Speckle2Void: Deep Self-Supervised SAR Despeckling with Blind-Spot Convolutional Neural Networks

# Speckle2Void: Deep Self-Supervised SAR Despeckling with Blind-Spot Convolutional Neural Networks

Andrea Bordone Molini, Diego Valsesia, Giulia Fracastoro, and Enrico Magli

***Abstract*—Information extraction from synthetic aperture radar (SAR) images is heavily impaired by speckle noise, hence despeckling is a crucial preliminary step in scene analysis algorithms. The recent success of deep learning envisions a new generation of despeckling techniques that could outperform classical model-based methods. However, current deep learning approaches to despeckling require supervision for training, whereas clean SAR images are impossible to obtain. In the literature, this issue is tackled by resorting to either synthetically speckled optical images, which exhibit different properties with respect to true SAR images, or multi-temporal SAR images, which are difficult to acquire or fuse accurately. In this paper, inspired by recent works on blind-spot denoising networks, we propose a self-supervised Bayesian despeckling method. The proposed method is trained employing only noisy SAR images and can therefore learn features of real SAR images rather than synthetic data. Experiments show that the performance of the proposed approach is very close to the supervised training approach on synthetic data and superior on real data in both quantitative and visual assessments.**

***Index Terms*—SAR, despeckling, convolutional neural networks, self-supervised**

## I. Introduction

Synthetic Aperture Radar (SAR) is a coherent imaging system and as such it strongly suffers from the presence of speckle, a signal dependent granular noise. Speckle noise makes SAR images difficult to interpret, preventing the effectiveness of scene analysis algorithms for, e.g., image segmentation, detection and recognition. Several despeckling methods applied to SAR images have been proposed working either in spatial or transform domain. The first attempts at despeckling employed filtering-based techniques operating in spatial domain such as Lee filter [1], Frost filter [2], Kuan filter [3], and Gamma-MAP filter [4]. Wavelet-based methods [5], [6] enabled multi-resolution analysis. More recently, non-local filtering methods attempted to exploit self-similarities and contextual information. A combination of non-local approach, wavelet domain shrinkage and Wiener filtering in a two-step process led to SAR-BM3D [7], a SAR-oriented version of BM3D [8].

In recent years, deep learning techniques have become the benchmark in many image processing tasks, achieving exceptional results in problems such as image restoration [9], super resolution [10], semantic segmentation [11], and many more. Recently, some despeckling methods based on convolutional neural networks (CNNs) have been proposed [12], [13], attempting to leverage the feature learning capabilities of CNNs. Such methods use a supervised training approach where the network weights are optimized by minimizing a distance metric between noisy inputs and clean targets. However, clean SAR images do not exist and supervised training methods resort to synthetic datasets where optical images are used as ground truth and their artificially speckled version as noisy inputs. This creates a domain gap between the features of synthetic training data and those of real SAR images, possibly leading to the presence of artifacts or poor preservation of radiometric features when despeckling real SAR images. SAR-CNN [14] addressed this problem by averaging multi-temporal SAR data of the same scene in order to obtain an approximate (finite number of looks) ground truth. However, acquisition of multi-temporal data, scene registration and robustness to temporal variations can be challenging, leading to a sub-optimal rejection of speckle.

Recently, self-supervised denoising methods [15]–[18] proved, under certain assumptions, to be a valid alternative when it is not possible to have access to clean images. In particular, the two methods in [16], [18] deal with a single noisy version of each image in the dataset. These two works make use of a modified version of the classical CNN, called blind-spot convolutional network, to reconstruct each clean pixel exclusively from its neighboring pixels. The target pixel itself is kept hidden by the blind spot operation during training in order to prevent the network from learning the identity mapping and just copying the noisy pixel in the final denoised image. Self-supervision thus allows to exploit the potential of deep learning in those fields where the ground truth is not accessible, such as SAR imaging.

Inspired by these works, in this paper we present Speckle2Void, a self-supervised Bayesian despeckling framework that enables direct training on real SAR images. Our method bypasses the problem of training a CNN on synthetically-speckled optical images, thus avoiding any domain gap and enabling learning of features from real SAR images. It also avoids the inherent difficulty in constructing multitemporal datasets, as done in [14]. Our main contributions can be summarized as follows:

- we formulate a Bayesian model to characterize the speckle and the prior distribution of pixels in the clean SAR image, conditioned on their neighborhoods;

The authors are with Politecnico di Torino – Department of Electronics and Telecommunications, Italy. email: {name.surname}@polito.it. This research has been funded by the Smart-Data@PoliTO center for Big Data and Machine Learning technologies. This material is based upon work supported by Google Cloud.

- we propose an improved version of the blind-spot CNN architecture in [18] and a regularized training procedure with a variable blind-spot shape in order to account for the autocorrelation of the speckle process;
- we present two versions of Speckle2Void: a local version with classical convolutional layers and a non-local version to incorporate information from both spatially-neighboring as well as distant pixels to exploit self-similarity, albeit at higher computational complexity;
- we achieve remarkable despeckling performance, showing how our self-supervised approach is better than model-based techniques, close to the deep learning methods requiring supervised training on synthetic images and superior to them on real SAR data.

A preliminary version of this work appeared in [19], showing the basic principles of the proposed approach. This paper significantly expands the treatment with improvements on network modeling, on the loss function and on the training procedure. In particular, it solves the problem of the residual granularity in the despeckled images in [19], by showing the importance of properly decorrelating the speckle process and carefully designing the blind-spot shape.

The remainder of this paper is organized as follows. Section II introduces related works on SAR despeckling. Section III provides the background knowledge on the Bayesian framework adopted in this work. Section IV details the proposed statistical models and the regularized blind-spot network with variable structure. Section V contains results and performance evaluation. Section VI draws some conclusions.

## II. Related work

### A. SAR Despeckling

The last decades have seen a multitude of SAR image despeckling methods, that can be broadly categorized into four main approaches: spatial-domain methods, wavelet-domain methods, non-local methods and deep learning methods. Filtering-based techniques such as Lee filter [1], Frost filter [2], Kuan filter [3] represent the early attempts to solve SAR despeckling and they operate in spatial domain. Subsequent works in spatial domain aimed to reduce speckle under a non-stationary multiplicative speckle assumption. A popular example is represented by the Bayesian maximum a posteriori (MAP) approaches aiming to give a statistical description to the SAR image. A few MAP-based works have been proposed and the most representative is the $\Gamma$-MAP filter [4] that solves the MAP equation modeling both the radar reflectivity and the speckle noise with a Gamma distribution.

Wavelet-based methods proved to be more effective than spatial domain ones, enabling multi-resolution analysis and boosting analysis under non-stationary characteristics. They despeckle SAR images in the transform domain by estimating despeckled coefficients and then by applying the inverse transform to obtain the cleaned SAR image. A first subclass of wavelet based methods solve the despeckling problem with a homomorphic approach, consisting in applying a logarithmic transform of the data to convert the multiplicative noise into an additive one. The works in [20], [21] applied the traditional wavelet shrinkage based on hard- and soft-thresholding with an empirical selection of the threshold. Further wavelet-based methods [22]–[25] introduce prior knowledge about the log-transformed reflectance in the wavelet domain, employing a MAP estimator. Most of the wavelet-based homomorphic approaches do not compensate for the bias in the reconstructed images resulting from the mean of the log-transform speckle. To cope with this problem, a non-homomorphic approach has been considered by some works [26]–[29] in the wavelet domain, dealing with a signal-dependent speckle whose distribution parameters are harder to be estimated.

In general, both spatial domain and wavelet domain techniques yield limited detail preservation with the introduction of severe artifacts. The amount of information provided by a local window is quite limited and the need of incorporating more information from the neighborhood led to the proliferation of non-local methods. The pioneering work in this field is represented by the non-local mean (NLM) filter [30] that performs a weighted average of all pixels in the image and the weights depend on their similarity with respect to the target pixel. The weights are defined by computing the Euclidean distance between a surrounding patch centered at a neighboring pixel and a local patch centered at the target pixel. In [31], the Probabilistic Patch-Based (PPB) algorithm has been proposed to adapt the non-local means approach to SAR despeckling. The authors devised a patch similarity measure that generalizes to the case of multiplicative, non-Gaussian speckle.

In [32], the authors proposed another extension of NLM for despeckling, called NL-SAR, to deal with arbitrary SAR modalities (SAR, polarimetric SAR, interferometric SAR) and any number of looks. They proposed a unified nonlocal framework where several non-local estimations are performed and the best one is locally selected to ensure adaptivity to local structures. Moreover, in order to ensure robustness to noise correlation, similarities are weighted using kernels learned from a homogeneous region.

NLM inspired a number of extensions in the Gaussian noise context such as the Block-Matching 3D (BM3D) algorithm [8], a combination of non-local approach, wavelet domain shrinkage and Wiener filtering in a two-step process.

One of the most popular SAR despeckling algorithm is the SAR version of BM3D [8] (SAR-BM3D) that follows the same BM3D phases with an adaptation to the SAR statistics in the grouping phase where the same PPB similarity measure is used. Moreover the hard-thresholding and Wiener filtering, suitable in the Gaussian noise context, are replaced with an LMMSE estimator (based on an additive signal-dependent noise model).

The success of deep learning on many tasks involving image processing has suggested that the powerful learning capabilities of CNNs could be exploited for SAR despeckling and a few works have started addressing the problem. Chierchia et al. [14] proposed SAR-CNN, which applies a DnCNN-like [33] supervised denoising approach to SAR data. They exploit the homomorphic approach to deal with multiplicative noise model and use a new similarity measure for speckle noise distribution as loss function rather than the usual Euclidean distance. Clean data for training are obtained by averaging

multitemporal SAR images. Wang et al. [12] proposed a residual CNN (ID-CNN) trained on synthetic SAR images, to directly estimate the noise in the original domain, and, hence, the despeckled image is obtained by dividing the noisy image by the estimated noise. Training is once again supervised using synthetically speckled optical images and carried out with the Euclidean distance and a total variation regularization as loss function. Several subsequent deep learning works [13], [34]–[38] proposed slight variations on the topic by introducing different architectures and losses, but all under the supervised training umbrella using synthetically speckled SAR images. In [34] the authors proposed IDGAN, a deep learning SAR despeckling method based on a generative adversarial network (GAN) and trained using a weighted combination of Euclidean loss, perceptual loss and adversarial loss. In [35], a dilated densely connected network (SAR-DDCN) trained with Euclidian distance, was proposed to enlarge the receptive field and to improve feature propagation and reuse. A combination of hybrid dilated convolutions and both spatial and channel attention modules through a residual architecture called HDRANet was proposed in [36], to further improve the feature extraction capability. More recently, Cozzolino et al. [39] proposed a method that combines the classical non-local means method with the power of CNN, where NLM weights are assigned by a convolutional neural network with non local layers.

Until now, the power of CNN has not been fully exploited yet, since most of the works in literature make use of synthetic SAR images. Inspired by the recent blind-spot CNN denoising works, we tackle SAR despeckling with a self-supervised Bayesian framework relying on blind-spot CNNs.

### *B. Self-supervised denoising with CNNs*

During the last year, significant advances have been made on deep learning approaches to denoising that do not require ground-truth, showing that it is possible to reach performance close to that exhibited by fully-supervised methods. These new self-supervised denoising methods have been developed on natural images, but it is quite clear that extending them to the SAR context is appealing, as significant speckle noise is always present in SAR acquisitions. Noise2Noise [15] proposed to use pairs of images with the same content but independent noise realizations. The main drawback of this method is the difficulty of accessing multiple versions of the same scene with independently drawn noise realizations. Yuan et al. [40] presented a despeckling method based on the idea of Noise2Noise [15], but still simulating speckle on a dataset based on ImageNet. Ma et al. [41] devised a method based on the Noise2Noise scheme, requiring multi-temporal SAR images to train the network. They coped with the possible temporal variations by introducing a similarity measure in order to weight the contribution of each pixel pair in the loss.

Noise2Void [16] and Noise2Self [17] further relax the constraints on the dataset, requiring only a single noisy version of the training images, by introducing the concept of blind-spot networks. Assuming spatially uncorrelated noise, and excluding the center pixel from the receptive field of the network, the network learns to predict the value of the center pixel from its receptive field by minimizing the $\ell_2$ distance between the prediction and the noisy value. The network is prevented from learning the identity mapping because the pixel to be predicted is removed from the receptive field. Notice that this is also the reason for the uncorrelated noise assumption. The blind-spot scheme used in Noise2Void [16] is carried out by a simple masking method that hides one pixel at a time, processing the entire image to learn to reconstruct a single cleaned pixel. Laine et al. [18] devised a novel blind-spot CNN architecture capable of processing the entire image at once, increasing the efficiency. They also introduced a Bayesian framework to include noise models and priors on the conditional distribution of the blind spot given the receptive field.

