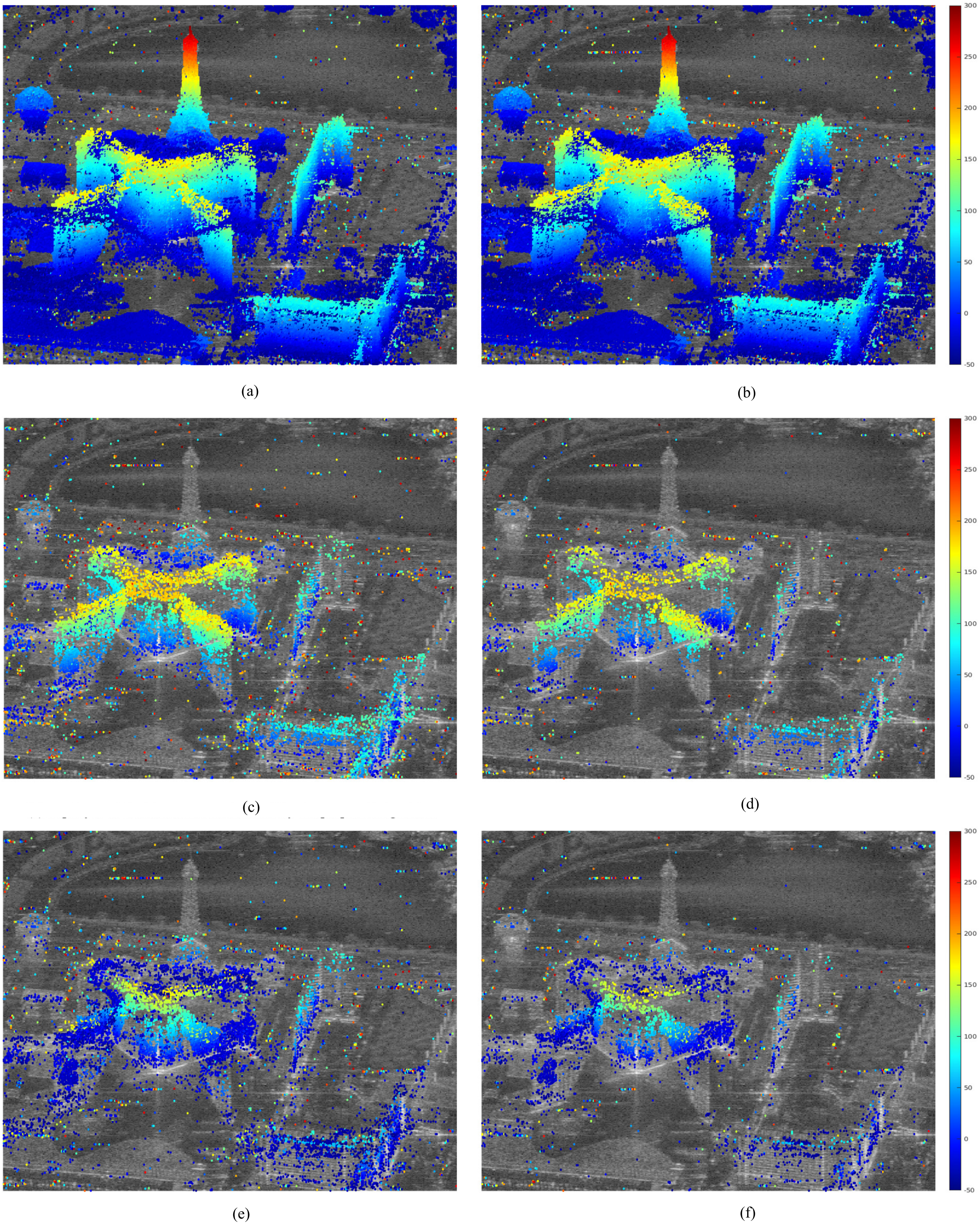
We employed the DLR's integrated wide area processor (IWAP) [37] to carry out preprocessing like multiple SAR images' coregistration and phase calibration. In addition, a coherence point on the ground was chosen as a reference.

We used the baselines of the test data stack to simulate training data. The simulation was conducted in the same way as introduced in the simulation setup in Section IV-A, and four million training samples were generated. When the network was well-trained, the proposed algorithm was directly applied to reconstruct the elevation of the test site.

