SAR Image Despeckling Using a Convolutional Neural Network

Puyang Wang, *Student Member, IEEE*, He Zhang, *Student Member, IEEE*, and Vishal M. Patel, *Senior Member, IEEE*

***Abstract*—Synthetic aperture radar (SAR) images are often contaminated by a multiplicative noise known as speckle. Speckle makes the processing and interpretation of SAR images difficult. We propose a deep-learning-based approach called, image despeckling convolutional neural network (ID-CNN), for automatically removing speckle from the input noisy images. In particular, ID-CNN uses a set of convolutional layers along with batch normalization and rectified linear unit activation function and a componentwise division residual layer to estimate speckle and it is trained in an end-to-end fashion using a combination of Euclidean loss and total variation loss. Extensive experiments on synthetic and real SAR images show that the proposed method achieves significant improvements over the state-of-the-art speckle reduction methods.**

***Index Terms*—Denoising, despecking, image restoration, synthetic aperture radar (SAR).**

I. Introduction

SYNTHETIC aperture radar (SAR) is a coherent imaging technology that is capable of producing high-resolution images of terrain and targets. SAR has the ability to operate at night and in adverse weather conditions, hence, it can overcome limitations of the optical and infrared systems. However, SAR images are often contaminated by multiplication noise known as speckle [1]. Speckle is caused by the constructive and destructive interference of the coherent returns scattered by small reflectors within each resolution cell. The presence of speckle noise in SAR images can often make the processing and interpretation difficult for computer vision systems as well as human interpreters. Hence, it is important to remove speckle from SAR images to improve the performance of various computer vision algorithms, such as segmentation, detection, and recognition.

Let $Y \in \mathbb{R}^{W\times H}$ be the observed image intensity, $X \in \mathbb{R}^{W\times H}$ be the noise free image, and $F \in \mathbb{R}^{W\times H}$ be the speckle noise. Then assuming that the SAR image is an average of L looks, the observed image Y is related to X by the following multiplicative model [2]:

$$Y = F \odot X \tag{1}$$

where F is the normalized fading speckle noise random variable and $\odot$ denotes the elementwise multiplication. The observed image Y is the Hadamard product of F and X. One common assumption on F is that it follows a Gamma distribution with unit mean and variance $\frac{1}{L}$ and has the following probability density function [2]

$$p(F) = \frac{1}{\Gamma(L)} L^L F^{L-1} e^{-LF} \tag{2}$$

where $\Gamma(\cdot)$ denotes the Gamma function and $F \geq 0$, $L \geq 1$.

Various methods have been developed in the literature to suppress speckle including multilook processing [3], [4], filtering methods [5]–[7], wavelet-based despecking methods [8]–[11], block-matching 3D (BM3D) algorithm [12] and total variation (TV) methods [13]. Note that some of these methods apply homomorphic processing in which the multiplicative noise is transformed into an additive noise by taking the logarithm of the observed data [14]. However, the logarithm operation often introduces a bias into the denoised image [14]. Furthermore, due to local processing nature of some of these methods, they often fail to preserve sharp features such as edges and often contain block artifacts in the denoised image [11], [14].

In recent years, deep-learning-based methods have shown to produce state-of-the-art results on various image processing tasks such are image restoration [15], [16] and super resolution [17]. However, to the best of our knowledge, image despeckling methods based on deep learning have not been extensively studied in the literature. Motivated by recent advances in deep learning, we propose a convolutional neural network (CNN) based approach for image despeckling. Rather than using a homomorphic transformation [11], we directly estimate speckle using the input images based on the observation model (1). The proposed image despeckling CNN (ID-CNN) method consists of several convolutional layers along with batch normalization [18] and rectified linear unit (ReLU) [19] activation function (see Fig. 1). In particular, the proposed architecture features a componentwise division-residual layer with skip-connection to estimate the denoised image. It is trained in an end-to-end fashion using a combination of Euclidean loss and TV loss. One of the main advantages of using deep-learning-based methods for image despeckling is that they learn parameters for image restoration directly from the training data rather than relying on

