Deep learning-based compression and despeckling of SAR images

Nils Foix-Colonier[a,b], Joel Amao-Oliva[a], and Francescopaolo Sica[c]
[a]German Aerospace Center (DLR), Münchener Strasse 20, 82234 Weßling, Germany
[b]Nantes Université – École Centrale Nantes – CNRS LS2N UMR 6004, F-44000 Nantes, France
[c]University of the Bundeswehr Munich, Werner-Heisenberg-Weg 39, 85579 Neubiberg, Germany

Abstract

Combining despeckling and compression tasks is worthwhile because a decrease in the amount of information to be encoded will result in a more efficient data downlink. This paper presents a self-supervised solution to performing joint compression and despeckling of SAR images, with an estimation of the reflectivity based on an original adaptation of recent machine learning-based advances in the fields of image compression and SAR images despeckling. The proposed solution was successfully tested on real-world data from TerraSAR-X, showing great potential for achieving state-of-the-art despeckling under the constraints of end-to-end optimized compression with variational autoencoders.

1 Introduction

The usefulness of spaceborne synthetic aperture radar (SAR) systems, with their wide range of valuable applications, is a well-established fact. They can be used anytime and under any weather conditions to monitor forest cover, to secure maritime areas, to measure geophysical parameters remotely, etc. This is why data downlink and its interpretability are key issues.

The transmission of satellite data to the ground segment is subject to multiple constraints, such as the length of the moment when the satellite is visible from a terrestrial antenna, the bandwidth limit on data flow, and the financial cost associated with the duration of use of a sophisticated ground antenna. The aim is to maximize on-board compression to increase the amount of data transmitted and to reduce the inherent costs in using such infrastructures.

It should be noted that an unavoidable multiplicative noise, speckle, largely disrupts SAR images. We will here consider speckle as unwanted high-frequency information, so it seems natural to introduce on-board denoising of SAR images before compression and transmission. Doing so will improve compression without resorting to the usual trade-off of sacrificing image quality, but rather by estimating the underlying SAR reflectivity before downlink.

The work presented in this article proposes a joint implementation of compression and despeckling, based on the latest state-of-the-art methods in the fields of despeckling [1] and image compression [2], which respectively provided the basis for a training strategy and a neural network architecture. These methods are adapted to propose a unique solution, which operates in a single step of joint compression and despeckling, which could be used on-board to produce a bitstream that could then be decoded on the ground. The proposed solution has been experimented on real-world data acquired by TerraSAR-X.

2 Background concepts

2.1 Self-supervised despeckling

The despeckling of a SAR image is no ordinary task, and classical filtering methods such as SAR-BM3D [3] are now largely outperformed by recent machine learning-based approaches with convolutional neural networks (CNNs). However, they are often limited because there is no noiseless SAR image to be used as a "ground truth" reference for supervised machine learning. The remaining options that are usually implemented are: supervised learning with a temporal average of co-registered single-look complex (SLC) images, self-supervised learning with two co-registered images (SAR2SAR) [4], or with masked values (Speckle2Void) [5]. The main drawbacks of such methods are the multiple – and often heavy – preprocessing steps involved, such as change detection between two co-registered images or spectral equalization.

Because the real and imaginary parts of an SLC may be considered as two independent identically distributed realizations of the same signal, a new approach called MERLIN (coMplex sElf-supeRvised despeckLINg) has been proposed [1]. It has shown significantly better results when compared to other methods [6], while proving easier to implement, with great flexibility regarding the choice of neural network architecture, since it is mainly a training and inference strategy.

Let a received SLC be denoted $z = Ae^{i\varphi} = a + ib$, with $(a, b) \in \mathbb{R}^2$, where A is the amplitude and φ the phase. The received signal's probability density function p_z can be re-written as follows:

$$\begin{aligned} p_z(z) &= \frac{1}{\pi\sigma} e^{-|z|^2/\sigma} = \frac{1}{\pi\sigma} e^{-|a^2+b^2|/\sigma} \\ &= \frac{1}{\sqrt{2\pi}\sqrt{\sigma/2}} e^{-a^2/\sigma} \frac{1}{\sqrt{2\pi}\sqrt{\sigma/2}} e^{-b^2/\sigma} \end{aligned} \tag{1}$$