## III. Background

CNN denoising methods estimate the clean image by learning a function that takes each noisy pixel and combines its value with the local neighboring pixel values (receptive field) by means of multiple convolutional layers interleaved with non-linearities. Taking this from a statistical inference perspective, a CNN is a point estimator of $p(x_i|y_i, \Omega_{y_i})$, where $x_i$ is the $i^{\text{th}}$ clean pixel, $y_i$ is the $i^{\text{th}}$ noisy pixel and $\Omega_{y_i}$ represents the receptive field composed of the noisy neighboring pixels, excluding $y_i$ itself. Noise2Void and Noise2Self predict the clean pixel $x_i$ by relying solely on the neighboring pixels and using $y_i$ as a noisy target. By doing so, the CNN learns to produce an estimate of $\mathbb{E}_{x_i}[x_i|\Omega_{y_i}]$, using the $\ell_2$ loss when in presence of Gaussian noise. The drawback of these methods is that the value of the noisy pixel $y_i$ is never used to compute the clean estimate.

The Bayesian framework devised by Laine et al. [18] explicitly introduces the noise model $p(y_i|x_i)$ and conditional pixel prior given the receptive field $p(x_i|\Omega_{y_i})$ as follows:

$$p(x_i|y_i, \Omega_{y_i}) \propto p(y_i|x_i)p(x_i|\Omega_{y_i}).$$

The role of the CNN is to predict the parameters of the chosen prior $p(x_i|\Omega_{y_i})$. The denoised pixel is then obtained as the posterior mean (MMSE estimate), i.e., it seeks to find $\mathbb{E}_{x_i}[x_i|y_i, \Omega_{y_i}]$. Under the assumption that the noise is pixel-wise i.i.d., the CNN is trained so that the data likelihood $p(y_i|\Omega_{y_i})$ for each pixel is maximized. The main difficulty involved with this technique is the definition of a suitable prior distribution that, when combined with the noise model, allows for closed-form posterior and likelihood distributions. We also remark that while imposing a handcrafted distribution as $p(x_i|\Omega_{y_i})$ may seem very limiting, it is actually not since i) that is the *conditional* distribution given the receptive field rather than the raw pixel distribution, and ii) its parameters are predicted by a powerful CNN on a pixel-by-pixel basis.

## IV. Proposed method

Following the notation in Sec. III, this section presents the Bayesian model we adopt for SAR despeckling, the training procedure and the blind-spot architecture. A summary is shown in Figs. 1 and 2.

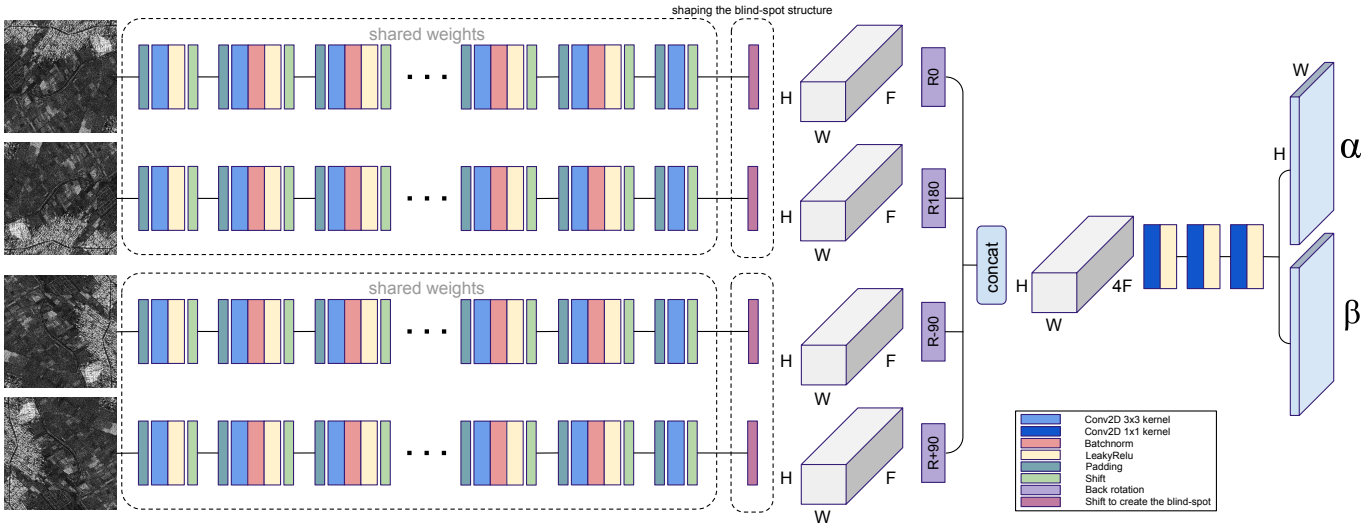


Fig. 1. Speckle2Void takes as input four rotated versions of an image. Each branch processes a specific rotation to compute the receptive field in a specific direction. Subsequently, the four half-plane receptive fields are shifted to achieve the desired blind-spot shape, rotated back and concatenated. As last, a series of 2D convolutions with kernel 1x1 are used to fuse the four receptive fields and generate the parameters of the inverse gamma for each pixel.

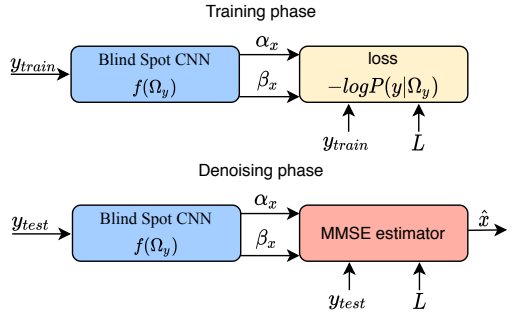


Fig. 2. Scheme depicting the training and the testing phases. During training phase the blind-spot network is trained to minimize the negative log of the noisy data likelihood to estimate $\alpha_{x_i}$ and $\beta_{x_i}$ for each pixel. In testing phase, the MMSE estimator generates the final clean image, combining together the parameters of the pixel prior, the noisy pixel and the parameter of noise distribution.

### A. Model

We consider the multiplicative SAR speckle noise model: $y_i = n_i x_i$, where $x$ represents the unobserved clean image in intensity format and $n$ the spatially uncorrelated multiplicative speckle. Concerning noise modeling, one common assumption is that it follows a Gamma distribution with unit mean and variance $1/L$ for an $L$-look image and has the following probability density function:

$$p(n) = \frac{1}{\Gamma(L)} L^L n^{L-1} e^{Ln}$$

where $\Gamma(.)$ denotes the Gamma function and $n \geq 0$, $L \geq 1$. The aim of despeckling is to estimate intensity backscatter $x$ from the observed intensity return $y$.

We model the conditional prior distribution given the receptive field as an inverse Gamma distribution with shape $\alpha_{x_i}$ and scale $\beta_{x_i}$:

$$p(x_i|\Omega_{y_i}) = \text{invGamma}(\alpha_{x_i}, \beta_{x_i}),$$

where $\alpha_{x_i}$ and $\beta_{x_i}$ depend on $\Omega_{y_i}$, since they are the outputs of the CNN at pixel $i$. Assuming the noise to be Gamma-distributed, i.e., $n_i \sim \Gamma(L, L)$, then by the scaling property of the Gamma distribution, we obtain that $y_i|x_i \sim \Gamma(L, \frac{L}{x_i})$. We can now write the unnormalized posterior distribution as:

$$\begin{aligned} p(x_i|y_i, \Omega_{y_i}) &\propto p(y_i|x_i)p(x_i|\Omega_{y_i}), \\ p(x_i|y_i, \Omega_{y_i}) &\propto \frac{1}{\Gamma(L)} \left(\frac{L}{x_i}\right)^L y_i^{L-1} e^{\frac{L}{x_i} y_i} \frac{\beta_{x_i}^{\alpha_{x_i}}}{\Gamma(\alpha_{x_i})} \frac{e^{\frac{\beta_{x_i}}{x_i}}}{x^{\alpha_{x_i}+1}}, \\ &\propto \frac{e^{\frac{Ly_i+\beta_{x_i}}{x_i}}}{x^{\alpha_{x_i}+L+1}} \end{aligned}$$

For the chosen prior and noise models, the posterior distribution has still the form of an inverse Gamma with shape $L + \alpha_{x_i}$ and scale $\beta_{x_i} + Ly_i$:

$$p(x_i|y_i, \Omega_{y_i}) = \text{invGamma}(L + \alpha_{x_i}, \beta_{x_i} + Ly_i). \quad (1)$$

The chosen prior distribution and noise model allow to conveniently obtain the marginal distribution of the noisy training data $p(y_i|\Omega_{y_i})$ in close form by solving the following integral:

$$p(y_i|\Omega_{y_i}) = \int p(y_i|x_i)p(x_i|\Omega_{y_i})dx_i \quad (2)$$

The probability density obtained by solving this integral is known as $G_I^0$, and has the following expression:

$$p(y_i|\Omega_{y_i}) = G_I^0 = \frac{L^L y_i^{L-1}}{\beta_{x_i}^{-\alpha_{x_i}} \text{Beta}(L, \alpha_{x_i})(\beta_{x_i} + Ly_i)^{L+\alpha_{x_i}}}, \quad (3)$$

According to [42], the $G_I^0$ distribution is a very general model, that is particularly suitable to model the observed intensity return $y$ of SAR images and able to accommodate different types of areas: from extremely heterogeneous scenes such as urban areas, to extremely homogeneous scenes such as deforested area as $-\alpha_{x_i}$ and $\beta_{x_i}$ become larger.

### B. Training

The training procedure learns the weights of the blind-spot CNN. The blind-spot CNN processes the noisy image to produce the estimates for parameters $\alpha_{x_i}$ and $\beta_{x_i}$ of the inverse gamma distribution $p(x_i|\Omega_{y_i})$ used as prior. It is trained to minimize the negative log likelihood $p(y_i|\Omega_{y_i})$ for each pixel, so that the estimates of $\alpha_{x_i}$ and $\beta_{x_i}$ fit the noisy observations.

As stated in Sec.II-B, training a blind-spot network requires noise to be spatially uncorrelated, so that the CNN is prevented from exploiting the latent correlation to reproduce the noise in the blind spot. While many works assume that SAR speckle is uncorrelated, the SAR acquisition and focusing system has a point spread function (PSF) that correlates the data. To cope with this, we apply a pre-processing whitening procedure, such as the one proposed by Lapini et al. [43] to decorrelate the speckle. In [43], the authors use the complex SAR data after focusing to estimate the PSF of the system and approximately invert it, achieving the desired decorrelation and showing that this step boosts the performance of any despeckling algorithm relying on the uncorrelated speckle assumption. This whitening step is especially critical in the proposed approach due to the high capacity of neural networks to overfit even random patterns.

However, perfect decorrelation is in practice impossible and the residual correlation could limit the performance of the blind-spot CNN. For this reason, we modify the basic design of the blind-spot CNN by Laine et al. [18], and introduce a variable-sized blind spot. If noise correlation cannot be removed by other means, one could consider the width of the autocorrelation function of the noise and set a blind spot that is wide enough to cover the peak of the autocorrelation. This ensures that the receptive field contains a negligible amount of information for the reproduction of the noise component of the pixel to be estimated. However, this inevitably reduces the amount of information that can be exploited by the CNN, as the content of the immediate neighbors of a pixel is the most similar to that of the pixel itself. Therefore, a larger blind spot trades off more effective noise suppression with a less accurate (appearing as blurry) prediction.

To achieve a finer control about this trade-off, we devise a regularized training procedure that allows to tune the degree of reliance of the CNN on the immediate neighbors, leading to an improvement of the high frequency details in the denoised image, while still suppressing most of the noise correlation. During training, we randomly alternate, with predefinied probabilities, a $1 \times 1$ blind spot and a larger blind spot that can have arbitrary shape to match the noise autocorrelation. This mechanism allows the network weights to learn how to partially exploit the neighboring pixels belonging to the larger blind-spot but at the same time not to rely too much on them, in order to prevent from overfitting the noise components. During testing, a $1 \times 1$ blind spot is used, thus only excluding the center pixel, and exploiting the closest neighbors. Due to their weak training, these neighbors allow to recover some high frequency image content, which is the stronger signal present, while not being able to exploit the weaker correlations in the noise. We refer the reader to Sec. V-D for the details on the chosen parameter settings and the specific SAR dataset used for training.

### C. Testing

In testing, the blind-spot CNN processes the noisy SAR image to estimate $\alpha_{x_i}$ and $\beta_{x_i}$ for each pixel. The despeckled image is then obtained through the MMSE estimator, i.e., the expected value of the posterior distribution in Eq. (1), as:

$$\hat{x}_i = \mathbb{E}[x_i|y_i, \Omega_{y_i}] = \frac{\beta_{x_i} + Ly_i}{L + \alpha_{x_i} - 1}.$$

Notice that this estimator combines both the per-pixel prior estimated by the CNN and the noisy observation.