Manuscript received July 10, 2017; revised August 24, 2017; accepted September 19, 2017. Date of publication September 29, 2017; date of current version October 16, 2017. This work was supported by an ARO Grant W911NF-16-1-0126. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ananda Shankar Chowdhury. *(Corresponding author: Puyang Wang.)*

The authors are with the Department of Electrical and Computer Engineering, Rutgers University, Piscataway, NJ 08854 USA (e-mail: puyang.wang@rutgers.edu; he.zhang92@rutgers.edu; vishal.m.patel@rutgers.edu).

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

Digital Object Identifier 10.1109/LSP.2017.2758203

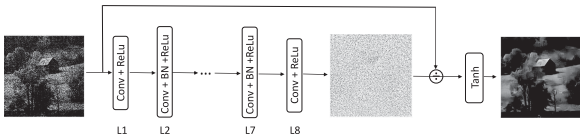

Fig. 1. Proposed ID-CNN network architecture for image despeckling.

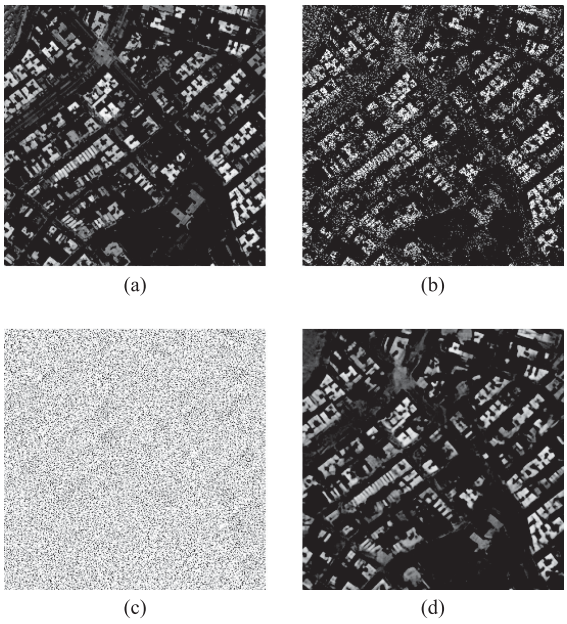

Fig. 2. Sample output of the proposed ID-CNN method for image despeckling. (a) Clean image, (b) speckled image, (c) estimated noise, and (d) despeckled image.

predefined image priors or filters. Fig. 2 shows a sample output from our ID-CNN method. Given the clean image in Fig. 2(a), the corresponding noisy image shown in Fig. 2(b) is synthesized through the aforementioned multiplicative noise model. ID-CNN estimates the denoised image shown in Fig. 2(d). The estimated speckle is also shown in Fig. 2(c). As can be seen by comparing Fig. 2(b) and (d) that our method is able to denoise the speckled image reasonably well.

II. Proposed Method

In this section, we provide details of the proposed ID-CNN method for image despeckling. The detailed architecture of the proposed ID-CNN is shown in Fig. 1, where Conv, BN, and ReLu stand for convolution, batch normalization, and rectified linear unit, respectively. Using a specific CNN architecture, we learn a mapping from an input SAR image to a despeckled image. One possible solution to this problem would be to transform the image into a logarithm space and then learn the corresponding mapping via CNN [20]. However, this approach needs extra steps to transfer the image into a logarithm space and from a logarithm space back to an image space. As a result, the overall algorithm cannot be learned in an end-to-end fashion. To address this issue, a division residual method is leveraged in our method where a noisy SAR image is viewed as a product of speckle with the underlying clean image [i.e., (1)]. By incorporating the proposed componentwise division residual layer into the network, the convolutional layers are forced to learn the speckle component during the training process. In other words, the output before the division residual layer represents the estimated speckle. Then, the despeckled image is obtained by simply dividing the input image by the estimated speckle.