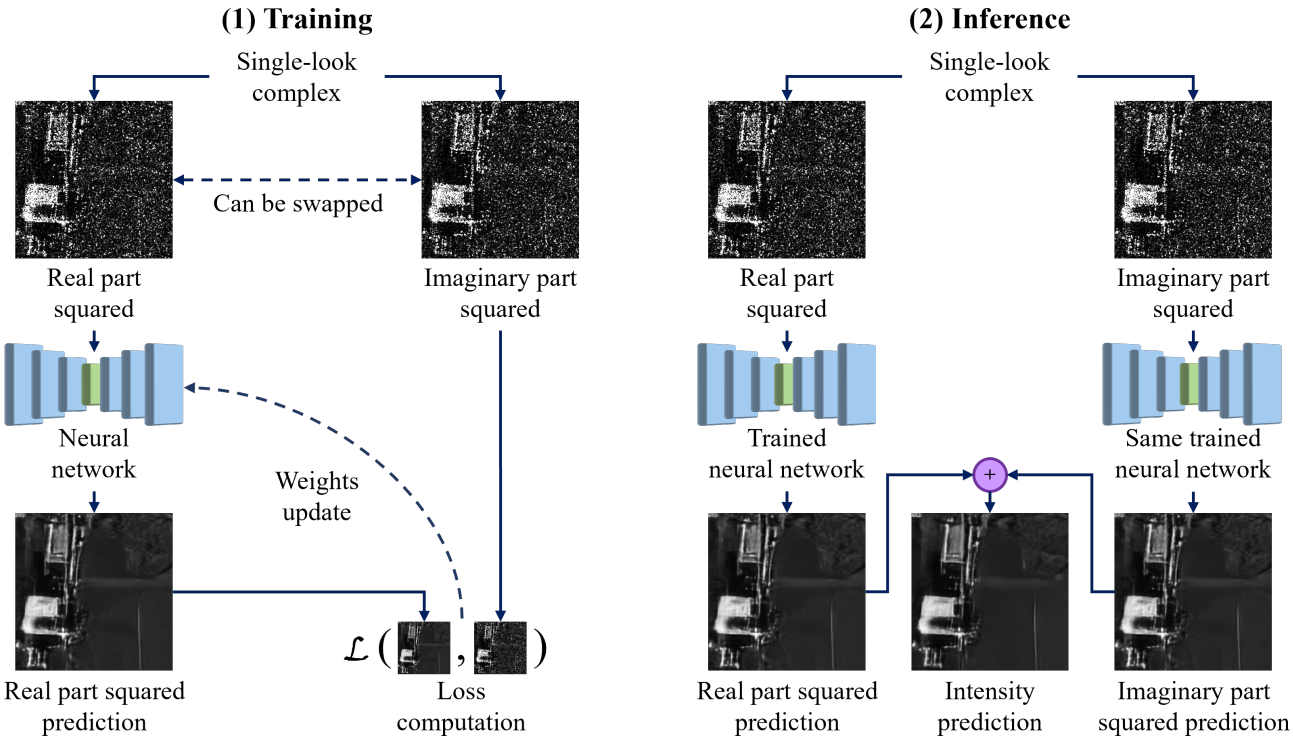


Figure 1 (1) MERLIN training strategy, in which the real part and the imaginary part may be swapped in order to increase the size of the training dataset. (2) Inference of the intensity with a trained network architecture used twice.

It is clear from equation (1) and from [1] that real and imaginary parts have the same distribution $\mathcal{N}(0, \sigma/2)$. When comparing real and imaginary parts – squared for visualization purposes – of real-world data, the main visible difference is the noise, which, under the assumption of Goodman's fully developed speckle model, is i.i.d. realizations of speckle. Consequently, in the same way that SAR2SAR trains a network to despeckle with two co-registered images, it is possible to do likewise, using $\boldsymbol{Re}^2$ the squared real and $\boldsymbol{Im}^2$ the squared imaginary parts of an SLC. The main advantage of this method is that there is no longer the need to co-register two images or to undertake changes detection [6]. The training strategy and the way to perform intensity image (amplitude image squared) inference are shown in Figure 1.