### D. Loss function

As mentioned in Sec. IV-B, the blind-spot CNN is trained by minimizing the negative log likelihood of the noisy observations, based on the estimated parameters $\alpha_{x_i}$ and $\beta_{x_i}$ of the prior. Moreover, we incorporate a total variation (TV) component, computed over the posterior image, to further promote smoothness. Our final loss function is as follows:

$$l = -\sum_i \log p(y_i|\Omega_{y_i}) + \lambda_{TV} TV(\hat{x})$$

where $p(y_i|\Omega_{y_i})$ is defined in Eq. (3), the TV term is the anisotropic version of the total variation $TV(\hat{x}) = \sum_{i,j} |\hat{x}_{i+1,j} - \hat{x}_{i,j}| + |\hat{x}_{i,j+1} - \hat{x}_{i,j}|$ and $\lambda_{TV}$ is a hyperparameter to tune the desired degree of smoothness.

### E. Blind-spot architecture

The rationale behind the blind-spot network is to introduce a pixel-sized hole in the receptive field, in order to prevent the network from learning the identity mapping. Our model is built upon the architecture by Laine et al. [18], who designed a CNN architecture to naturally account for the blind spot in the receptive field, thus increasing training efficiency. They cleverly implemented shift and padding operations on the feature maps at each layer, in order to limit the receptive field to grow in a specific direction, excluding the center pixel from the computation. Their architecture is composed of four different CNNs, each responsible of limiting the receptive field to extend in a single direction by means of shift and padding operations on the feature maps at each layer. The four subnetworks produce four limited receptive fields that extend strictly above, below, leftward and rightward of the target pixel. In order to reduce the number of trainable parameters, they feed four rotated versions of each input image to a single network that computes the receptive field in a specific direction. The four limited receptive fields are finally combined through a series of 2D convolutions with $1 \times 1$ filters, ensuring no further expansion of the receptive field. To perform this particular computation, classical 2D convolutional layers are used but their receptive field is limited to grow in a direction by shifting the feature map in the opposite direction by an offset of $\lfloor k/2 \rfloor$ pixels, where $k \times k$ is the kernel size, before performing the convolution operation. At the end of the network, each of the four limited receptive fields still contains

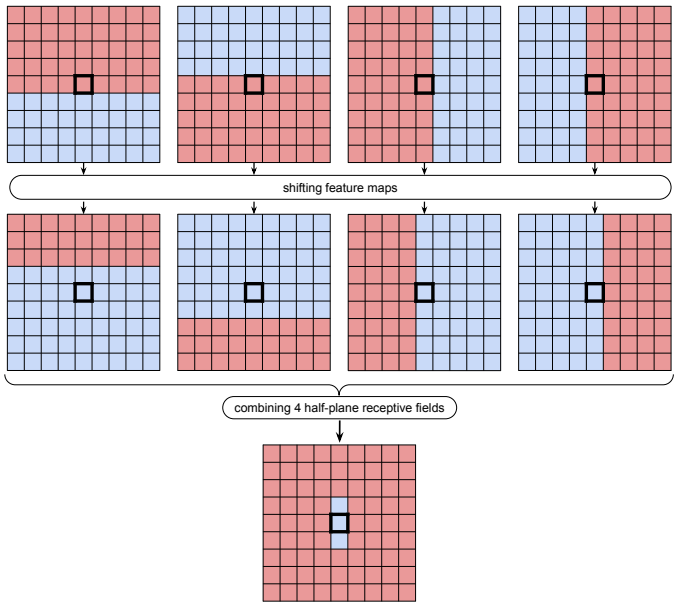


Fig. 3. Visual depiction of the operations performed by the blind-spot network to constrain the receptive field related to the center pixel to exclude the center pixel itself and two pixels in the vertical direction. The first row represents, in pink color, the four limited receptive fields extending in four directions. As the center pixel is still included in the receptive fields, each feature map is shifted in the opposite direction with respect to the growing direction of the receptive field. This shifting operation allows the pink pixels in the second row to be the new receptive fields associated to the center pixel. The shift is 1 in azimuth direction and 2 in the range one. The last row represents the final receptive field, related to the center pixel, as the result of a combination of the four receptive fields depicted in the second row.

the center row/column, so the center pixel as well. To exclude it, the feature maps are shifted by one pixel before combining them.

An overview of the blind-spot network used by Speckle2Void is shown in Fig. 1. Speckle2Void modifies the basic architecture by Laine et al. [18] described above to allow more flexibility in shaping the blind-spot. In principle, if the final shift applied to each of the four directional receptive fields was different from one another, we would be able to control the size of the blind spot in each direction. In SAR images, the azimuth and range directions may exhibit different statistical properties, including the residual noise autocorrelation. We therefore account for that by only sharing weights between the two branches processing the receptive field oriented as the azimuth or range directions, instead of sharing them for all four branches as in [18]. Furthermore, as shown in Fig. 3, Speckle2Void can apply one shift in the azimuth direction and a different shift in the range one.

### *F. Non local convolutional layer and its adaptation to blind-spot networks*

The blind-spot CNN used by Speckle2Void also comes in two versions. The "local" version of Speckle2Void is composed by a series of classic 2D convolutional layers, each followed by Batch normalization [44] and a Leaky-ReLU non-linearity. The "non-local" version adds several non-local layers, as defined in [45]. Such layers introduce a dynamic weighted function of the feature vectors that help retrieving more information from a wider image context. While the "local" version of Speckle2Void employs classical 2D convolutions, so only local information is exploited at each layer, non-local layers leverage non-local structural similarity across spatially distant patches within an image, enabling the CNN to combine both spatially-neighboring as well as distant pixels. In particular, non-local self-similarity can be effective in recovering the information hidden by the blind spot, without encountering the problem of noise correlation as it is drawn from spatially-distant areas. However, exploiting non-locality incurs a significant penalty in terms of computational cost.

The non-local module proposed by NLRN [45] uses a soft block matching approach and applies the Euclidean distance with linearly embedded Gaussian kernel as distance metric. The rational behind this module is to perform a weighted combination of all the feature vectors in a patch (search window) to compute the new feature vector at its center, where the used weights dynamically depend on the similarity between the center feature vector and all the others within the patch, and repeat it for each feature vector in the feature map. This non-local layer is designed to work in a traditional CNN architecture, and requires introducing a masking technique to adapt it to the blind-spot architecture used by Speckle2Void. In [45], the linear embeddings are defined as follows:

$$\Phi(X_{ij}) = \phi(X_{ij}, X_{p_{ij}}) = \exp\{\theta(X_{ij})\psi(X_{p_{ij}}))\}, \forall i, j,$$
$$\theta(X_{ij}) = X_{ij}W_\theta, \psi(X_{p_{ij}}) = X_{p_{ij}}W_\psi, G(X_{ij}) = X_{p_{ij}}W_g, \forall i, j.$$

$\Phi(X_{ij})$ represents the distance metric to encode the non local correlation between the feature vector in position $i, j$ and each neighbours in the patch $X_{p_{ij}}$. $\Phi(X_{ij})$ has shape $1 \times q \times q$ where $q \times q$ denotes the spatial size of the neighbour patch centered at pixel $i, j$. $\theta(X_{ij})$ represents the embedding associated to the feature vector in position $i, j$ with shape $1 \times l$ where $l$ is the number of features. $\psi(X_{p_{ij}})$ represents the embeddings associated to each feature vector in the neighbour patch $p$ centered at $i, j$ with shape $q \times q \times m$ where $m$ is the number of features. The transformation weights $W_\theta, W_\psi, W_g$ used to compute the embeddings have shape $m \times l$, $m \times l$, $m \times m$ respectively, and are trainable weights. We add a masking operation to the non-local layer proposed in [45] and the final formulation is obtained as:

$$Z_{ij} = \frac{1}{\delta'(X_{ij})}(M_i \odot \exp\{X_{ij}W_\theta W_\psi^T X_{p_{ij}}^T)\})X_{p_{ij}}W_g, \forall i, j,$$

where $\delta'(X_{ij}) = \sum_{p_{ij}} M_i \odot \phi(X_{ij}, X_{p_{ij}})$ is the normalization factor, $Z_{ij}$ is the output feature vector at spatial location $i, j$ and $M_i$ is a mask, associated to row $i$, aiming to get rid of the contribution of specific feature vectors in the computation of the new feature vector $Z_{ij}$. Considering the receptive field extending upwards, all the pixels in a specific row $i$ are associated with a mask $M_i$ which has weight 1 in row $i$ and all the rows above, and 0 everywhere else. This allows to disregard all Euclidian distances with respect to feature vectors that are not contained in the receptive field extending upwards. The construction of the mask $M_i$ is not influenced by the shape of the blind-spot structure. The blind-spot shaping always happens right after the four receptive fields are computed, by

shifting each of the four feature maps according to the desired final shape, as in the "local" version.

## V. Experimental results and discussions

In this section, we evaluate the performance of Speckle2Void, both quantitatively and qualitatively. First, we compare our method with several state-of-the-art methods on a synthetic dataset, where the availability of ground truth images allows to compute objective performance metrics, and then on a real-world SAR dataset, relying on several established no-reference performance metrics and visual results. We also test the proposed method against a benchmarking dataset, composed of a set of simulated canonical images, to highlights its behavior in all the major types of regions found in SAR images. Moreover, we perform an ablation study to show the impact of various design choices on the despeckling performance.Finally, a comparison on the computational time is provided to assess the different complexity of CNN-based methods with respect to the traditional methods.

Speckle2Void code is publicly available online: https://github.com/diegovalsesia/speckle2void.

### A. Quality assessment criteria

The evaluation reference metric used to assess quantitative results on synthetic SAR images corrupted by simulated speckle is the PSNR. This allows to understand the denoising capability of our self-supervised method when compared with traditional methods and CNN-based ones with supervised training. In the second set of experiments, conducted on real SAR images, we compare the various despeckling methods by relying on some no-reference performance metrics such as equivalent number of looks (ENL), moments of the ratio image ($\mu_r$, $\sigma_r$), quality index $\mathcal{M}$ [46] and the ratio image structuredness RIS [47]. The ENL is estimated over apparently homogeneous areas in the image and is defined as the ratio of the squared average intensity to the variance. Computing the ENL on the noisy SAR image provides an approximate estimate of its nominal number of looks. Moments of the ratio image $\mu_r$ and $\sigma_r$ measure how close the obtained ratio image is to the statistics of pure speckle ($\mu_r = 1$, $\sigma_r = 1$ are desirable for a single-look image). The previous metrics lack in conveying information about the detail preservation capability of a filter and the visual inspection of the ratio image would provide an indication of the remaining structure of what ideally should be pure speckle with no visible pattern. To avoid the subjectiveness of the visual interpretation of ratio images, Gomez et al. [46] designed the quality index $\mathcal{M}$. This index evaluates the goodness of a filter by integrating two measures together: a first-order component measuring the deviation from ideal ENL and from ideal speckle mean over $n$ automatically selected textureless areas and a second-order component measuring the remaining geometrical content within the ratio image through the homogeneity textural descriptor proposed by Haralick et al. [48]. Ideally, $\mathcal{M}$ should tend to zero. RIS [47] is a metric closely related to the second-order component of $\mathcal{M}$, allowing to evaluate solely the remaining geometrical content within the ratio image. Similarly to Gomez et al. [46], it employes the homogeneity textural descriptor proposed by Haralick et al. [48] to measure the similarity among neighbouring pixels. RIS is zero when the ratio image consists of independent identically distributed speckle samples.

### B. Reference methods

The following state-of-the-art references are compared with our method on both optical and SAR datasets:

1) PPB [31];
2) SAR-BM3D [7];
3) NL-SAR [32];
4) CNN baseline with the improved loss defined in [14];
5) ID-CNN [12].

These methods have been chosen for their popularity and diffusion in the SAR community. For PPB [31], SAR-BM3D [7] and NL-SAR [32] methods, we selected parameters as suggested in the original papers. As a CNN baseline we used the well-known network architecture proposed in [33], employing a homomorphic approach and the loss proposed in [14] that better adapts to deal with the speckle noise distribution. ID-CNN has been implemented from scratch following the indications in the original paper for what concerns the CNN architecture and the hyperparameters. Notice that both CNNs follow a supervised training approach with synthetically speckled natural images. We remark that we do not directly compare with the results in SAR-CNN [14] or the more recent work in [39] as they use multitemporal data, which would make the setting unfair with respect to the single observation of a scene in our case. In addition, the dataset used in those works is not publicly available.