TABLE I
Network Configuration

	Layer	Filter size	#Filters	Output size
L1	Conv	$3 \times 3 \times 1$	64	$256 \times 256 \times 64$
	ReLU			$256 \times 256 \times 64$
L2-L7	Conv	$3 \times 3 \times 64$	64	$256 \times 256 \times 64$
	BN			$256 \times 256 \times 64$
	ReLU			$256 \times 256 \times 64$
L8	Conv	$3 \times 3 \times 64$	1	$256 \times 256 \times 1$
	ReLU			$256 \times 256 \times 1$

A. Network Architecture

The noise-estimating part of ID-CNN network consists of eight convolutional layers (along with batch normalization and ReLU activation functions), with appropriate zero-padding to make sure that the output of each layer shares the same dimension with that of the input image. Batch normalization is added to alleviate the internal covariate shift by incorporating a normalization step and a scale and shift step before the nonlinearity in each layer. Each convolutional layer (except for the last convolutional layer) consists of 64 filters with stride of one. Then, the division residual layer with skip connection divides the input image by the estimated speckle. A hyperbolic tangent layer is stacked at the end of the network, which serves as a nonlinear function. The details of the network configuration are given in Table I. Here, L1 and L8 stand for the sequence of Conv–ReLU layers as depicted in Fig. 1. Similarly, L2–L7 denote Conv–BN–ReLU layers.

B. Loss Function

Loss functions form an important and integral part of learning process, especially in CNN-based image reconstruction tasks. Several works have explored different loss functions and their combinations for effective learning of tasks such as super-resolution [17], semantic segmentation [21], and style transfer [22]. Previous works on CNN-based image restoration optimized over pixelwise L2-norm (Euclidean loss) or L1-norm between the predicted and ground truth images.

Given an image pair $\{X, Y\}$, where Y is the noisy input image and X is the corresponding ground truth, the per-pixel Euclidean loss is defined as

$$L_E(\phi) = \frac{1}{WH} \sum_{w=1}^{W} \sum_{h=1}^{H} \|\phi(Y^{w,h}) - X^{w,h}\|_2^2 \qquad (3)$$

where ϕ is the learned network (parameters) for generating the despeckled output and $\hat{X} = \phi(Y^{w,h})$. Note that we have assumed that X and Y are of size $W \times H$.

While the Euclidean loss has shown to work well on many image restoration problems, it often results in various artifacts on the final estimated image. To overcome this issue, an additional

Fig. 3. Sample images used to train our network.

TV loss is incorporated into the loss function to encourage more smooth results. The TV loss is defined as follows

$$L_{\mathrm{TV}} = \sum_{w=1}^{W} \sum_{h=1}^{H} \sqrt{(\hat{X}^{w+1,h} - \hat{X}^{w,h})^2 + (\hat{X}^{w,h+1} - \hat{X}^{w,h})^2}. \tag{4}$$

Finally, the overall loss function is defined as follows

$$\mathrm{Loss} = L_E + \lambda_{\mathrm{TV}} L_{\mathrm{TV}} \tag{5}$$

where L_E represents the per-pixel Euclidean loss function and L_{TV} represents the TV loss. Here, $\lambda_{\mathrm{TV}} > 0$ is a predefined weight for the TV loss function. We adjust this parameter to control the importance of the TV loss. Since the Euclidean loss is directly related to the accuracy of despeckling and the TV loss controls the smoothness, it is evident that λ_{TV} should be far less than 1 so that the recovered image contains pixel level details and at the same time maintains the smoothness.

III. Experimental Results

In this section, we present the results of our proposed ID-CNN algorithm on both synthetic and real SAR images. We compare the performance of our method with that of the following six despeckling algorithms: Lee filter [5], Kuan filter [23], probabilistic patch-based filter (PPB) [24], SAR-BM3D [25], CNN [15], and SAR-CNN [20]. Note that [15], [20], [24], and [25] are the most recent state-of-the-art image restoration algorithms. For all the compared methods, parameters are set as suggested in their corresponding papers. For the baseline CNN method, we adopt the network structure proposed in [15] and omit the residual learning part so that the network learns a mapping from a noisy input image to a clean target image. Note that one of the major differences among our method and [15] and [20] is that we do not take the log of the observed noisy data but rather directly operate on the original noisy image. Another fundamental difference between [20] and our work is that the output of [20] is the log noise, whereas ours is the denoised image.