2.2 End-to-end optimized compression

Traditional data compression techniques are generally based on three successive steps that produce a bitstream, followed by the three associated inverse operations which estimate the original data. The transform, quantization and entropy coding operations form the encoder, while their counterparts constitute the decoder, as shown in Figure 2. The transform allows data to be represented in a space more suited to compression. For instance, JPEG2000 uses discrete wavelet transform to perform compression in the scales domain. The idea behind end-to-end optimized compression with neural networks is to take advantage of the efficient latent representations that autoencoders can achieve. The latter performs dimensionality reduction but not compression. Consequently, variational autoencoders trained to produce bitstreams – hence the characterization of "end-to-end" – have been developed. The variational nature of these autoencoders provides priors on the latent space, enabling its entropy encoding. State-of-the-art methods for image compression are based on such neural networks and have been further improved [2] by introducing a scale hyperprior. The loss function used to train such networks combines a distortion term $D(x, \hat{x})$, which measures the differences between original and estimated data, and a rate term $R(c)$, which evaluates the efficiency of the compression. Both terms are minimized at once, one with respect to the other by introducing a Lagrangian multiplier λ.

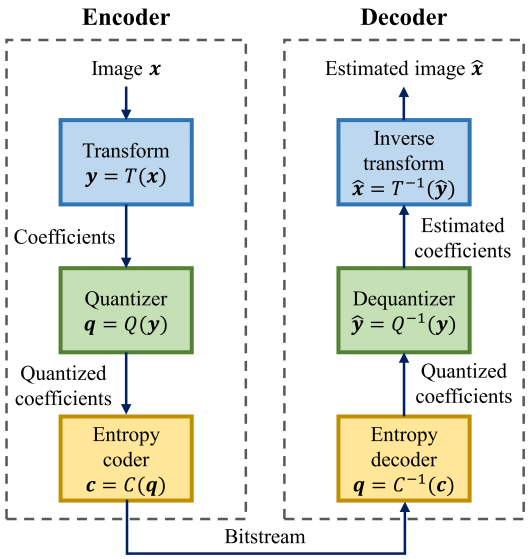


Figure 2 Usual framework for transform coding.