As described in Sec. IV, Speckle2Void employs four branches where the horizontal and the vertical directions are processed separately with a different set of parameters, as shown in Fig. 1. The first part of the architecture consists of 17 blocks composed of 2D convolution with $3\times3$ kernels with 64 filters each, batch normalization and Leaky ReLU nonlinearity. After that, the branches are merged with a series of three $1\times1$ convolutions. The non-local version of our method maintains the same general structure with an addition of 5 non-local layers, one every 3 local layers. The same architecture is used in both the experiments with the only difference that in the case of synthetic images the blind-spot shape is $1 \times 1$, since the injected speckle is pixel-wise i.i.d and therefore there is no need to use an enlarged blind-spot. Instead, in the real SAR case the blind-spot shape is variable across training.

For both experiments, the Adam optimization algorithm [49] is employed, with momentum parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. We use the Tensorflow framework to train the proposed network on a PC with 64 GB RAM, an AMD Threadripper 1920X, and an Nvidia 1080Ti GPU.

### C. Synthetic dataset

In this experiment we use natural images to construct a synthetic SAR-like dataset. Pairs of noisy and clean images

TABLE I
SYNTHETIC IMAGES - PSNR (DB)

| **Image** | **PPB** [31] | **SAR-BM3D** [7] | **NL-SAR** [32] | **Baseline CNN** | **ID-CNN** [12] | **Speckle2Void** | **Speckle2Void+TV** | **Speckle2Void+NL** |
|---|---|---|---|---|---|---|---|---|
| Cameraman | 23.02 | 24.76 | 24.37 | **26.26** | 25.83 | 25.90 | 25.90 | 25.85 |
| House | 25.51 | 27.55 | 25.75 | 28.17 | **28.32** | 27.96 | 27.94 | 28.08 |
| Peppers | 23.85 | 24.92 | 23.62 | **26.30** | 26.26 | 25.99 | 26.02 | 26.09 |
| Starfish | 21.13 | 22.71 | 21.84 | 23.39 | 23.42 | 23.32 | 23.31 | **23.50** |
| Butterfly | 22.76 | 24.48 | 23.82 | 25.96 | **26.09** | 25.82 | 25.80 | 25.98 |
| Airplane | 21.22 | 22.71 | 21.83 | 23.78 | **23.90** | 23.67 | 23.65 | 23.61 |
| Parrot | 21.88 | 24.17 | 24.13 | **25.91** | 25.85 | 25.44 | 25.45 | 25.46 |
| Lena | 26.64 | 27.85 | 26.80 | 28.66 | **28.71** | 28.54 | 28.58 | 28.44 |
| Barbara | 24.08 | **25.37** | 23.13 | 24.30 | 24.38 | 24.36 | 24.31 | 24.74 |
| Boat | 24.22 | 25.43 | 24.55 | **26.06** | 26.00 | 26.02 | 25.57 | 25.88 |
| *Average* | *23.43* | *24.99* | *23.98* | ***25.88*** | ***25.88*** | *25.70* | *25.69* | *25.76* |

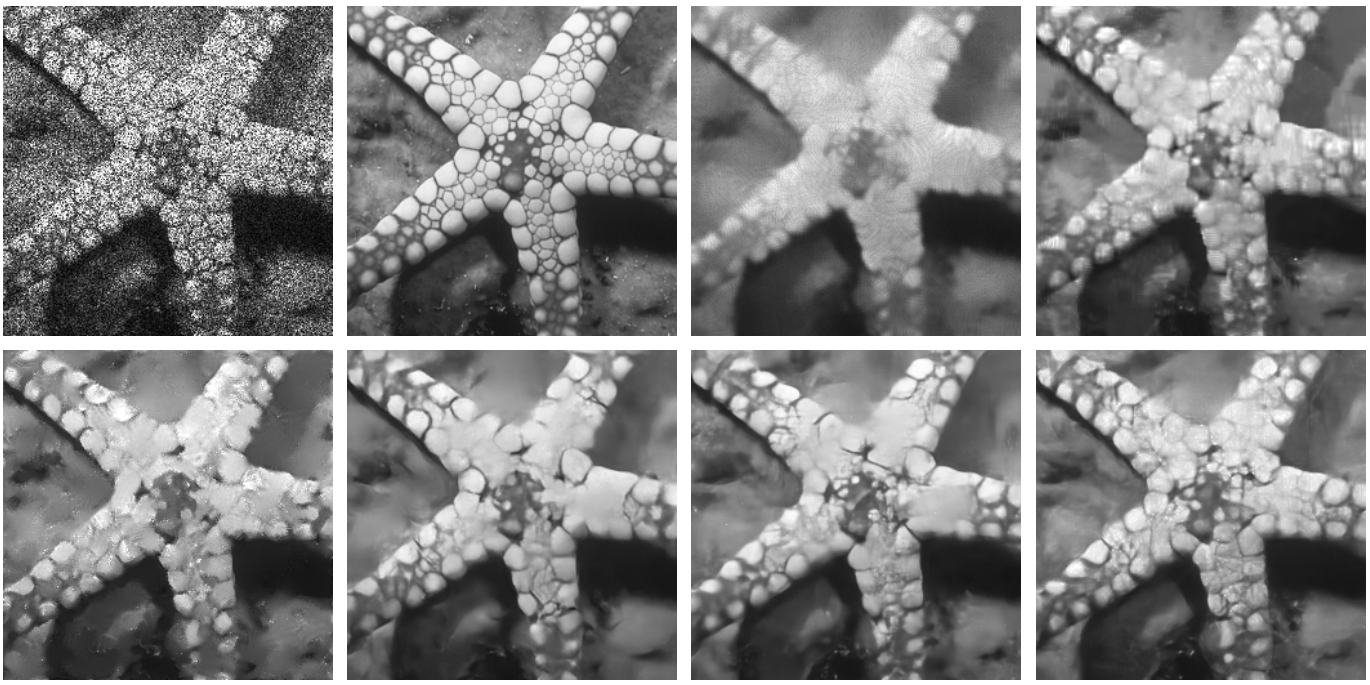

Fig. 4. Synthetic images: Noisy, Clean, PPB (21.13 dB), SAR-BM3D (22.71 dB), NL-SAR (21.89 dB), CNN-based baseline (23.37 dB), ID-CNN (23.42 dB), synthetic Speckle2Void (23.32 dB).

are built by generating i.i.d. speckle to simulate a single-look intensity image ($L = 1$).

During training, patches are extracted from 450 different images of the Berkeley Segmentation Dataset (BSD) [50]. The network has been trained for about 400 epochs with a batch size of 16 and learning rate equal to $10^{-5}$. All the CNN-based methods have been trained with the same synthetic dataset. Table I shows performance results on a set of well-known testing images in terms of PSNR. It can be seen that all the CNN-based methods outperform the non-local traditional methods by a significant margin. Despite ID-CNN employs the suboptimal $\ell_2$ loss, the TV regularizer helps smoothing out the artifacts, showing approximately the same result as the CNN baseline. It can be noticed that our self-supervised method outperforms PPB, SAR-BM3D and NL-SAR. Moreover, it is interesting to notice that while the proposed approach does not use the clean data for training, it achieves comparable results with respect to the supervised ID-CNN and CNN-based baseline methods. This happens for the non-local version and TV version as well. We can observe a slight improvement when non-locality is employed. Even if we analyze the performance from a qualitative perspective, as done in Fig. 4, we observe the same behaviour. Despite the absence of the true clean images during training, our method produces images as visually pleasing as those produced by the CNN-based reference approaches with comparable edge-preservation capabilities. This is a significant result because it shows that, in theory, we do not need supervised training to achieve the outstanding despeckling results obtained by CNN-based methods.

### *D. TerraSAR-X dataset*

In this experiment we employ single-look TerraSAR-X images[1]. Notice that optimal results are obtained by training a model that is specific to a given SAR platform (e.g., TerraSAR-X in our example). We suggest retraining from random initialization to optimize the model for a different platform. This should not be an issue since we only require

[1]https://tpm-ds.eo.esa.int/oads/access/collection/TerraSAR-X/tree

TABLE II
PERFORMANCE METRICS ON 5 REAL TERRASAR-X TEST IMAGES

| Metric | Image | PPB [31] | SAR-BM3D [7] | NL-SAR [32] | CNN baseline | ID-CNN [12] | Speckle2Void | Speckle2Void NL |
|---|---|---|---|---|---|---|---|---|
| ENL ↑ | 1 | 82 | 46.2 | 77.3 | 52.9 | 76.5 | **88.5** | 86.5 |
| | 2 | 78.6 | 49.1 | 60.6 | 48.7 | 69.9 | **89.9** | 81.8 |
| | 3 | 76.9 | 58.1 | 59.4 | 52.5 | 73.1 | 84.0 | **86.0** |
| | 4 | 54.2 | 40.4 | 45.0 | 37.6 | 46.2 | **54.7** | 53.1 |
| | 5 | **22.9** | 16.2 | 16.8 | 14.6 | 16.6 | 18.9 | 17.5 |
| $\mu_r$ ↑ | 1 | 0.887 | 0.919 | 0.921 | 0.963 | 0.943 | 0.966 | **0.970** |
| | 2 | 0.925 | 0.938 | 0.933 | **0.969** | 0.964 | 0.966 | 0.967 |
| | 3 | 0.926 | 0.941 | 0.936 | **0.974** | 0.969 | 0.968 | 0.968 |
| | 4 | 0.933 | 0.942 | 0.936 | 0.974 | 0.976 | 0.962 | **0.977** |
| | 5 | 0.853 | 0.894 | 0.902 | **0.950** | 0.918 | 0.947 | 0.946 |
| $\sigma_r$ ↑ | 1 | **0.847** | 0.627 | 0.692 | 0.726 | 0.745 | 0.803 | 0.800 |
| | 2 | **0.886** | 0.674 | 0.734 | 0.740 | 0.803 | 0.829 | 0.817 |
| | 3 | **0.874** | 0.684 | 0.739 | 0.756 | 0.817 | 0.816 | 0.814 |
| | 4 | **0.876** | 0.688 | 0.746 | 0.755 | 0.846 | 0.823 | 0.837 |
| | 5 | **0.891** | 0.549 | 0.621 | 0.683 | 0.664 | 0.748 | 0.736 |
| $\mathcal{M}$ [46] ↓ | 1 | 24.4 | 16.5 | 13.8 | 11.9 | 14.6 | 7.72 | **6.71** |
| | 2 | 10.1 | 11.6 | 15.4 | 11.6 | 9.12 | 9.11 | **8.04** |
| | 3 | 9.82 | 11.3 | 13.0 | 11.3 | 6.93 | 6.24 | **5.44** |
| | 4 | 10.6 | 10.5 | 16.9 | 12.3 | 9.7 | 8.07 | **7.74** |
| | 5 | 14.4 | 14.3 | 11.7 | 9.76 | 10.4 | 8.91 | **7.9** |
| RIS [47] ↓ | 1 | 0.402 | 0.186 | 0.098 | 0.145 | 0.242 | 0.0929 | **0.0817** |
| | 2 | 0.114 | 0.0765 | 0.111 | 0.0925 | 0.112 | 0.0918 | **0.075** |
| | 3 | 0.114 | 0.0782 | 0.076 | 0.113 | 0.0643 | 0.0396 | **0.0257** |
| | 4 | 0.0962 | **0.0392** | 0.129 | 0.127 | 0.106 | 0.0873 | 0.0804 |
| | 5 | 0.159 | 0.114 | 0.0643 | 0.0566 | 0.130 | 0.0708 | **0.0547** |

a modest number of noisy images and we also do not need careful curation of multitemporal data.

As mentioned in Sec. IV-B, both training and testing images are pre-processed through the blind speckle decorrelator in [43] to whiten them. To ensure fairness, the whitening procedure is applied to the images for all the tested methods.

During training, $64 \times 64$ patches are extracted from 30000 whitened SAR images of size $256 \times 256$. The network has been trained for 300000 iterations with a batch size of 16 and an initial learning rate of $10^{-4}$ multiplied by 0.1 at 150000 iterations. In this case, in addition to two versions (L/NL) of the proposed method used for the synthetic images, we add the TV regularizer to the loss with a $\lambda_{TV}$ of $5 \times 10^{-5}$ and we apply the regularized training procedure described in Sec. IV-B, carefully choosing the blind-spot shape. By empirical observation we found non-negligible residual noise correlation in the vertical direction after the whitening stage, so we adapted the structure of the blind spot accordingly. The regularized training alternates between a $3 \times 1$ and $1 \times 1$ shape with probabilities 0.9 and 0.1, respectively. This allows us to take into account the wider vertical autocorrelation of the speckle. In the ablation study presented in Sec. V-F1 we also show the results obtained when only a $1 \times 1$ blind spot is used.