All training based methods including our method, ID-CNN, share the same training dataset. To train the proposed ID-CNN, we generate a dataset that contains 3665 image pairs. Training images are collected from the uncompressed colour image database (UCID), BSD-500, and scraped Google Maps images [26] and the corresponding speckled images are generated using (1). All images are resized to 256×256. Sample images used for training our network are shown in Fig. 3. The entire network is trained using the adaptive moment estimation (ADAM) optimization method [27], with minibatches of size 16 and learning rate of 0.0002. During training, the regularization parameter λ_{TV} is set equal to 0.002.

(a) (b)

Fig. 4. Sample results of the proposed ID-CNN with and without TV loss. Note that white artifacts exist when TV is not considered in the loss function. (a) ID-CNN without TV loss and (b) ID-CNN with TV loss.

TABLE II
Quantitative Results for Various Experiments on Synthetic Images

	Metric	Noisy	Lee	Kuan	PPB	SAR-BM3D	CNN	SAR-CNN	ID-CNN
$L = 1$	PSNR	14.53	21.48	21.95	21.74	22.99	21.04	23.59	**24.74**
	SSIM	0.369	0.511	0.592	0.619	0.692	0.630	0.640	**0.727**
	UQI	0.374	0.450	0.543	0.488	0.591	0.560	0.561	**0.621**
	DG	–	16.01	17.08	14.30	17.17	14.98	20.87	**23.51**
$L = 4$	PSNR	18.49	22.12	22.84	23.72	24.96	22.60	26.20	**26.89**
	SSIM	0.525	0.555	0.650	0.725	0.782	0.722	0.771	**0.818**
	UQI	0.527	0.485	0.594	0.605	0.679	0.648	0.688	**0.723**
	DG	–	8.35	10.00	10.52	14.89	13.76	17.74	**19.33**
$L = 10$	PSNR	20.54	22.30	23.11	24.92	26.45	23.52	27.63	**28.07**
	SSIM	0.602	0.571	0.671	0.779	0.834	0.741	0.825	**0.853**
	UQI	0.599	0.498	0.613	0.678	0.745	0.683	0.741	**0.765**
	DG	–	4.06	5.93	7.75	13.61	7.82	16.32	**17.35**

A. Ablation Study

We perform an ablation study to demonstrate the effects of different losses in the proposed method. Each of the losses is added one by one to the network and the results for each configuration is compared. In the first configuration, only L_E loss is minimized to train the network. The number of looks L is set equal to 1. The restored image is shown in Fig. 4(a). It can be seen that most of the speckle is removed from the noisy image; however, the denoised image still suffers from some artifacts. When the network is trained by minimizing the overall loss which consists of both Euclidean and TV losses, the results get better. The use of TV loss removes those unwanted artifacts that were present in Fig. 4(a). This can be clearly seen by comparing the zoomed in patches shown at the top right corner of these figures. This experiment clearly shows the significance of having both Euclidean and TV losses in our framework.

B. Results on Synthetic Images

We randomly selected 85 speckled images out of the 3665 images in the dataset. The remaining 3580 images are used for training the network. Experiments are carried out on three different noise levels. In particular, the number of looks L is set equal to be 1, 4, and 10, respectively. The peak signal to noise ratio (PSNR), structural similarity index (SSIM) [28], universal quality index (UQI) [29], and despeckling gain (DG) [30] are used to measure the denoising performance of

Fig. 5. From left to right: SAR images, PPB [24], SAR-BM3D [25], SAR-CNN [20], and ID-CNN.