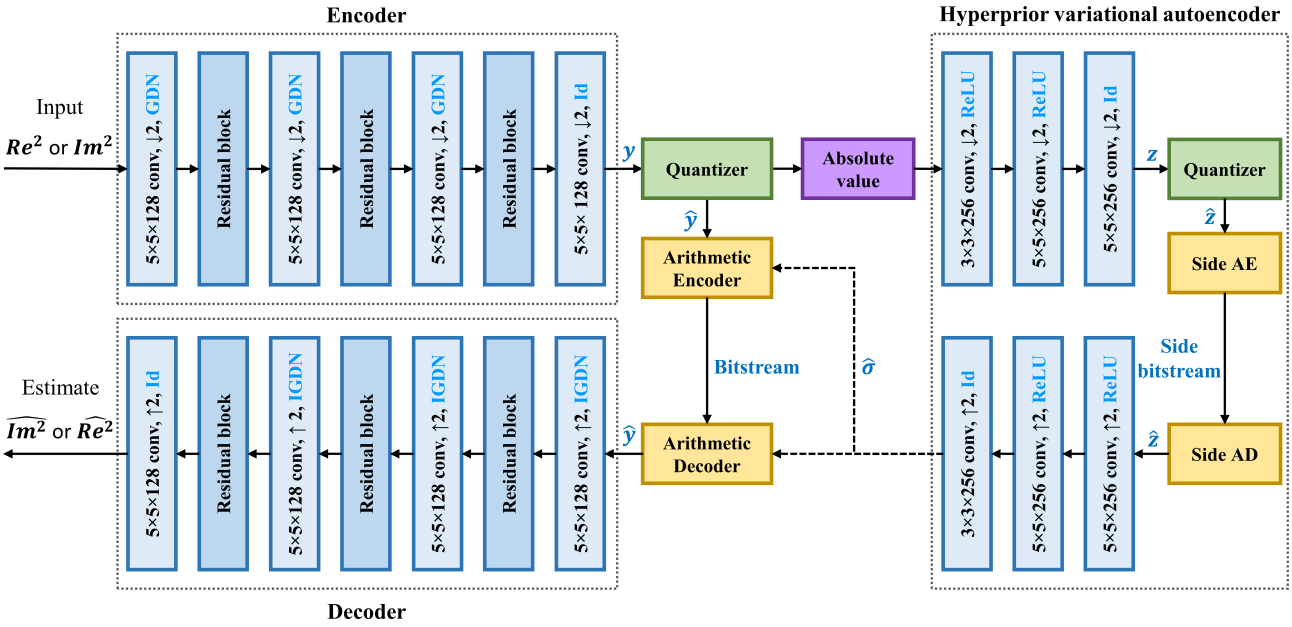


Figure 3 The framework proposed for training the neural network to perform joint despeckling and compression on a squared real or imaginary part of an SLC. The hyperprior variational autoencoder provides the scales $\hat{\sigma}$ used by the main entropy encoder and decoder. The activation functions are written in cyan, see [7] for technicalities on GDN and IGDN. Downscaling by a factor of two along height and width is denoted ↓2, while ↑2 corresponds to upscaling. The first two numbers describing the convolutions are for the 2D kernel dimensions and the last one is the number of filters.

3 Methodology

3.1 Joint approach

We propose a joint approach for SAR image compression and despeckling. Joint approaches for compression and denoising with deep learning have been developed, but only for optical images, [8] using simulated noise. Since noiseless SAR images do not exist, the same method cannot be applied here. The simulation of SAR data and of speckle does not generally allow models trained in this way to be used effectively on real data because of the domain gap [9].

More specifically, the despeckling method presented in subsection 2.1 provides a training strategy but does not limit the choice of neural network architecture, allowing an autoencoder to be chosen. In addition, end-to-end optimized compression can be performed using the concepts presented in subsection 2.2. It should be noted that unlike the U-Net network used in [1], an autoencoder does not have skip connections, which largely contribute to restoring details while decoding the latent space. This makes it more difficult to obtain near-quality results with autoencoders. However, since the compression task subjects us to certain constraints, we have chosen a variational autoencoder architecture, with residual blocks, as shown in Figure 4, to further preserve detail. It is equipped with quantization and entropy coding/decoding of the latent space, with a side hyperprior variational autoencoder, which offers better compression rates (even when considering the addition of the side bitstream used to transmit the scales for the priors) and provides the network with a high degree of adaptability regarding the input data.

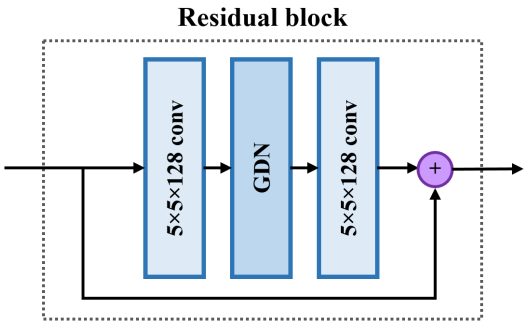


Figure 4 Residual blocks inner architecture.

3.2 Data processing

For the proposed work, we used X-band stripmap mode SLC data with HH polarization acquired by the TerraSAR-X satellite (ESA archive). Three images were used to create the training, validation, and test sets. As neural networks only accept static data shapes, each image was divided into 256×256 patches. Around 30,000 patches were used. However, the number of elements in the dataset can be doubled. Indeed, it is possible with a single complex data patch to create two input/reference pairs for training the neural network: $(\boldsymbol{Re^2}, \boldsymbol{Im^2})$ and $(\boldsymbol{Im^2}, \boldsymbol{Re^2})$.

For the data to be used within the framework shown in Figure 3, it needs to undergo several pre-processing steps: spectrum centering and symmetrization to avoid correlations between real and imaginary parts, logarithmic domain transformation and normalization to limit the dynamic range and increase training efficiency. Finally, it is divided into patches.