Table II and Figs. 5,6,7 show the results obtained on a set of $1000 \times 1000$ test images[2], that were not included in the training set. Speckle2Void outperforms all other methods for almost all testing images in terms of ENL, showing a better speckle suppression capability on smooth areas. The non local version of Speckle2Void scores a slightly lower ENL with respect to the local version as it recovers finer details, generating an additional texture over the apparently homogeneous areas as shown in Fig. 6. The metric $\mu_r$ is very close to the desired statistic of the ratio image for all the considered methods, in particular for the CNN-based ones. The reference method PPB [31] provides the best result in terms of $\sigma_r$ showing a strong speckle suppression, but a very poor detail preservation capability as confirmed by the qualitative comparison in Figs. 6 and 7. Despite SAR-BM3D [7] provides worse results in terms of $\sigma_r$ with respect to PPB [31], it produces images with higher fidelity and finer details, as can be observed both visually in Fig. 5 and quantitatively with the RIS [47]. However, several areas in the SAR-BM3D image still present artifacts like streaks or unrealistic texture. NL-SAR [32] shows a stronger speckle suppression than SAR-BM3D [7], providing better results in terms of ENL and $\sigma_r$.

Overall, the CNN-based methods show a greater speckle suppression than SARBM3D [7] and PPB [31]. However, both the CNN baseline and ID-CNN [12] tend to oversmooth and produce cartoon-like edges. The test image in Fig. 5 presents strong artifacts, making the recovered details look quite unrealistic. This is due to the domain gap between natural images and real SAR images and it represents a strong argument against supervised training with synthetically speckled images. On the contrary, Speckle2Void does not hallucinate artifacts over homogeneous regions and produces higher quality images with respect to any other reference method, with much more realistic details in regions with man-made structures and sharp edges. This is confirmed qualitatively by a visual inspection of the cleaned image in Fig. 5, 6, 7. Instead, Fig. 8 shows the image obtained as the ratio between the noisy and despeckled images. Ideally, no structure should be evident in the ratio image. Also in this case, we can observe the capability of Speckle2Void to remove the speckle effectively, with a minimal amount of visible patterns. The outstanding visual quality of Speckle2Void demonstrates the effectiveness of both direct

[2]High-resolution visualization: https://diegovalsesia.github.io/speckle2void

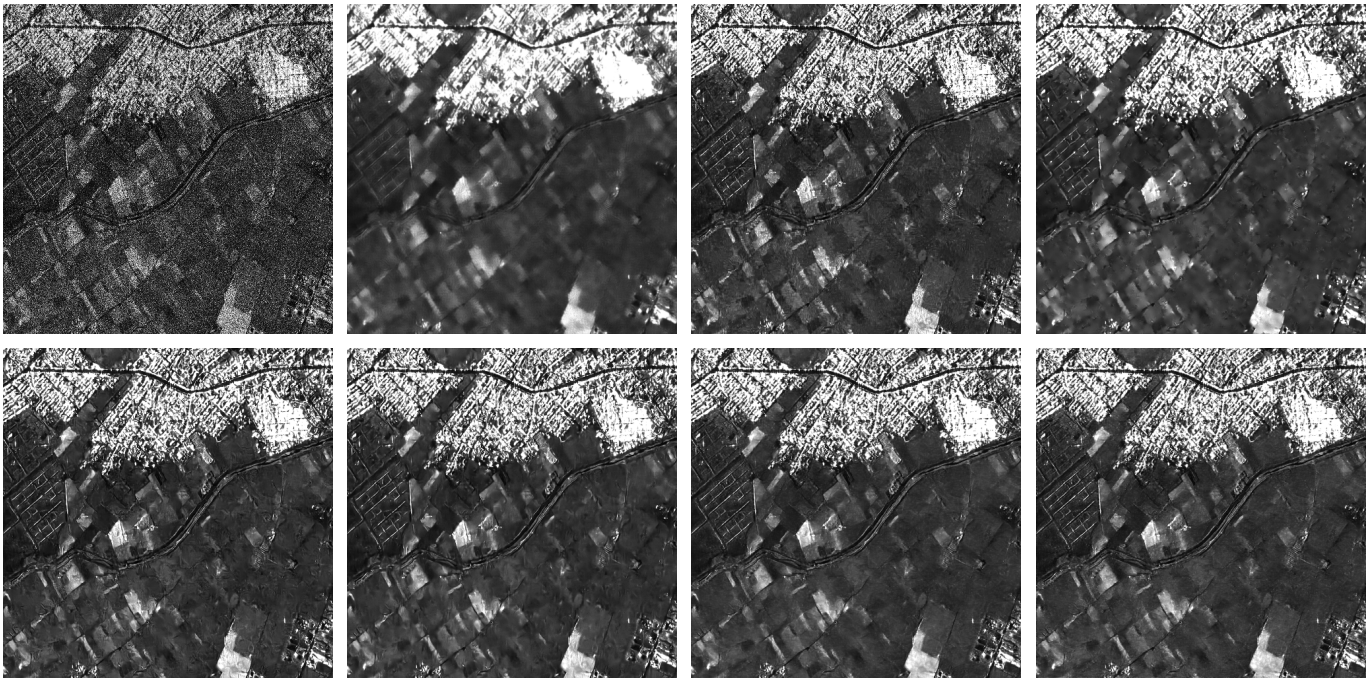

Fig. 5. TerraSAR-X image 1. Top-Left to bottom-right: Noisy, PPB, SARBM3D, NL-SAR, CNN-based baseline, ID-CNN, Speckle2Void, Speckle2Void+NL

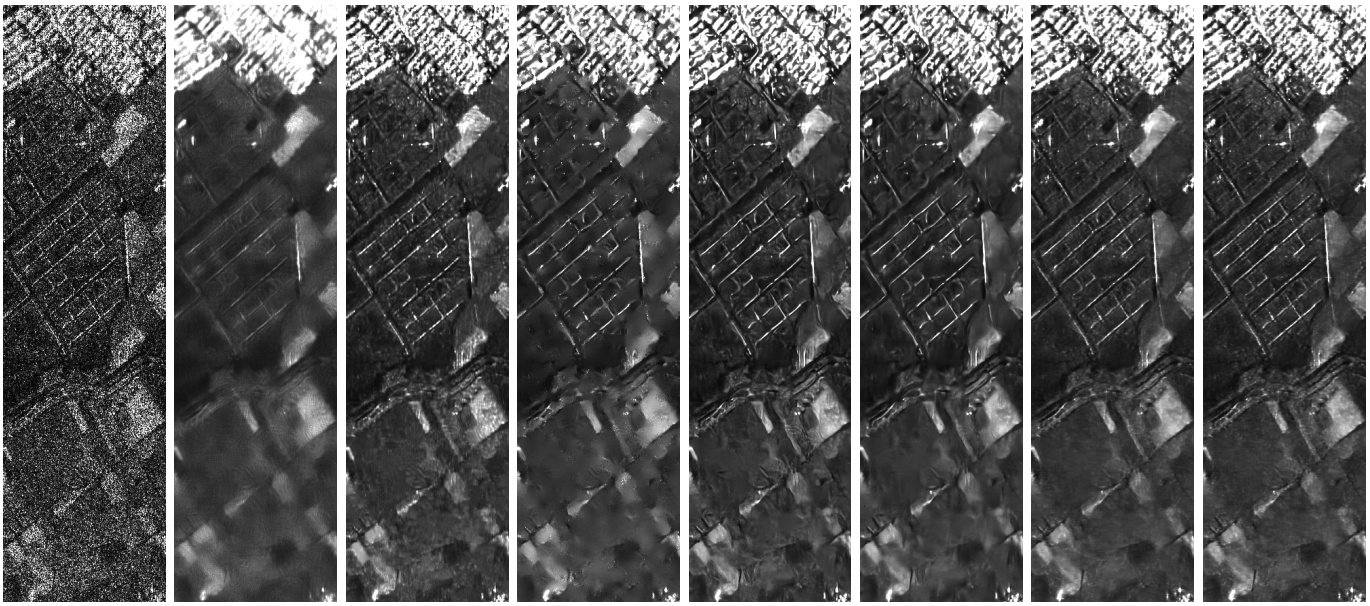

Fig. 6. TerraSAR-X image 1 detail. From left to right: Noisy, PPB, SARBM3D, NL-SAR, CNN-based baseline, ID-CNN, Speckle2Void, Speckle2Void+NL

training on real SAR images and of the adopted regularized training procedure to tackle the residual local noise correlation structure.

Moreover, if we compare the two versions of the proposed method, we can notice that adding the non-local layers provides a marginal improvement in the preservation of the details, yielding lower values for $\mathcal{M}$ [46] and RIS [47]. The drawback of the non local version of Speckle2Void is its higher computational overhead, leading to a much longer training and inference time.

### *E. Benchmarking dataset*

The presented quantitative assessment relies on no-reference metrics as the lack of clean images prevents from using full-reference measures. In [51] the authors introduce a standard benchmark for the objective assessment of SAR despeckling techniques. The use of this framework enriches our quantitative assessment on no-reference metrics by evaluating the behaviour of the compared methods on a set of canonical scenes, generated through physical SAR simulation. Five different scenes have been simulated to assess specific features of the despeckling methods:

- homogeneous scene (water, bare soils, and vegetated

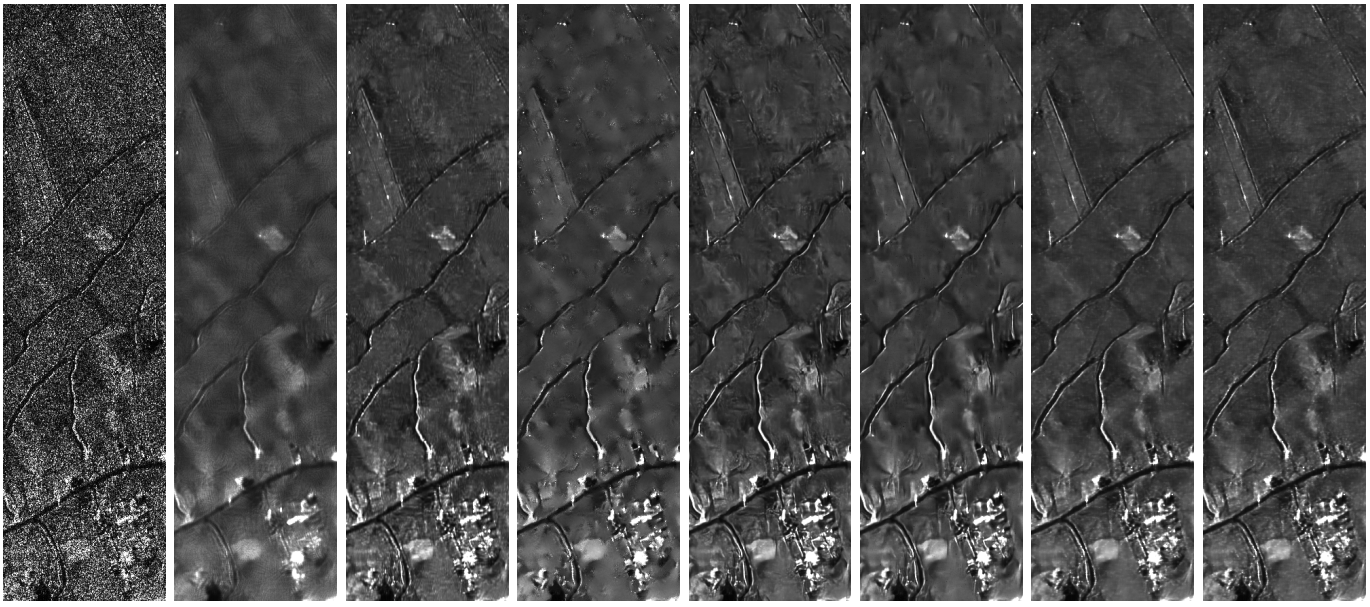

Fig. 7. TerraSAR-X image 2 detail. From left to right: Noisy, PPB, SARBM3D, NL-SAR, CNN-based baseline, ID-CNN, Speckle2Void, Speckle2Void+NL

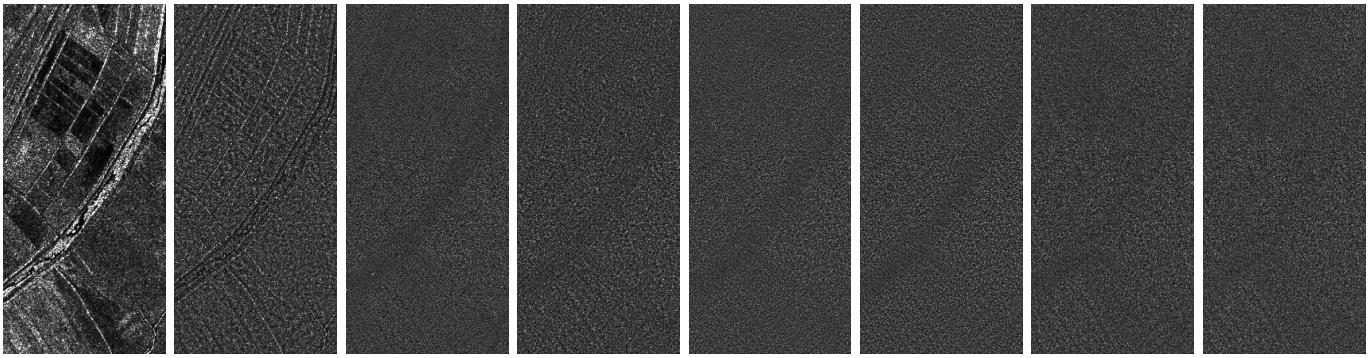

Fig. 8. TerraSAR-X image 4 detail. From left to right: Noisy and ratio images (PPB, SARBM3D, NL-SAR, CNN-based baseline, ID-CNN, Speckle2Void, Speckle2Void+NL)

areas) to focus on speckle suppression ability;
- texture scene to specifically evaluate the scene feature preservation on a nonflat terrain;
- scene with edges (roads, rivers, and region boundaries) to evaluate the preservation of contours;
- scene with isolated point target to assess the amount of radiometric distortion;
- scene with urban areas to assess the preservation of man-made structures.