The reconstruction results of the test site are demonstrated in Fig. 14 and compared to the results derived by γ-Net. Fig. 14(a) and (b) illustrates color-coded elevation of single scatterers detected by both algorithms. Fig. 14(c)–(f) depicts the reconstruction of detected double scatterers of both algorithms. The double scatterers are separated into the top and bottom layers according to their elevation estimates, and the top and bottom layers are demonstrated separately. By comparing the reconstruction results of both algorithms, we can see that the proposed algorithm detects the double scatterers with a higher density, indicating that the proposed algorithm has stronger super-resolution power. A closer inspection of the reconstruction of double scatterers shows that a serious layover exists on the top of the cross-building. Moreover, the elevation estimates of detected double scatterers indicate that the top layer is mainly caused by reflections from the building roof and building facade, whereas the bottom layer is composed of scatterers on the ground or lower infrastructures.

To provide a more intuitive comparison of the super-resolution power of both algorithms, we summarized the scatterers' detection of both algorithms in Table V. As it is shown in Table V, most pixels are detected as zero scatterers by the two algorithms because the fountain and many low infrastructures in the test site exhibit no strong scattering, which can be seen in Fig. 13(b). Compared to γ-Net, the proposed algorithm detected fewer single scatterers (33.30%) but more double scatterers. Comparison between the double scatterers detected by both algorithms shows that the proposed algorithm is able to detect 95.2% of the double scatterers detected by γ-Net. Moreover, it detects 50% more double scatterers than γ-Net.

Further investigation was conducted to inspect the improvement of double-scatterer detection. The histogram of detected double scatterers' elevation difference from the proposed algorithm and $\boldsymbol{\gamma}$-Net is shown in Fig. 15. In the nonsuper-resolution region, especially when the distance between double scatterers is larger than twice Raleigh resolution, the two algorithms have a comparable performance of double scatterers' detection. However, in the super-resolution region,



(a) (b) (c) (d) (e) (f)

Fig. 14. Reconstructed and color-coded elevation of detected scatterers. From (Left) to (Right): elevation estimates derived by the proposed algorithm and γ-Net, respectively. From (Top) to (Bottom): Color-coded elevation of detected single scatterers, the top layer of detected double scatterers, and the bottom layer of detected double scatterers, respectively. (a) Single scatterer detected by the proposed algorithm. (b) Single scatterer detected by γ-Net. (c) Top layer of double scatterers detected by the proposed algorithm. (d) Top layer of double scatterers detected by γ-Net. (e) Bottom layer of double scatterers detected by the proposed algorithm. (f) Bottom layer of double scatterers detected by γ-Net.

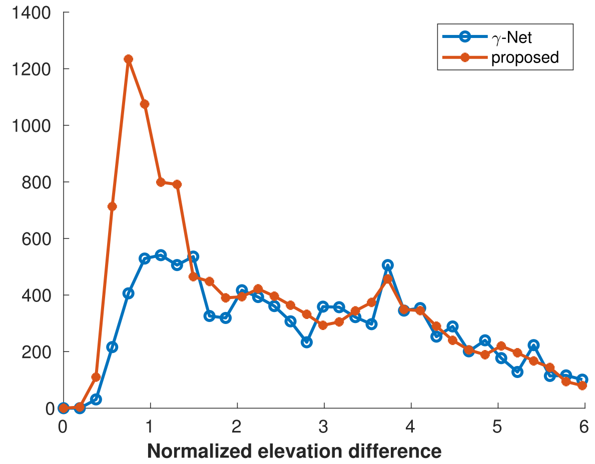


Fig. 15. Histogram of the elevation distance between the detected double scatterers from the proposed algorithm and γ-Net. The proposed algorithm shows significantly more detection in the super-resolution region.

the proposed algorithm delivers obviously stronger resolution ability.