TABLE III
ESTIMATED ENL RESULTS ON REAL SAR IMAGES

# chip	PPB	SAR-BM3D	CNN	SAR-CNN	ID-CNN
1	42.49	69.26	32.32	50.76	**89.43**
2	8.63	10.95	7.50	8.93	**13.90**
3	103.25	127.38	31.65	99.13	**193.00**
4	34.84	63.83	7.65	43.13	**69.40**

TABLE IV
RUNTIME COMPARISONS FOR DESPECKLING AN IMAGE OF SIZE 256×256

PPB	SAR-BM3D	CNN	SAR-CNN	ID-CNN
9.82 s	14.40 s	2.42 s	2.49 s	**0.56 s**

different methods. Average results of 85 test images corresponding this experiment are shown in Table II. As can be seen from this table, in all three noise levels, ID-CNN provides the best performance compared to the other despeckling methods. Interestingly, the CNN method that directly learns a mapping from a noisy input image to a clean target image using a Euclidean loss performs worse than our method and PPB [24] in many cases. This experiment clearly shows the significance of the proposed componentwise division residual layer-based CNN as well as the use of Euclidean + TV loss for image despeckling.

C. Results on Real SAR Images

We also evaluate the performance of the proposed method and recent state-of-the-art methods on real SAR images captured by RADARDAT-1 operating on *C* band [31]. Since the true reflectivity fields are not available, we use the equivalent number of looks (ENL) [32], [30], [33] to measure the performance of different image despeckling methods. The ENL values are estimated from the homogeneous regions (shown with red boxes in Fig. 5). The ENL results are also tabulated in Table III. It can be observed from these results that the proposed ID-CNN outperforms the others compared methods in all four homogeneous blocks. These results also demonstrate that our ID-CNN method can achieve much better performance in suppressing speckle in real SAR images.

When a clean reference is missing, visual inspection is another way to qualitatively evaluate the performance of different methods. The despeckled results corresponding to the real images are shown in Fig. 5. The second to fifth columns of Fig. 5 show the despeckled images corresponding to PPB, SAR-BM3D, SAR-CNN, and ID-CNN, respectively. It can be observed that ENL results are consistent with the visual results. No obvious speckle exist in ID-CNN while PPB and SAR-CNN suffer from some noticeable artifacts. It is also evident from these figures that filter-based reconstructions such as PPB and SAR-BM3D generally generate blurry edges compared to SAR-CNN and ID-CNN.

D. Runtime Comparisons

Experiments were carried out in MATLAB R2016b using the MatConvNet toolbox, with an Intel Xeon CPU at 3.00 GHz and an Nvidia Titan-X GPU. For ID-CNN, the entire network was trained using the torch framework [34]. The runtimes reported in Table IV are the average results obtained after multiple executions of each algorithm. Interestingly, once the training is over, ID-CNN exhibits the lowest run-time complexity. Note that even compared with the other CNN-based methods, our ID-CNN still has a lower run-time possibly because it has only eight fully convolutional layers compared to 17 in other CNN-based methods [15], [20].

IV. CONCLUSION

We have proposed a new method of speckle reduction in SAR imagery based on CNNs. Compared to nonlocal filtering and BM3D image despeckling methods, our CNN-based method generates the despeckled version of an SAR image through a single feedforward process. Another uniqueness of our method is that the network is designed to recover the noise by convolutional layers and then the noisy input is divided by the estimated noise that results in the denoised output. This strategy, similar to residual learning, is inspired by the observation that an SAR image can be viewed as a product of a clean reference and noise. Results on synthetic and real SAR data show promising qualitative and quantitative results.

REFERENCES

[1] J. W. Goodman, "Some fundamental properties of speckle," *J. Opt. Soc. Amer.*, vol. 66, no. 11, pp. 1145–1150, Nov. 1976.

[2] F. Ulaby and M. C. Dobson, *Handbook of Radar Scattering Statistics for Terrain*. Norwood, MA, USA: Artech House, 1989.

[3] C. Oliver and S. Quegan, *Understanding Synthetic Aperture Radar Images*. Norwood, MA, USA: Artech House, 1998.