In [51] the authors also propose to use a set of reference and no-reference measures associated to each test image. Table III shows that the proposed methods achieve comparable results for most of the test images and in some cases outperform the other methods. We remark that Speckle2Void is optimized on the real TerraSAR-X dataset, which present different statistics with respect to the simulated SAR images considered in the benchmark, such as a different residual noise correlation.This leads us to believe that the despeckling action of the proposed method is actually slightly sub-optimal when evaluated on the simulated SAR test images rather than on TerraSAR-X images.

*1) Homogeneous case:* This test case represents a flat surface. The performance is evaluated using the following metrics: the mean value of the filtered image (MoI), that should be preserved after despeckling; the mean and the variance of the ratio image (MoR and VoR) that should match the pure speckle statistics; the ENL and the despeckling gain (DG), which measure the speckle reduction factor on a logarithmic scale by exploiting the available clean reference. All the compared methods do not introduce any notable distortion on the mean. However, the two version of Speckle2Void present the mean indicators that are overall the closest to 1. In addition, the VoR indicates that the proposed methods are the ones that more strongly suppress speckle. The DG metric shows comparable performance for all the compared methods. The latter measure is slightly biased by the fact that the reference image is not really clean.

*2) Texture case (Digital Elevation Model):* The texture image represents an artificial canonical fractal DEM. The performance is evaluated measuring MoI, MoR, VoR, DG and the coefficient of variation $C_{\hat{x}}$, i.e., the ratio between the estimated standard deviation and the mean of the image. The latter metric measures the texture preservation. The two means show slightly worse performance for the proposed methods with respect to the references, denoting a slight radiometric

TABLE III
MEASURES FOR SIMULATED SAR TEST IMAGES

| Image | Metric | PPB [31] | SAR-BM3D [7] | NL-SAR [32] | CNN baseline | ID-CNN [12] | Speckle2Void | Speckle2Void NL |
|---|---|---|---|---|---|---|---|---|
| Homogeneous | MoI ↑ | 0.997 | 0.978 | **1.000** | 0.991 | 0.978 | 0.987 | 0.988 |
| | MoR ↑ | 0.960 | 0.979 | 0.972 | 0.979 | **0.995** | 1.01 | 0.989 |
| | VoR ↑ | 0.820 | 0.814 | 0.837 | 0.844 | **0.903** | 0.898 | 0.88 |
| | ENL ↑ | 127.68 | 102.44 | 104.52 | **125.69** | 122.94 | 120.48 | 112.96 |
| | DG ↑ | 20.29 | 19.40 | 19.46 | **20.2** | 20.04 | 20.03 | 19.8 |
| Texture | MoI ↑ | **0.998** | 0.968 | 0.915 | 0.931 | 0.836 | 0.867 | 0.846 |
| | MoR ↑ | **0.911** | 0.833 | 0.857 | 0.807 | 0.893 | 0.847 | 0.808 |
| | VoR ↑ | 0.560 | 0.415 | 0.485 | 0.475 | 0.766 | **0.848** | 0.822 |
| | $C_x$ (2.40) | 2.71 | **2.43** | 2.31 | 2.25 | 2.29 | 2.24 | 2.21 |
| | DG ↑ | 3.68 | **5.32** | 4.83 | 4.25 | 3.77 | 3.5 | 3.45 |
| Squares | ES (up) ↓ | 0.07 | 0.036 | 0.07 | **0.026** | 0.033 | 0.057 | 0.058 |
| | ES (down) ↓ | 0.209 | 0.113 | 0.198 | **0.0825** | 0.0873 | 0.138 | 0.158 |
| | FOM ↑ | 0.837 | **0.847** | 0.799 | 0.818 | 0.82 | 0.825 | 0.834 |
| Corner | $C_{NN}$ ↑ | 3.75 | 7.39 | 5.67 | **7.8** | 7.77 | 7.79 | 7.79 |
| | $C_{BG}$ ↑ | 32.69 | 35.45 | 33.75 | 36.53 | 36.51 | **36.55** | 36.54 |
| Building | $C_{DR}$ ↑ | 64.90 | 65.91 | 64.44 | 65.92 | **65.98** | 65.91 | 65.9 |
| | BS ↓ | 3.13 | 1.46 | 6.827 | 0.3082 | **0.2612** | 0.272 | 0.4031 |

distortion. All the reference techniques present a small value of VoR, showing the challenge of speckle removal in case of a highly textured image. The VoR values of the two proposed methods are the closest to 1. The coefficient of variation $C_x$ should match the theoretical one computed on the clean image, which corresponds to 2.40. The two versions of Speckle2Void present a comparable $C_{\hat{x}}$ with respect to the other CNN-based methods. DG shows similar results for all the compared methods, showing a good speckle suppression even for this challenging image.

*3) Edges (Squares):* This test case represents a flat surface divided in 4 regions with different intensity levels, creating straight contours aligned to the range and azimuth coordinates as shown in Fig. 9. The performance is evaluated through the measure of edge smearing (ES), which gives an indication of the edge degradation and the smoothing action applied by the despeckling methods, and an indirect measure called Pratt's FOM, which quantifies the ability of an automatic edge detection algorithm to recognize the edges in the clean estimate. Table III reports the ES measures for the two vertical edges, characterized by lower (up) and higher (down) contrast, along with the FOM for the detected edges. Lower ES values indicate less smearing. The worst results comes from the methods producing the blurriest edges such as PPB [31] and NL-SAR [32]. However, this metric does not give a complete insight about the edge preservation and it is quite unreliable. FOM represents the best measure to evaluate edge preservation by quantifying their recognition through a detector algorithm. The FOM values in Table III should be higher than the FOM resulting from the noisy image (0.792) and as close as possible to the one resulting from the clean reference image (0.993). The two proposed methods present FOM values that are higher than the ones produced by the supervised CNN-based methods and consistent with the best results, provided by PPB [31] and SAR-BM3D [7].

*4) Isolated point target case (Corner):* The corner image represents a point target produced by a corner reflector at the center of a flat scene. The performance is evaluated through two intensity contrast measures in logarithmic scale, quantifying the preservation of the point target with respect to the average intensity in the surrounding region ($C_{NN}$) and the average intensity of the whole background ($C_{BG}$). All the CNN-based methods in Table III perform prior classification as they have been trained without the point targets. In testing, a thresholding procedure is performed to remove the point targets prior to filtering and to copy them back right after. Overall, CNN-based techniques tend to present the highest values for these two metrics.

*5) Urban area case (Building):* The building image represents an isolated building over a homogeneous flat surface. The intense double reflection line resulting from the multiple scattering mechanisms should be preserved by the despeckling technique. The performance is evaluated employing a building smearing measure BS and an intensity contrast measure $C_{DR}$ in logarithmic scale. $C_{DR}$ quantifies the preservation of the double reflection segment with respect to the average intensity of the background. This is another case where the CNN-based methods better preserve the radiometric features of the building, presenting a BS closer to zero and a higher $C_{DR}$.

### F. Ablation study

In the following study, we want to assess the benefits of some of the features proposed for Speckle2Void.

*1) Original vs whitened:* First, we show the importance of the pixel-wise noise independence condition when training a blind-spot network. To assess it, we train Speckle2Void with two different datasets. One dataset is composed of real single-look complex images as they are provided by the focusing algorithm for the TerraSAR-X satellite, while the other dataset is composed of the same real SAR images but pre-processed by the decorrelator defined in [43]. For both datasets we use a $1 \times 1$ blind-spot shape, including solely the center pixel during the entire training. To better highlight the effect of the whitening procedure, we do not add the TV regularization in the loss. Fig. 10 shows the two resulting cleaned images together with the one obtained by the full Speckle2Void method (whitening+variable blind spot). The visual difference between the left image and the middle one shows that the decorrelator improves drastically the qualitative performance, since barely any denoising is performed in the first image.

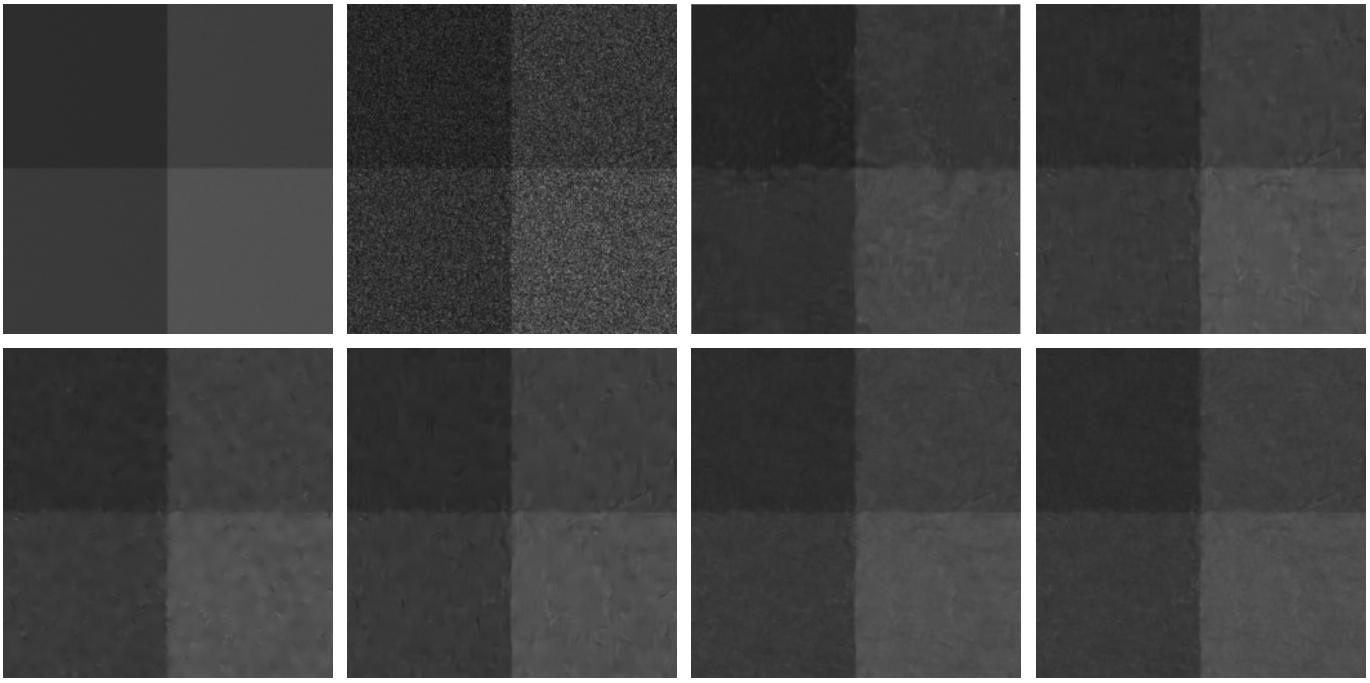

Fig. 9. *Squares* benchmark image. Top-Left to bottom-right: Clean, Noisy, SARBM3D, NL-SAR, CNN-based baseline, ID-CNN, Speckle2Void, Speckle2Void+NL

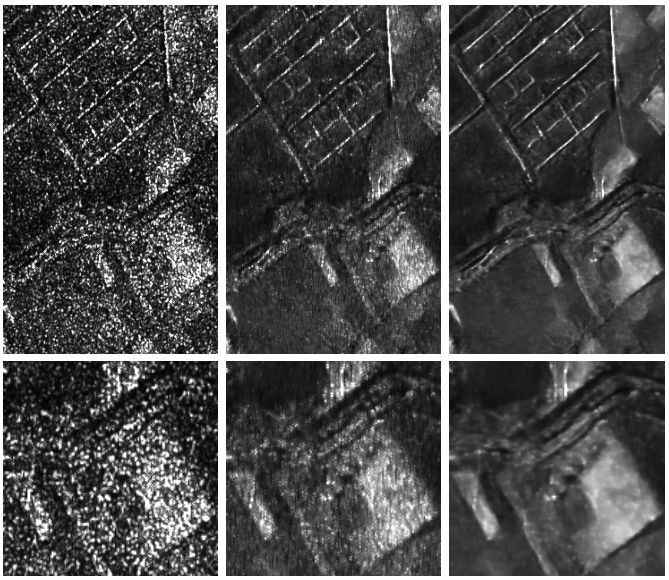

Fig. 10. From left to right: cleaned image resulting from the training with the original TerraSAR-X dataset (ENL 1.28), cleaned image resulting from the training with the whitened TerraSAR-X dataset (ENL 14.5) and Speckle2Void (ENL 88.5).