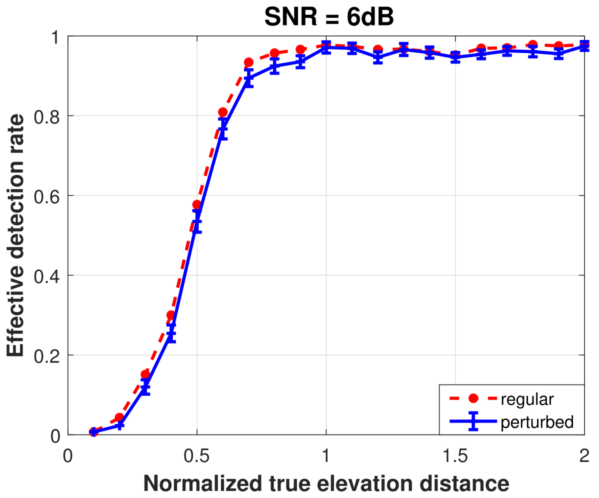


Fig. 16. Effective detection rate as a function of α at different baselines' distributions. The proposed algorithm shows a good generalization ability against baselines' discrepancy with the effective detection rate decreasing only 5%–8%.

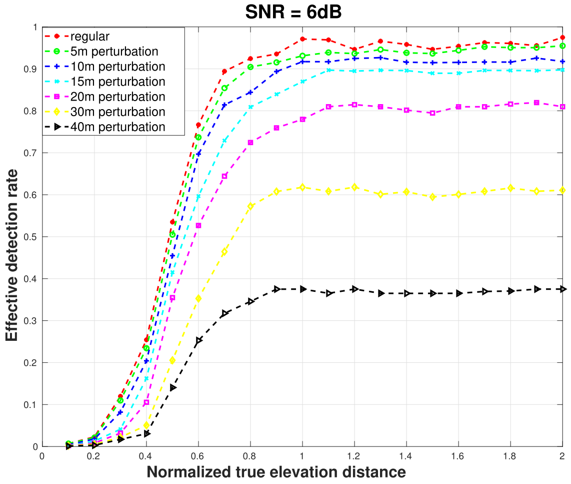


Fig. 17. Effective detection rate as a function of α at baselines with increasing perturbation. First, the effective detection rate decreases slowly with the increase in the baseline perturbation. When the perturbation is larger than 15 m, the performance of the proposed algorithm degrades dramatically.

V. Discussion

A. Generalization Ability Against Baselines' Discrepancy

The effective baseline in an SAR image varies according to the range and azimuth location. A deep learning model trained with a fixed set of baselines may have undesired performance when being applied to the whole image stack, as baseline discrepancies between training and testing data may cause data domain shifts. In this experiment, we verify the generalization ability against baseline discrepancies. The network with six CV-SMGUs is trained using 25 regularly distributed baselines as introduced in the simulation setup. Then, we add random perturbation uniformly distributed in the range [5 m, 10 m], i.e., about [7%, 14%] of the standard deviation of the 25 regularly distributed baselines, to the 25 regularly distributed baselines. 100 different baselines' distributions were generated. For each baseline distribution, we carry out a Monte Carlo simulation at 6-dB SNR for each baselines' distribution with 0.2 million Monte Carlo trials at each discrete normalized distance. Fig. 16 demonstrates the effective detection rate of the proposed algorithm when we apply the pretrained network to the data generated with baseline perturbations. The red line represents the reference, i.e., the pretrained network is applied to data simulated with the same baselines' distribution. The green line indicates the average effective detection rate of the 100 Monte Carlo simulations with the blue error bars depicting the standard deviation. As one can see, the proposed algorithm shows a good generalization ability against baselines' discrepancy with the effective detection rate decreasing only 5%–8% compared to the reference. Therefore, we see the proposed algorithm as a promising tool for large-scale TomoSAR processing since the biggest baselines' difference of a typical spaceborne SAR image will not exceed the perturbation that we simulated.

However, for baselines with large perturbations or even completely different distributions, the proposed algorithm is not an estimation efficient method. We carried out an additional experiment to test the boundary of the generalization ability by further increasing baseline perturbation. As we can see in Fig. 17, with the increase in the baseline discrepancy, the effective detection rate decreases slowly at first. When the perturbation is larger than 15 m, the performance of the proposed algorithm degrades dramatically. According to the test result, it indicates that 15 m might be the boundary for the proposed algorithm to have reasonable performance for the baseline setting in this simulation.

When we set out sights on global urban mapping using TomoSAR, the huge discrepancy between baselines of different data stacks will be a severe challenge. We still need to explore a more general and also computationally efficient algorithm.