[4] P. Thompson, D. E. Wahl, P. H. Eichel, D. C. Ghiglia, and C. V. Jakowatz, *Spotlight-Mode Synthetic Aperture Radar: A Signal Processing Approach*. Norwell, MA, USA: Kluwer, 1996.

[5] J.-S. Lee, "Speckle analysis and smoothing of synthetic aperture radar images," *Comput. Graph. Image Process.*, vol. 17, no. 1, pp. 24–32, 1981.

[6] 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 Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-4, no. 2, pp. 157–166, Mar. 1982.

[7] A. Baraldi and F. Parmiggiani, "A refined gamma MAP SAR speckle filter with improved geometrical adaptivity," *IEEE Trans. Geosci. Remote Sens.*, vol. 33, no. 5, pp. 1245–1257, Sep. 1995.

[8] H. Xie, L. E. Pierce, and F. T. Ulaby, "SAR speckle reduction using wavelet denoising and Markov random field modeling," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 10, pp. 2196–2212, Oct. 2002.

[9] F. Argenti and L. Alparone, "Speckle removal from SAR images in the undecimated wavelet domain," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 11, pp. 2363–2374, Nov. 2002.

[10] A. Achim, P. Tsakalides, and A. Bezerianos, "SAR image denoising via Bayesian wavelet shrinkage based on heavy-tailed modeling," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 8, pp. 1773–1784, Aug. 2003.

[11] V. M. Patel, G. R. Easley, R. Chellappa, and N. M. Nasrabadi, "Separated component-based restoration of speckled SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 2, pp. 1019–1029, Feb. 2014.

[12] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising with block-matching and 3D filtering," *Proc. SPIE*, vol. 6064, 2006, Art. no. 606414.

[13] J. M. Bioucas-Dias and M. A. T. Figueiredo, "Multiplicative noise removal using variable splitting and constrained optimization," *IEEE Trans. Image Process.*, vol. 19, no. 7, pp. 1720–1730, Jul. 2010.

[14] F. Argenti, A. Lapini, T. Bianchi, and L. Alparone, "A tutorial on speckle reduction in synthetic aperture radar images," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 3, pp. 6–35, Sep. 2013.

[15] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[16] X. Fu, J. Huang, X. Ding, Y. Liao, and J. Paisley, "Clearing the skies: A deep network architecture for single-image rain removal," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2944–2956, Jun. 2017.

[17] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2014, pp. 184–199.

[18] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Machine Learn. (ICML-15)*, 2015, pp. 448–456.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[20] G. Chierchia, D. Cozzolino, G. Poggi, and L. Verdoliva, "SAR image despeckling through convolutional neural networks," arXiv:1704.00275, 2017.

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

[22] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2414–2423.

[23] D. T. Kuan, A. A. Sawchuk, T. C. Strand, and P. Chavel, "Adaptive noise smoothing filter for images with signal-dependent noise," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-7, no. 2, pp. 165–177, Mar. 1985.

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

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

[26] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1125–1134.

[27] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.,* 2015.

[28] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[29] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Process. Letters*, vol. 9, no. 3, pp. 81–84, Mar. 2002.

[30] G. Di Martino, M. Poderico, G. Poggi, D. Riccio, and L. Verdoliva, "Benchmarking framework for SAR despeckling," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 3, pp. 1596–1615, Mar. 2014.

[31] I. Cumming and F. Wong, *Digital Processing of Synthetic Aperture Radar Data: Algorithms and Implementation*. Norwood, MA, USA: Artech House, 2005.

[32] J.-S. Lee, "Speckle analysis and smoothing of synthetic aperture radar images," *Comput. Graph. Image Process.*, vol. 17, no. 1, pp. 24–32, Sep. 1981.

[33] L. Gomez, R. Ospina, and A. C. Frery, "Unassisted quantitative evaluation of despeckling filters," *Remote Sens.*, vol. 9, no. 4, 2017, Art. no. 389.

[34] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A MATLAB-like environment for machine learning," in *Proc. BigLearn, NIPS Workshop*, 2011.