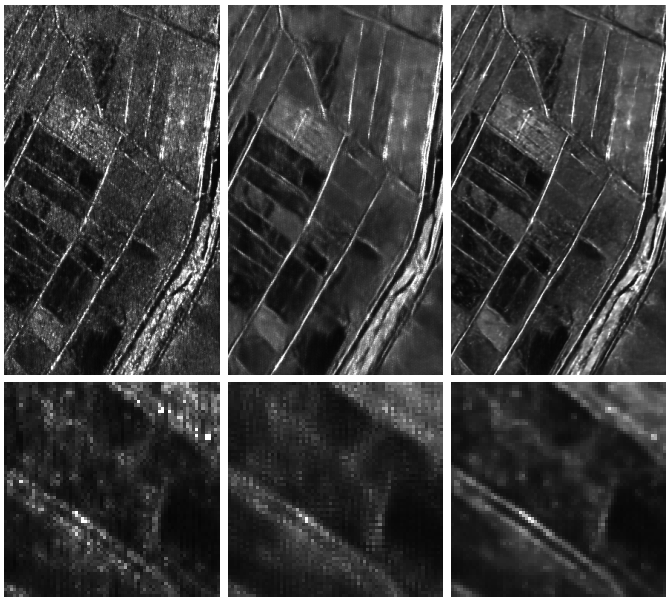

Fig. 11. From left to right: network with $1 \times 1$ blind-spot, network with $3 \times 3$ blind-spot, Speckle2Void

*2) Enlarging the blind-spot:* In our regularized training procedure we vary the shape of the blind-spot to account for the residual noise correlation that persists even after the whitening procedure. To better understand the effect of enlarging the size of the blind-spot structure, we compare Speckle2Void trained with the canonical $1 \times 1$ blind-spot shape against a $3 \times 3$ shape. Notice that, in this experiment, the latter uses the $3 \times 3$ blind-spot in testing as well, differently from the regularization procedure explained in IV-B which always uses a $1 \times 1$ blind spot in testing. Moreover, to better highlight the effect of the shape of the blind-spot, we do not add the TV regularization in the loss. Fig. 11 shows a visual comparison between the two methods. The left image is the result produced by the network with blind-spot of shape $1 \times 1$. We can notice sharper edges and more details with respect to the middle image produced by the network with blind-spot of shape $3 \times 3$, which looks more blurry. However, we also see more residual noise in the image on the left. Enlarging the shape of blind-spot structure leads to a more effective speckle noise reduction as the network uses surrounding pixels that

TABLE IV
BLIND-SPOT SIZE. MEASURES FOR SIMULATED SAR TEST IMAGES

| Image | Metric | 1x1 | 3x3 | Speckle2Void |
|---|---|---|---|---|
| Homogeneous | MoI ↑ | 0.977 | **1.000** | 0.988 |
| | MoR ↑ | **1.000** | 0.976 | 0.989 |
| | VoR ↑ | 0.874 | 0.861 | **0.88** |
| | ENL ↑ | 20.05 | 103.09 | **112.96** |
| | DG ↑ | 13.00 | 19.43 | **19.8** |
| Texture | MoI ↑ | 1.020 | **0.987** | 0.846 |
| | MoR ↑ | 0.834 | **0.838** | 0.808 |
| | VoR ↑ | **0.963** | 0.719 | 0.822 |
| | $C_x$ (2.40) | 2.45 | **2.43** | 2.21 |
| | DG ↑ | 3.34 | **4.03** | 3.45 |
| Squares | ES (up) ↓ | 0.064 | 0.074 | **0.058** |
| | ES (down) ↓ | **0.145** | 0.171 | 0.158 |
| | FOM ↑ | 0.783 | 0.795 | **0.834** |
| Corner | $C_{NN}$ ↑ | 7.77 | 7.77 | **7.79** |
| | $C_{BG}$ ↑ | **36.61** | 35.51 | 36.54 |
| Building | $C_{DR}$ ↑ | 65.9 | 65.86 | **65.92** |
| | BS ↓ | 0.4394 | 0.4159 | **0.4031** |

are less correlated with center pixel. A downside of expanding the blind-spot is to reduce the amount of relevant information for the network to estimate the center pixel, resulting in a smoother image with a loss of high frequency details, failing to preserve the original edges. In the image on the right we report the result of Speckle2Void, showing that the proposed method is able to achieve stronger speckle suppression with an impressive preservation of details.

Table IV provides a quantitative comparison using the benchmark dataset proposed in [51]. For the homogeneous case, Speckle2Void provides a stronger speckle suppression than the network with a blind-spot of shape $1\times1$ or with shape $3\times3$. The latter method presents a despeckling gain (DG) very close to the one of Speckle2Void and much higher than the one produced by the network with blind-spot of shape $1 \times 1$. This suggests the ability of the $3\times3$ blind-spot to disregard the strong noise correlation of the immediate neighboring pixels with respect to the center pixel, when producing the clean estimate. For the same reason, the network with blind-spot of shape $3 \times 3$ provides the best despeckling suppression ability in the DEM test case. The FOM metric for the squares case shows that a bigger blind-spot allows a better edge detection even in the presence of blurrier contours. Speckle2Void adds to the filtered image the necessary high frequency details to help the downstream detector algorithm. For the corner and building cases, the results of the three methods are comparable, since the radiometric preservation of the point targets strongly depends on the prior classification step that is the same in all the three methods.

*3) Effect of the TV regularizer:* Speckle2Void employs TV in the loss as an additional spatial regularizer. We aim to understand its impact by comparing Speckle2Void with a version trained without TV. Fig. 12 shows the resulting cleaned images, revealing the reduced amount of artifacts and smoother flat areas when the TV regularization is employed.

*4) Prior vs posterior:* The Bayesian framework, exploited in our method, makes use of the noisy SAR image to obtain the despeckled version by computing the expected value of the posterior distribution. The blind-spot CNN produces the parameters of the prior distribution. If we compute its expected value we obtain the prior despeckled image. In Fig. 13, the prior and the posterior images highlight the great qualitative improvement brought by the use of the noisy observations in the estimation of the cleaned image with the posterior mean. The prior image shows fuzzy edges and a disturbing granular pattern that makes the posterior image visually preferable.

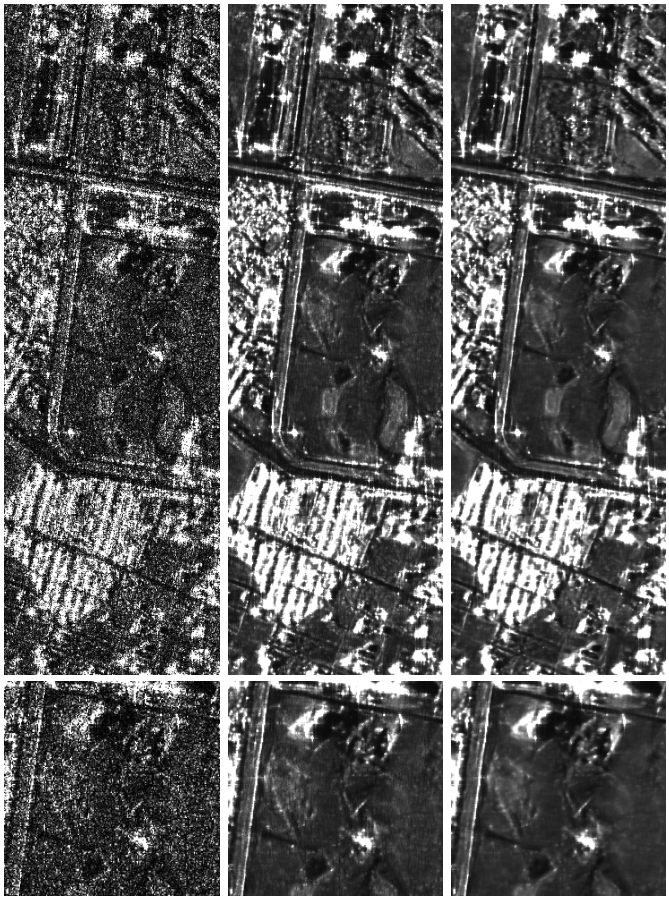

Fig. 12. From left to right: Noisy, Speckle2Void w/o TV and Speckle2Void.

### G. Transferability to Sentinel-1

In this section we present a result to show the performance of the Speckle2Void model trained on TerraSAR-X data when applied to Sentinel-1 single look images. Fig. 14 shows a qualitative result while the caption reports quantitative metrics. It is interesting to notice that Speckle2Void provides excellent performance, both qualitatively by showing strong speckle suppression while maintaining several details of the scene, and quantitatively according to the metrics presented in the previous sections. A more detailed study on how to train optimally on Sentinel-1, either by finetuning a pretrained model or from scratch, is out of the scope of this paper, but it would be an interesting future developement, especially in the context of studying how well self-supervised representations transfer across platforms.

### H. Training time and runtime comparisons

The training and inference run-times for all the methods considered in the experimental evaluation are shown in Table

TABLE V
TRAINING TIME AND RUNTIME COMPARISONS

| **Image** | **PPB** [31] | **SAR-BM3D** [7] | **NL-SAR** [32] | **Baseline CNN** | **ID-CNN** [12] | **Speckle2Void** | **Speckle2Void+NL** |
|---|---|---|---|---|---|---|---|
| Training | - | - | 0.8645 s (100x100) | 3 days 2 h | 7 h | 1 day 3 h | 6 days 19h |
| Inference (1000x1000) | 27.54 s | 223.51 s | 23.39 s | 0.587 s | 0.1627 s | 1.26 s | 432.41 s |

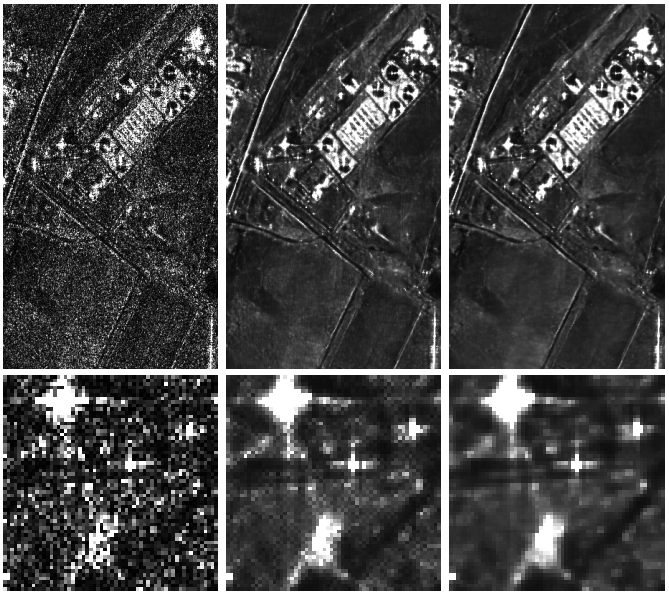

Fig. 13. From left to right: Noisy, Speckle2Void (Prior mean image), Speckle2Void (Posterior mean image).

V. The experiments have been performed on a PC with 64-GB RAM, an AMD Threadripper 1920X CPU, and an Nvidia 1080Ti GPU. All the CNN-based methods have been trained using the Tensorflow framework. The CNN-based methods have the lowest inference times except for the nonlocal version of Speckle2Void. This version is more expensive due to the non-local layers, which have to compute dynamic aggregation weights for all the pixels in a search window. Moreover, due to GPU memory constraints, the nonlocal version of Speckle2Void processes SAR images in multiple smaller patches, resulting in a longer inference time to reconstruct the entire clean image. The local version of Speckle2Void takes, on average, 1.26 seconds to process a $1000 \times 1000$ image, which is slightly higher than the inference times of the baseline CNN and ID-CNN models because it has to process the same image four times to compute the four half-plane receptive fields. However, it is significantly lower than the inference times of model-based methods. The training times affect only the CNN-based methods and span from some hours to several days.