3.3 Deep learning-related aspects

The loss used to train the model with the architecture and the training strategy shown in Figure 3 is an original combination of the losses found in [1] and [2] :

$$\mathcal{L}\left(\hat{\boldsymbol{x}}, \boldsymbol{x}^{\boldsymbol{ref}}\right) = R\left(\hat{\boldsymbol{y}}\right) + R\left(\hat{\boldsymbol{z}}\right) + \lambda\, D\left(\hat{\boldsymbol{x}}, \boldsymbol{x}^{\boldsymbol{ref}}\right) \quad (2)$$

Where R is an entropy-based estimation of the bit-rate in bit per pixel (BPP), and D is the distortion term found in [1]. However, the actual distortion measurement can be any similarity metric (MSE, SSIM, etc.). The Lagrangian multiplier λ is a user-defined value before the training of the network, defining an arbitrary rate-distortion trade-off.

To train the neural network as shown in Figure 3, the following hyperparameters were used: 50 epochs, Adam optimizer with a learning rate of 10^{-4} with exponential decay, batch size of 2, a validation rate of 15% and a test rate of 35% (so half of the data presented is used for training purposes). Note that in the proposed training method, the aim is not to achieve zero distortion between input and output, as this would imply that the model would be able to precisely predict the noise of the reference image, which is not possible (apart from overfitting cases) because, according to what was presented in subsection 2.1, these are two different realizations of the same noise.

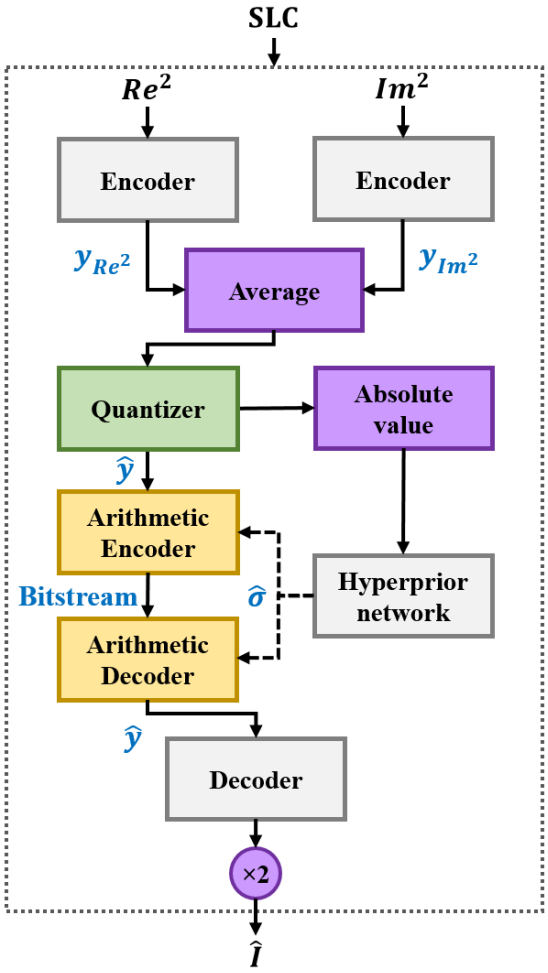


Figure 5 Proposed inference framework.

Once the training is done, the inference of the intensity $\hat{\boldsymbol{I}}$ could be done as shown in Figure 1. However, it would lead to the creation of two bitstreams conveying very similar information. Therefore we propose a new inference method, adapted to our joint approach, as illustrated in Figure 5. In order to create only one bitstream, but still offer a full power-based prediction, the real and imaginary parts squared are averaged in the latent space before the quantization. Mathematically, denoting the encoder by f_e and the decoder by f_d, it can be written as follows :

$$\hat{\boldsymbol{I}} = 2 \times f_d\left(Q\left(\frac{f_e\left(\boldsymbol{Re}^2\right) + f_e\left(\boldsymbol{Im}^2\right)}{2}\right)\right) \quad (3)$$