B. Convergence Analysis

In this section, we propose to investigate the influence of CV-SGMUs on convergence performance in comparison with CV-SLSTMs. We use an RNN with six CV-SLSTMs

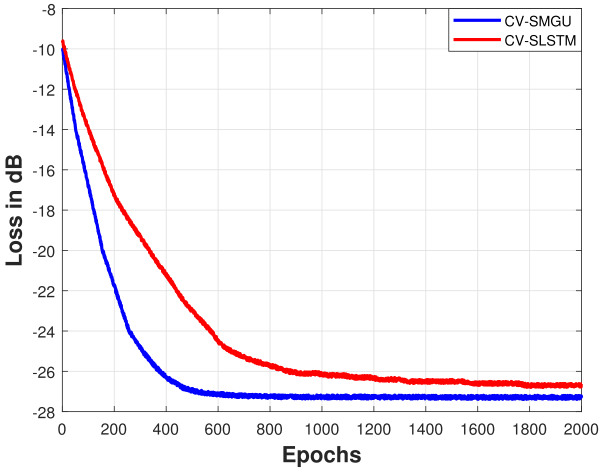


Fig. 18. Training loss [dB] versus epochs on simulated data. CV-SMGUs have faster convergence and lower overall loss.

as a baseline. Fig. 18 compares the objective loss [see (10)] with increasing training epochs. From Fig. 18, we can observe that CV-SMGUs contribute to faster convergence. To be specific, the RNN with CV-SMGUs needs only about 500 epochs to achieve convergence, while the RNN with CV-SLSTMs requires more than 1000 epochs to converge. Furthermore, CV-SMGUs lead to slightly lower overall costs than CV-SLSTMs.

C. Requirement of Training Data

As we have clarified in Section III-B, the CV-SMGU has only one gate, i.e., the minimum number of gates; thus, it has fewer trainable parameters and a simpler structure. In this experiment, we study how this simpler model contributes to reducing the requirement for training data. We compare two RNNs with six CV-SMGUs and six CV-SLSTMs, respectively, in terms of effective detection rate at 6-dB SNR. The distance between double scatterers was fixed at 0.6 Rayleigh resolution, and the double scatterers were set to have identical phase and amplitude. The result is demonstrated in Fig. 19. As can be seen, the RNN with CV-SMGUs has a better performance when the two RNNs are trained with the same amount of training samples. In addition, the RNN with CV-SMGUs requires obviously fewer training samples to achieve optimal performance.

VI. Conclusion

In this article, we proposed a novel gated RNN-based BPDN solver for sparse reconstruction. The proposed gated RNN adopted a novel architecture, termed SMGU, to avoid information loss caused by shrinkage by incorporating historical information into optimization. With the assistance of SMGUs, we are able to capture and maintain long-term dependence on information in previous layers. To be specific, important information will be automatically accumulated, while useless or redundant information will be forgotten in the dynamic of the network. Moreover, we extended the SMGU to the CV domain as CV-SMGU and applied it to solve TomoSAR inversion. Laboratory and real data experiments demonstrated that the proposed gated RNN built with CV-SMGUs outperforms the state-of-the-art deep learning-based TomoSAR method $\boldsymbol{\gamma}$-Net. The encouraging results open up a new prospect for SAR tomography using deep learning and motivate us to further investigate the potential of RNNs with gated units in practical TomoSAR processing.

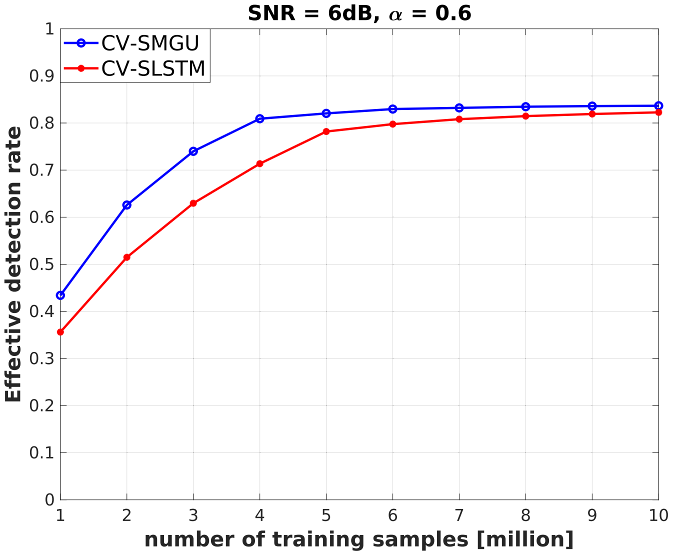


Fig. 19. Effective detection rate versus the number of training samples. The RNN with CV-SMGUs requires less training samples to achieve optimal performance.