## VI. CONCLUSION

In this paper we have presented Speckle2Void, a self-supervised Bayesian denoising framework for despeckling. The main obstacle in applying classical supervised deep learning methods to despeckling is represented by the vast content disparity between speckle injected natural images and real SAR images, often resulting in unfaithful cleaned images. Speckle2Void exploits a customized version of the blind-spot convolutional networks where the receptive field is constrained to exclude a variable amount of pixels throughout training to account for the correlation structure of the noise, introducing one of the first deep learning despeckling method purely based on real single-look complex SAR images. Speckle2Void is able to learn to produce excellent images with faithful details and no visible residual speckle noise.

## REFERENCES


[1] J.-S. Lee, "Speckle analysis and smoothing of synthetic aperture radar images," *Computer Graphics and Image Processing*, vol. 17, no. 1, pp. 24 – 32, 1981. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0146664X81800056

[2] V. S. Frost, J. A. Stiles, K. S. Shanmugan, and J. C. Holtzman, "A model for radar images and its application to adaptive digital filtering of multiplicative noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-4, no. 2, pp. 157–166, March 1982.

[3] D. Kuan, A. Sawchuk, T. Strand, and P. Chavel, "Adaptive restoration of images with speckle," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 35, no. 3, pp. 373–383, March 1987.

[4] A. Lopes, E. Nezry, R. Touzi, and H. Laur, "Structure detection and statistical adaptive speckle filtering in SAR images," *International Journal of Remote Sensing*, vol. 14, no. 9, pp. 1735–1758, 1993. [Online]. Available: https://doi.org/10.1080/01431169308953999

[5] Hua Xie, L. E. Pierce, and F. T. Ulaby, "SAR speckle reduction using wavelet denoising and Markov random field modeling," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 10, pp. 2196–2212, Oct 2002.

[6] F. Argenti and L. Alparone, "Speckle removal from SAR images in the undecimated wavelet domain," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 11, pp. 2363–2374, Nov 2002.

[7] S. Parrilli, M. Poderico, C. V. Angelino, and L. Verdoliva, "A nonlocal SAR image denoising algorithm based on LLMMSE wavelet shrinkage," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 2, pp. 606–616, Feb 2012.

[8] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, Aug 2007.

[9] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, July 2017.

[10] A. B. Molini, D. Valsesia, G. Fracastoro, and E. Magli, "DeepSUM: Deep Neural Network for Super-Resolution of Unregistered Multitemporal Images," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–13, 2019.

[11] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 3431–3440.

[12] P. Wang, H. Zhang, and V. M. Patel, "SAR Image Despeckling Using a Convolutional Neural Network," *IEEE Signal Processing Letters*, vol. 24, no. 12, pp. 1763–1767, Dec 2017.

[13] Q. Zhang, Q. Yuan, J. Li, Z. Yang, X. Ma, H. Shen, and L. Zhang, "Learning a dilated residual network for sar image despeckling," *Remote Sensing*, vol. 10, pp. 1–18, Feb 2018.

[14] G. Chierchia, D. Cozzolino, G. Poggi, and L. Verdoliva, "SAR image despeckling through convolutional neural networks," in *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 2017, pp. 5438–5441.

[15] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2Noise: Learning image restoration without clean data," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 80. PMLR, 10–15 Jul 2018, pp. 2965–2974.

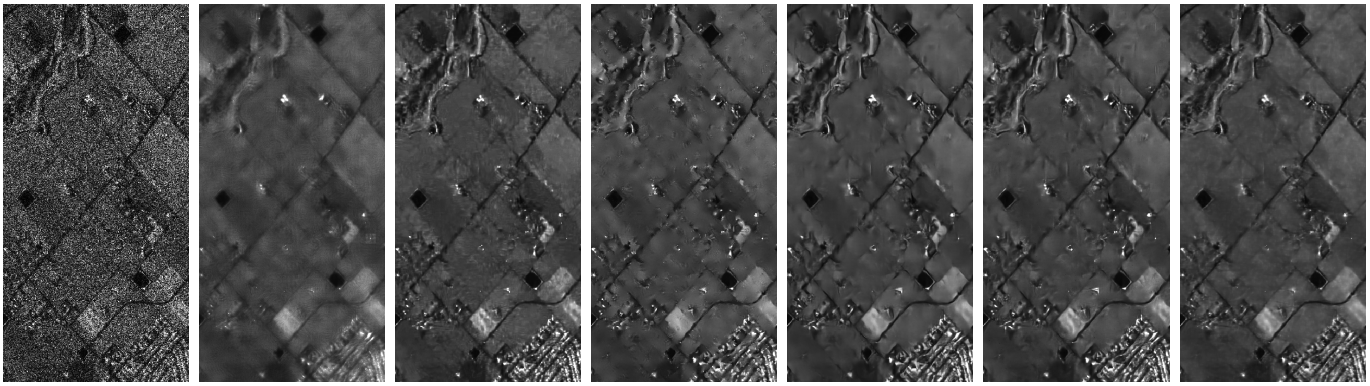

Fig. 14. Sentinel-1 image detail. From left to right: Noisy, PPB (ENL $= 141$, $\mu_r = 0.926$, $\sigma_r = 0.89$, $\mathcal{M} = 9.17$, RIS $= 0.1032$), SARBM3D (ENL $= 245$, $\mu_r = 0.954$, $\sigma_r = 0.787$, $\mathcal{M} = 4.8$, RIS $= 0.0227$), NL-SAR (ENL $= 150$, $\mu_r = 0.944$, $\sigma_r = 0.778$, $\mathcal{M} = 9.7$, RIS $= 0.0080$), CNN baseline (ENL $= \mathbf{384}$, $\mu_r = 0.979$, $\sigma_r = 0.900$, $\mathcal{M} = 3.27$, RIS $= 0.0128$), ID-CNN (ENL $= 259$, $\mu_r = 0.968$, $\sigma_r = 0.867$, $\mathcal{M} = 3.22$, RIS $= 0.0102$), Speckle2Void (ENL $= 299$, $\mu_r = \mathbf{0.981}$, $\sigma_r = \mathbf{0.939}$, $\mathcal{M} = \mathbf{2.70}$, RIS $= \mathbf{0.0016}$)


[16] A. Krull, T. Buchholz, and F. Jug, "Noise2void - learning denoising from single noisy images," in *2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2124–2132.

[17] J. Batson and L. Royer, "Noise2self: Blind denoising by self-supervision," in *Proceedings of the 36th International Conference on Machine Learning, ICML*, ser. Proceedings of Machine Learning Research, vol. 97. PMLR, 2019, pp. 524–533.

[18] S. Laine, T. Karras, J. Lehtinen, and T. Aila, "High-quality self-supervised deep image denoising," in *Advances in Neural Information Processing Systems*, 2019, pp. 6968–6978.

[19] A. Bordone Molini, D. Valsesia, G. Fracastoro, and E. Magli, "Towards Deep Unsupervised SAR Despeckling with Blind-Spot Convolutional Neural Networks," in *2020 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Oct 2020.

[20] H. Guo, J. E. Odegard, M. Lang, R. A. Gopinath, I. W. Selesnick, and C. S. Burrus, "Wavelet based speckle reduction with application to sar based atd/r," in *Proceedings of 1st International Conference on Image Processing*, vol. 1, 1994, pp. 75–79.

[21] L. Gagnon and A. Jouan, "Speckle filtering of SAR images: a comparative study between complex-wavelet-based and standard filters," in *Wavelet Applications in Signal and Image Processing V*, vol. 3169, International Society for Optics and Photonics. SPIE, 1997, pp. 80–91.

[22] A. Achim, P. Tsakalides, and A. Bezerianos, "Sar image denoising via bayesian wavelet shrinkage based on heavy-tailed modeling," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 8, pp. 1773–1784, 2003.

[23] S. Solbo and T. Eltoft, "Homomorphic wavelet-based statistical despeckling of sar images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 4, pp. 711–721, 2004.

[24] M. I. H. Bhuiyan, M. O. Ahmad, and M. N. S. Swamy, "Spatially adaptive wavelet-based method using the cauchy prior for denoising the sar images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 4, pp. 500–507, 2007.

[25] A. Achim, E. E. Kuruoglu, and J. Zerubia, "Sar image filtering based on the heavy-tailed rayleigh model," *IEEE Transactions on Image Processing*, vol. 15, no. 9, pp. 2686–2693, 2006.

[26] Hua Xie, L. E. Pierce, and F. T. Ulaby, "Despeckling sar images using a low-complexity wavelet denoising process," in *IEEE International Geoscience and Remote Sensing Symposium*, vol. 1, 2002, pp. 321–324.

[27] Min Dai, Cheng Peng, A. K. Chan, and D. Loguinov, "Bayesian wavelet shrinkage with edge detection for sar image despeckling," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 8, pp. 1642–1648, 2004.

[28] S. Foucher, G. B. Benie, and J. . Boucher, "Multiscale map filtering of sar images," *IEEE Transactions on Image Processing*, vol. 10, no. 1, pp. 49–60, 2001.

[29] F. Argenti, T. Bianchi, and L. Alparone, "Multiresolution map despeckling of sar images based on locally adaptive generalized gaussian pdf modeling," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3385–3399, 2006.

[30] B. Coll and J.-M. Morel, "A review of image denoising algorithms, with a new one," *SIAM Journal on Multiscale Modeling and Simulation*, vol. 4, Jan 2005.

[31] C. Deledalle, L. Denis, and F. Tupin, "Iterative weighted maximum likelihood denoising with probabilistic patch-based weights," *IEEE Transactions on Image Processing*, vol. 18, no. 12, pp. 2661–2672, Dec 2009.

[32] C. Deledalle, L. Denis, F. Tupin, A. Reigber, and M. Jäger, "Nl-sar: A unified nonlocal framework for resolution-preserving (pol)(in)sar denoising," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 2021–2038, 2015.

[33] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[34] P. Wang, H. Zhang, and V. M. Patel, "Generative adversarial network-based restoration of speckled sar images," in *2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2017, pp. 1–5.

[35] Y. Gui, L. Xue, and X. Li, "Sar image despeckling using a dilated densely connected network," *Remote Sensing Letters*, vol. 9, pp. 857–866, Sep 2018.

[36] J. Li, Y. Li, Y. Xiao, and Y. Bai, "Hdranet: Hybrid dilated residual attention network for sar image despeckling," *Remote Sensing*, vol. 11, p. 2921, Dec 2019.

[37] J. Zhang, W. Li, and Y. Li, "Sar image despeckling using multiconnection network incorporating wavelet features," pp. 1–5, 2019.

[38] F. Lattari, B. Leon, F. Asaro, A. Rucci, C. Prati, and M. Matteucci, "Deep learning for sar image despeckling," *Remote Sensing*, vol. 11, p. 1532, June 2019.

[39] D. Cozzolino, L. Verdoliva, G. Scarpa, and G. Poggi, "Nonlocal cnn sar image despeckling," *Remote Sensing*, vol. 12, p. 1006, March 2020.

[40] Y. Yuan, J. Sun, and J. Guan, "Blind SAR Image Despeckling Using Self-Supervised Dense Dilated Convolutional Neural Network," *ArXiv*, vol. abs/1908.01608, 2019.

[41] X. Ma, C. Wang, Z. Yin, and P. Wu, "Sar image despeckling by noisy reference-based deep learning method," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 12, pp. 8807–8818, 2020.

[42] A. C. Frery, H. . Muller, C. C. F. Yanasse, and S. J. S. Sant'Anna, "A model for extremely heterogeneous clutter," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 35, no. 3, pp. 648–659, May 1997.

[43] A. Lapini, T. Bianchi, F. Argenti, and L. Alparone, "Blind speckle decorrelation for SAR image despeckling," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 2, pp. 1044–1058, Feb 2014.

[44] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[45] D. Liu, B. Wen, Y. Fan, C. C. Loy, and T. S. Huang, "Non-local recurrent network for image restoration," in *Advances in Neural Information Processing Systems*, vol. 31, 2018, pp. 1673–1682.

[46] L. Deniz, R. Ospina, and A. Frery, "Unassisted quantitative evaluation of despeckling filters," *Remote Sensing*, vol. 9, April 2017.

[47] S. Vitale, D. Cozzolino, G. Scarpa, L. Verdoliva, and G. Poggi, "Guided patchwise nonlocal sar despeckling," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6484–6498, 2019.

[48] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, 1973.

[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[50] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th Int'l Conf. Computer Vision*, vol. 2, July 2001, pp. 416–423.
[51] G. Di Martino, M. Poderico, G. Poggi, D. Riccio, and L. Verdoliva, "Benchmarking framework for sar despeckling," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 3, pp. 1596–1615, 2014.