4 Results

4.1 Metrics

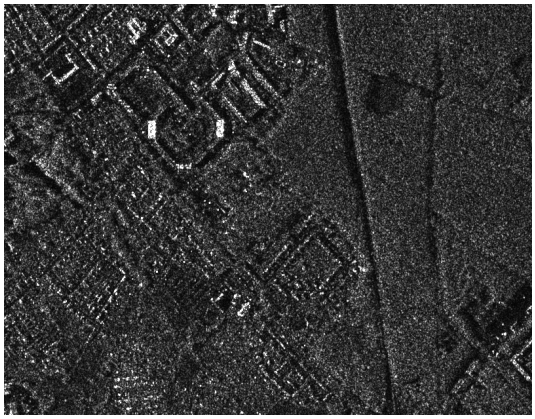
The compression and despeckling tasks were evaluated separately to assess the proposed method's effectiveness. Firstly, the compression was quantified through the bits per pixel measurement (BPP), i.e. the average number of bits needed to encode a pixel, computed for an image that the network had never seen during training or validation. Note that the value obtained will not be comparable with the results obtained in [2], as the SLC data is not comparable with the usual data in image compression. Regarding speckle reduction, the equivalent number of looks (ENL) is computed over homogeneous areas and represents how much speckle reduction has been achieved. For the said area, it is defined with μ_r as the mean value and σ_r as its standard deviation:

$$\mathrm{ENL} = \frac{\mu_r^2}{\sigma_r^2} \quad (4)$$

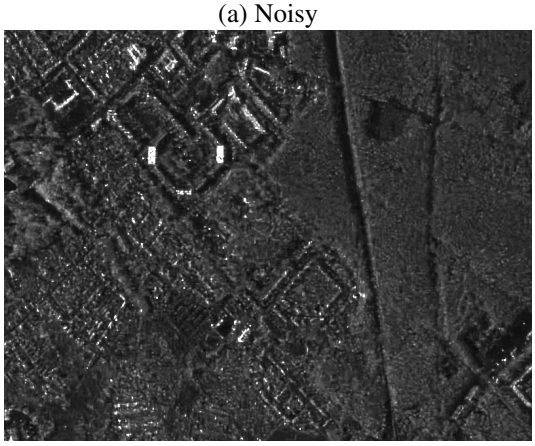
Usual denoising-related metrics such as peak signal-to-noise ratio (PSNR), mean square error (MSE) or multi-scale structural similarity (MSSIM) are not suitable since there is no noiseless reference SAR image.

4.2 Experiments

The training method described in section 3 was successfully applied, and the resulting network was used to perform the joint compression and despeckling of a test image. To compare the result obtained in terms of despeckling, two despeckling-only methods [1][3] were implemented and applied to the same image, as shown in Figure 6. The ENLs measured on the test image are respectively : (a) 0.96, (b) 4.17, (c) 130.55, (d) 266.67. Finally, the test image was encoded with only 0.366 BPP, showing very few compression artifacts. As a result, some smoothing can be seen over homogeneous areas and a good preservation of details over point targets can be observed, particularly over the buildings in the upper right corner of Fig. 6. Preliminary analysis shows good preservation of the radiometric information found in the compressed image when compared to the noisy SLC, showing similar performance to state-of-the-art methods.



(a) Noisy

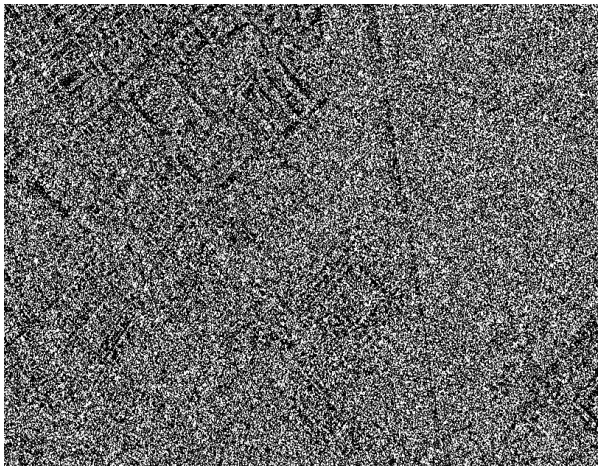


(b) SAR-BM3D (no compression)