Appendix
$\boldsymbol{\gamma}$-Net Formulation

Fig. 20 illustrates a K-layer $\boldsymbol{\gamma}$-Net. Each block in Fig. 20 indicates one layer of $\boldsymbol{\gamma}$-Net and is formally defined as

$$\tilde{\boldsymbol{\gamma}}_i = \eta_{ss_{\theta_i}^{\rho^i}}\left\{\tilde{\boldsymbol{\gamma}}_{i-1} + \tilde{\mathbf{W}}^i\left(\tilde{\mathbf{g}} - \tilde{\mathbf{R}}\tilde{\boldsymbol{\gamma}}_{i-1}\right), \boldsymbol{\theta}_i\right\} \tag{12}$$

where

$$\tilde{\mathbf{W}}^i = \begin{bmatrix} \mathrm{Re}\left(\mathbf{W}^i\right) & -\mathrm{Im}\left(\mathbf{W}^i\right) \\ \mathrm{Im}\left(\mathbf{W}^i\right) & \mathrm{Re}\left(\mathbf{W}^i\right) \end{bmatrix}, \quad \tilde{\mathbf{R}} = \begin{bmatrix} \mathrm{Re}(\mathbf{R}) & -\mathrm{Im}(\mathbf{R}) \\ \mathrm{Im}(\mathbf{R}) & \mathrm{Re}(\mathbf{R}) \end{bmatrix}$$

$$\tilde{\mathbf{g}} = \begin{bmatrix} \mathrm{Re}(\mathbf{g}) \\ \mathrm{Im}(\mathbf{g}) \end{bmatrix}, \quad \tilde{\boldsymbol{\gamma}}_i = \begin{bmatrix} \mathrm{Re}\left(\hat{\boldsymbol{\gamma}}_i\right) \\ \mathrm{Im}\left(\hat{\boldsymbol{\gamma}}_i\right) \end{bmatrix}.$$

$\boldsymbol{\theta}_i = [\theta_i^1, \theta_i^2, \ldots, \theta_i^5]$ denotes the set of parameters to be learned for the piecewise linear function in the ith layer. $\mathbf{W}^i$ indicates the trainable weight matrix in the ith layer, and it is initialized using the system steering matrix $\mathbf{R}$ with $\mathbf{W^i} = \beta\mathbf{R}^H$. β is the step size. Usually, a proper step size can be taken as $(1/L_s)$, with L_s being the largest eigenvalue of $\mathbf{R^H R}$. $\hat{\boldsymbol{\gamma}}_i$ is the output of the ith layer. Re$(\cdot)$ and Im$(\cdot)$ denote the real and imaginary operators, respectively.

SS in $\boldsymbol{\gamma}$-Net indicates a special thresholding scheme called support selection, which is formally defined as follows:

$$\eta_{ss_{\theta_i}^{\rho^i}}\left(\tilde{\boldsymbol{\gamma}}_i\right) = \begin{cases} \tilde{\boldsymbol{\gamma}}_i & i \in \mathcal{S}^{\rho^i}(\tilde{\boldsymbol{\gamma}}) \\ \eta_{pwl}\left(\tilde{\boldsymbol{\gamma}}_i, \boldsymbol{\theta}_i\right) & i \notin \mathcal{S}^{\rho^i}(\tilde{\boldsymbol{\gamma}}). \end{cases} \tag{13}$$

In the ith layer, the support selection will select ρ^i percentage of entries with the largest magnitude and trust them as “true support,” which will be directly fed to the next layer, bypassing the shrinkage step. The remaining part will go through the shrinkage step as usual. The shrinkage is executed using the

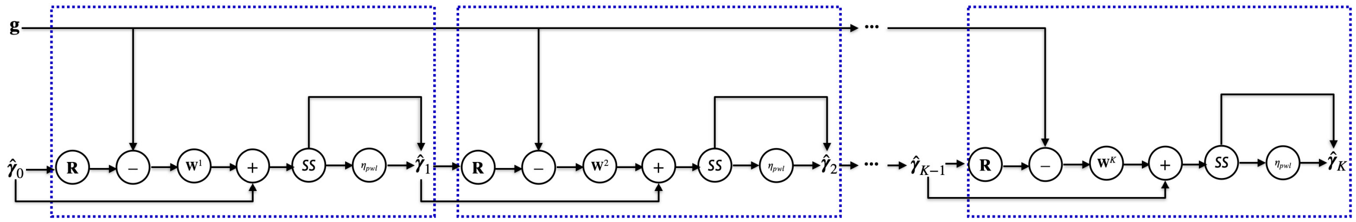


Fig. 20. Illustration the learning architecture of a K-layer $\boldsymbol{\gamma}$-Net.

piecewise linear function η_{pwl}, which is a novel shrinkage thresholding function to promote sparsity while improving convergence rate and reducing reconstruction error in the meanwhile and expressed as

$$\eta_{pwl}(\hat{\boldsymbol{\gamma}}, \boldsymbol{\theta}_i) = \begin{cases} \theta_i^3 \hat{\boldsymbol{\gamma}}, & |\hat{\boldsymbol{\gamma}}| \le \theta_i^1 \\ e^{j \cdot \angle \hat{\boldsymbol{\gamma}}} \left[\theta_i^4 \left(|\hat{\boldsymbol{\gamma}}| - \theta_i^1 \right) + \theta_i^3 \theta_i^1 \right], & \theta_i^1 < |\hat{\boldsymbol{\gamma}}| \le \theta_i^2 \\ e^{j \cdot \angle \hat{\boldsymbol{\gamma}}} \left[\theta_i^5 \left(|\hat{\boldsymbol{\gamma}}| - \theta_i^2 \right) + \theta_i^4 \left(\theta_i^2 - \theta_i^1 \right) + \theta_i^3 \theta_i^1 \right], & |\hat{\boldsymbol{\gamma}}| > \theta_i^2. \end{cases} \tag{14}$$

References

[1] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, Jan. 2001.

[2] X. X. Zhu and R. Bamler, "A sparse image fusion algorithm with application to pan-sharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 5, pp. 2827–2836, May 2013.

[3] J. Bieniarz, E. Aguilera, X. X. Zhu, R. Müller, and P. Reinartz, "Joint sparsity model for multilook hyperspectral image unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 4, pp. 696–700, Apr. 2015.

[4] B. Zhang, W. Hong, and Y. Wu, "Sparse microwave imaging: Principles and applications," *Sci. China Inf. Sci.*, vol. 55, no. 8, p. 33, 2012.

[5] X. X. Zhu and R. Bamler, "Tomographic SAR inversion by L_1-norm regularization—The compressive sensing approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 10, pp. 3839–3846, Jun. 2010.

[6] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[7] R. G. Baraniuk, "Compressive sensing," *IEEE Signal Process. Mag.*, vol. 24, no. 2, pp. 118–121, Jan. 2007.

[8] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[9] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413–1457, Nov. 2004.

[10] Y. Li and S. Osher, "Coordinate descent optimization for L_1 minimization with application to compressed sensing; a greedy algorithm," *Inverse Problems Imag.*, vol. 3, no. 3, pp. 487–503, 2009.

[11] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Nov. 2010, doi: 10.1561/2200000016.

[12] S. J. Wright, *Primal-Dual Interior-Point Methods*. Society for Industrial and Applied Mathematics, 1997.

[13] X. X. Zhu and R. Bamler, "Super-resolution power and robustness of compressive sensing for spectral estimation with application to spaceborne tomographic SAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 1, pp. 247–258, Jan. 2012.

[14] X. X. Zhu and R. Bamler, "Very high resolution spaceborne SAR tomography in urban environment," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 12, pp. 4296–4308, Dec. 2010.

[15] G. Fornaro, F. Serafino, and F. Soldovieri, "Three-dimensional focusing with multipass SAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 3, pp. 507–517, Mar. 2003.

[16] J. R. Hershey, J. L. Roux, and W. Felix, "Deep unfolding: Model-based inspiration of novel deep architectures," 2014, *arXiv:1409.2574*, doi: 10.48550/arXiv.1409.2574.

[17] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. 27th Int. Conf. Int. Conf. Mach. Learn.* Madison, WI, USA: Omnipress, 2010, pp. 399–406.

[18] Y. Yang, J. Sun, H. Li, and Z. Xu, "ADMM-CSNet: A deep learning approach for image compressive sensing," *IEEE Trans. Pattern Anal. Mach. Intell.* vol. 42, no. 3, pp. 521–538, Mar. 2020.