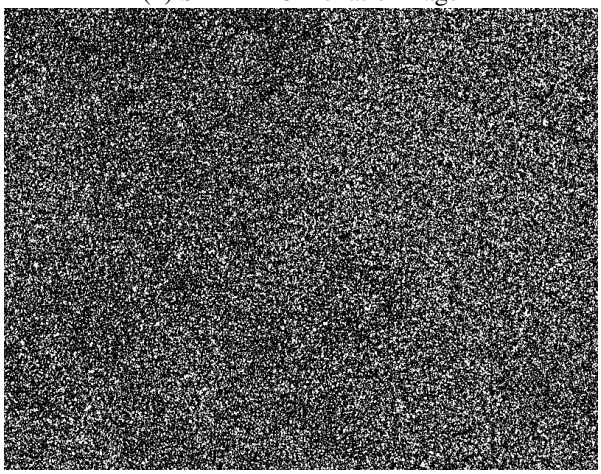
(c) MERLIN (baseline, no compression)

(d) Our result for a compressed image ($\lambda = 0.004$)

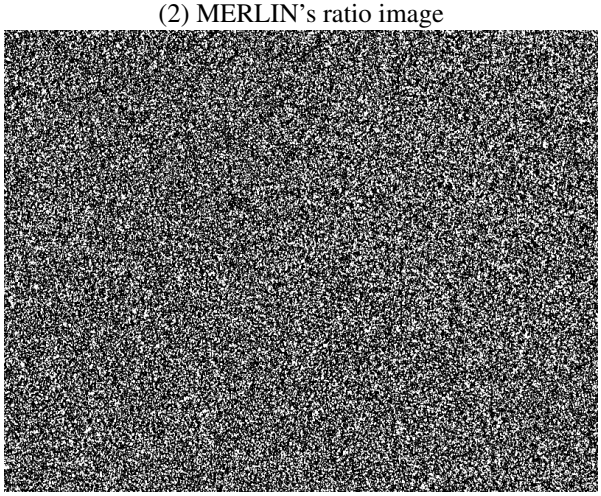
Figure 6 Comparison of results from different methods.



(1) SAR-BM3D's ratio image



(2) MERLIN's ratio image



(3) Our result's ratio image ($\lambda = 0.004$)

Figure 7 Ratio images for the three methods applied.

After computing the ratio images found in Fig. 7, we see practically no presence of structures in the ratio image of our approach in (3), in contrast to the ratio image of the SAR-BM3D in (1) where the buildings in the upper right corner and the road are clearly visible, pointing to an under- or overestimation of the resulting reflectivity. The MERLIN ratio image (2) also shows the slight presence of visible structures.

5 Conclusion

This paper presented an original joint compression and despeckling approach enabled by a framework combining two deep learning-based state-of-the-art methods that independently perform such tasks, thanks to its neural network architecture or its training strategy. Combining these methods led to the development of a framework allowing the despeckling of SAR images with near state-of-the-art results, while compressing it efficiently into a bitstream. Therefore, the proposed method may effectively be used onboard for the purpose of maximizing the data sent or of decreasing the time – and the associated cost – for such transmissions. In the future, the development of other joint approaches that can be performed onboard may be considered with regard to the downlink limited capability of current systems.

6 Literature

[1] E. Dalsasso, L. Denis, and F. Tupin, "As If by Magic: Self-Supervised Training of Deep Despeckling Networks With MERLIN", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–13, 2022.

[2] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior", ICLR 2018.

[3] 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, 2012.

[4] E. Dalsasso, L. Denis, and F. Tupin, "SAR2SAR: A Semi-Supervised Despeckling Algorithm for SAR Images", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 4321–4329, 2021.

[5] A. B. Molini, D. Valsesia, G. Fracastoro, and E. Magli, "Speckle2Void: Deep Self-Supervised SAR Despeckling With Blind-Spot Convolutional Neural Networks", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–17, 2022.

[6] E. Dalsasso, L. Denis, M. Muzeau, and F. Tupin, "Self-supervised training strategies for SAR image despeckling with deep neural networks", in *14th European Conference on Synthetic Aperture Radar (EUSAR)*, July 2022.

[7] J. Ballé, V. Laparra, and E. P. Simoncelli, "Density Modeling of Images using a Generalized Normalization Transformation", ICLR 2016.

[8] V. Alves de Oliveira et al., "Satellite Image Compression and Denoising With Neural Networks", *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.

[9] S. Vitale, G. Ferraioli, and V. Pascazio, "Analysis on the building of training dataset for deep learning sar despeckling", *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2021.