[19] M. Wang et al., "CSR-Net: A novel complex-valued network for fast and precise 3-D microwave sparse reconstruction," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4476–4492, 2020.

[20] S. Wei, J. Liang, M. Wang, J. Shi, X. Zhang, and J. Ran, "AF-AMPNet: A deep learning approach for sparse aperture ISAR imaging and autofocusing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2022.

[21] J. Gao, Y. Ye, S. Li, Y. Qin, X. Gao, and X. Li, "Fast super-resolution 3D SAR imaging using an unfolded deep network," in *Proc. IEEE Int. Conf. Signal, Inf. Data Process. (ICSIDP)*, 2019, pp. 1–5, doi: 10.1109/ICSIDP47821.2019.9173392.

[22] S. Rangan, P. Schniter, and A. Fletcher, "Vector approximate message passing," *IRE Prof. Group Inf. Theory*, vol. 65, no. 10, pp. 6664–6684, 2019.

[23] K. Qian, Y. Wang, Y. Shi, and X. X. Zhu, "γ-Net: Superresolving SAR tomographic inversion via deep learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, 2022.

[24] X. Chen, J. Liu, Z. Wang, and W. Yin, "Theoretical linear convergence of unfolded ISTA and its practical weights and thresholds," 2018, *arXiv:1808.10038*.

[25] N. Qian, "On the momentum term in gradient descent learning algorithms," *Neural Netw.*, vol. 12, no. 1, pp. 145–151, 1999.

[26] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012, *arXiv:1212.5701*, doi: 10.48550/arXiv.1212.5701.

[27] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, Feb. 2011.

[28] J. T. Zhou et al., "SC2Net: Sparse LSTMs for sparse coding," in *Proc. 32th AAAI Conf. Artif. Intell.*, New Orleans, LA, USA, Feb. 2018, pp. 4588–4595.

[29] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Proc. NIPS Workshop Deep Learn.*, Dec. 2014, pp. 1–9.

[30] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *Proc. 32nd Int. Conf. Int. Conf. Mach. Learn.*, vol. 37, 2015, pp. 2342–2350.

[31] K. Greff, R. K. Srivastava, J. Koutnìk, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space Odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.

[32] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *NIPS Workshop Deep Learn.*, 2014, pp. 1–9.

[33] G.-B. Zhou, J. Wu, C.-L. Zhang, and Z.-H. Zhou, "Minimal gated unit for recurrent neural networks," *Int. J. Automat. Comput.*, vol. 13, no. 3, pp. 226–234, Jun. 2016.

[34] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1–15.

[35] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2019, pp. 8024–8035.

[36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*, doi: 10.48550/arXiv.1412.6980.

[37] F. R. Gonzalez, N. Adam, A. Parizzi, and R. Brcic, "The integrated wide area processor (IWAP): A processor for wide area persistent scatterer interferometry," in *Proc. ESA Living Planet Symp.*, Edinburgh, U.K., 2013. [Online]. Available: http://www.livingplanet2013.org/abstracts/850572.htm

Kun Qian received the double B.Sc. degree in remote sensing and information engineering from Wuhan University, Wuhan, China, and in aerospace engineering and geodesy from the University of Stuttgart, Stuttgart, Germany, in 2016, and the M.Sc. degree in aerospace engineering and geodesy from the University of Stuttgart in 2018. He is currently pursuing the Ph.D. degree in data science in Earth observation with the Technical University of Munich, Munich, Germany.

His research focus includes data-driven methods, deep unfolding algorithms, and their application in multibaseline synthetic aperture radar (SAR) tomography.

Yuanyuan Wang (Member, IEEE) received the B.Eng. degree (Hons.) in electrical engineering from The Hong Kong Polytechnic University, Hong Kong, China, in 2008, and the M.Sc. and Dr.Ing. degrees from the Technical University of Munich, Munich, Germany, in 2010 and 2015, respectively.

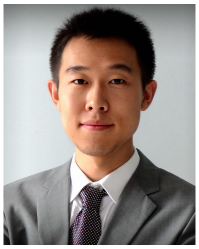
In June and July 2014, he was a Guest Scientist with the Institute of Visual Computing, ETH Zürich, Zürich, Switzerland. He is currently a Guest Professor with the German International AI Future Laboratory (AI4EO), Technical University of Munich. He is also with the Department of EO Data Science, Remote Sensing Technology Institute, German Aerospace Center, Wessling, Germany, where he leads the working group Big SAR Data. His research interests include optimal and robust parameters estimation in multibaseline synthetic aperture radar interferometry (InSAR) techniques, multisensor fusion algorithms of synthetic aperture radar (SAR) and optical data, nonlinear optimization with complex numbers, machine learning in SAR, uncertainty quantification and mitigation in machine learning, and high-performance computing for big data.

Dr. Wang was one of the best reviewers of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING in 2016.

Peter Jung (Member, IEEE) received the Dipl.Phys. degree in high-energy physics from Humboldt University, Berlin, Germany, in 2000, in cooperation with DESY Hamburg, Hamburg, Germany, and the Dr.rer.nat (Ph.D.) degree in Weyl–Heisenberg representations in communication theory from the Technical University of Berlin (TUB), Berlin, Germany, in 2007.

Since 2001, he has been with the Department of Broadband Mobile Communication Networks, Fraunhofer Institute for Telecommunications, Heinrich-Hertz-Institut (HHI), Berlin. Since 2004, he has been with the Fraunhofer German-Sino Laboratory for Mobile Communications, Berlin. He is currently working under Deutsche Forschungsgemeinschaft (DFG) grants at TUB in the field of signal processing, information and communication theory, and data science. He is also a Visiting Professor with the Technical University of Munich (TU Munich), Munich, Germany, and associated with the Munich AI Future Laboratory (AI4EO), Munich. He is giving lectures in compressed sensing, estimation theory, and inverse problems. His research interests include the area compressed sensing, machine learning, time–frequency analysis, dimension reduction, and randomized algorithms.

Dr. Jung is also a member of Verband Deutscher Elektrotechniker/Informationstechnische Gesellschaft (VDE/ITG).



Yilei Shi (Member, IEEE) received the Dipl.Ing. degree in mechanical engineering and the Dr.Ing. degree in signal processing from the Technische Universität München (TUM), Munich, Germany, in 2010 and 2019, respectively.

He is currently a Senior Scientist with the Chair of Remote Sensing Technology, TUM. His research interests include fast solver and parallel computing for large-scale problems, high-performance computing and computational intelligence, advanced methods on synthetic aperture radar (SAR) and SAR interferometry (InSAR) processing, machine learning and deep learning for a variety of data sources, such as SAR, optical images, and medical images, and partial differential equation (PDE)-related numerical modeling and computing.

Xiao Xiang Zhu (Fellow, IEEE) received the M.Sc. degree, the Dr.Ing. degree, and the Habilitation degree in signal processing from the Technical University of Munich (TUM), Munich, Germany, in 2008, 2011, and 2013, respectively.

She was the Founding Head of the Department of EO Data Science, Remote Sensing Technology Institute, German Aerospace Center (DLR), Germany. Since 2019, she has been a Co-Coordinator of the Munich Data Science Research School, Munich. Since 2019, she has been heading the Helmholtz Artificial Intelligence, Munich—research fields: aeronautics, space, and transport. Since May 2020, she has been the Director of the International Future AI Lab "AI4EO—Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond," Munich. Since October 2020, she has been the Co-Director of the Munich Data Science Institute (MDSI), TUM. She was a Guest Scientist or a Visiting Professor with the Italian National Research Council (CNR-IREA), Naples, Italy, Fudan University, Shanghai, China, The University of Tokyo, Tokyo, Japan, and the University of California at Los Angeles, Los Angeles, CA, USA, in 2009, 2014, 2015, and 2016, respectively. She is currently the Chair Professor for data science in Earth observation (former: signal processing in Earth observation) with TUM. She is currently a Visiting AI Professor with the Phi-Lab, European Space Agency (ESA), Paris. Her main research interests are remote sensing and Earth observation, signal processing, machine learning, and data science, with a special application focus on global urban mapping.

Dr. Zhu is a member of the Young Academy (Junge Akademie/Junges Kolleg) at the Berlin-Brandenburg Academy of Sciences and Humanities, the German National Academy of Sciences Leopoldina, and the Bavarian Academy of Sciences and Humanities. She serves on the Scientific Advisory Board in several research organizations, among others the German Research Center for Geosciences (GFZ) and the Potsdam Institute for Climate Impact Research (PIK). She is an Associate Editor of IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING. She serves as an Area Editor responsible for special issues of *IEEE Signal Processing Magazine*.