LEARNING SPECKLE SUPPRESSION IN SAR IMAGES WITHOUT GROUND TRUTH: APPLICATION TO SENTINEL-1 TIME-SERIES

Alexandre Boulch, Pauline Trouvé, Elise Koeniguer, Fabrice Janez, Bertrand Le Saux

DTIS, ONERA, University Paris Saclay, F-91123 Palaiseau, France

ABSTRACT

This paper proposes a method of denoising SAR images, using a deep learning method, which takes advantage of the abundance of data to learn on large stacks of images of the same scene. The approach is based on the use of convolutional networks, used as auto-encoders. Learning is led on a large pile of images acquired on the same area, and assumes that the images of this stack differ only by the speckle noise. Several pairs of images are chosen randomly in the stack, and the network tries to predict the slave image from the master image. In this prediction, the network can not predict the noise because of its random nature. Also the application of this network to a new image fulfills the speckle filtering function. Results are given on Sentinel 1 images. They show that this approach is qualitatively competitive with literature.

Index Terms— SAR, speckle filtering, deep learning

1. INTRODUCTION

Since radar imaging systems are coherent, radar images are corrupted by "speckle" noise and thus look visually very noisy. Denoising Synthetic Aperture Radar images is a common pre-processing and has been the object of several works. However most of existing methods rely either on a strong statistical modeling of the SAR signal, or on machine learning requiring a non-noisy ground truth or objective image, which is not always available. Moreover, the specificity of this data still makes this type of work marginal.

Whether in computer vision or radar remote sensing literature, we can generally discriminate the same categories of algorithms. The methods the most used and easy to implement are based on the observation of the local neighborhood of the pixel. This is the case with local mean or median, bilateral filtering [1] or the σ filter [2] and its SAR counterpart for multiplicative noise [3]. Another family of approaches searches for similar patches and operates a weighted sum of these patches to reduce the noise. It is the case for NL-means [4] variation for SAR data NL-SAR [5]. In BM3D [6], the authors group similar patches and denoise globally at group level. An extension to SAR, SAR-BM3D, is proposed in [7] by taking into account the multiplicative aspect of radar noise. Finally, more recent approaches tackle the denoising problem using machine learning tools, particularly deep convolutional neural networks. It is a very active field, among the papers we can quote the convolutional version of BM3D [8] or the network of [9] which only uses seven dilated convolutions and reaches very high performances. However, this type of methods makes assumptions about the noise model and has never been tested to our knowledge on SAR images. Moreover, it does not benefit from the redundancy of time series.

Today, SAR image series as Sentinel 1 become easily accessible. The amount of temporal acquisitions on a same site is a benefit to consider to improve quality of images. In this paper we investigate the gain of denoising approaches for multitemporal SAR images, in particular for change detection.

Our contribution is two-fold. First, we present a denoising method, based on the deep neural networks [9] trained without ground truth, but only using data redundancy of time series of SAR images. We also introduce a local histogram loss that improves visual performances of our networks, and compare our results with alternative methods. Second, we show that change detection algorithms can benefit from speckle filtering.

The paper is organized as follows: section 2 presents the denoising method and the training process and section 3 exposes the results of speckle removing on single image denoising and change detection.

2. AUTOENCODED DENOISER

Let X be an observed image, and X^* be the objective image, for any noise model. We also suppose that:

$$X = f(X^*, \Sigma) \tag{1}$$

where f is a noise model function, and Σ is the noise component, supposedly independent from X^*. Our goal is to create a transfer function Φ such that $\Phi(X) \approx X^*$. This Φ function is learned using our neural network. Please note that one strength of our approach is that we never use an explicit expression of the noise model.

In the following paragraph, we illustrate the method with additive noise ($X = X^* + \Sigma$) and multiplicative noise ($X = X^* * \Sigma$).

Unsupervised denoising. The usual approach to denoise by machine learning is, given an image without noise (the objective), add noise to create the noisy image and try to revert the process by a supervised machine learning method. In this study, we do not have access to X^*, so supervised machine learning techniques cannot be directly applied. To circumvent this issue, one could use supervised learning by predicting the a temporal mean over the area, but that would require a large stack, i.e large temporal horizon and changes between an image and the target may not be considered as inexistent (e.g. for vegetation). Instead we exploit the small revisit time of Sentinel-1. Given two acquisitions X_1 and X_2, for additive and multiplicative noise model:

$$X_1 = X_1^* + \Sigma_1 \text{and} X_2 = X_2^* + \Sigma_2 = X_1^* + (X_2^* - X_1^*) + \Sigma_2 \quad (2)$$

$$X_1 = X_1^* * \Sigma_1 \text{ and } X_2 = X_2^* * \Sigma_2 = X_1^* * (X_2^*/X_1^*) * \Sigma_2 \quad (3)$$

where $(X_2^* - X_1^*)$ (resp. (X_2^*/X_1^*)) is the structural change image, representing the changes between X_1 and X_2 not related to noise.

First, given the small revisit time of a Sentinel-1 time series, we suppose the change are small between two consecutive dates: $(X_2^* - X_1^*) = 0$ (resp. $X_2^*/X_1^* = 1$).

Second, considering the two noise vectors Σ_1 and Σ_2 are independent realizations of the noise process, a statistical estimator aiming at predicting X_2 given X_1 cannot do better than predict X_2^*. Following these assumptions, we train our estimator to predict X_2 given X_1.

Histogram loss. The loss criterion is the objective function of the optimization process for the previously described network. It should reach a minimum value when predicted values and objective values are similar.

ℓ_2 and ℓ_1 losses are widely used for regression task. However these losses deal with pixels independently without taking into account the local arrangement of the pixel values, i.e. if ℓ_i losses are first order penalization (value penalization), we would like a second order penalization (local order penalization).

Given a pixel $x \in X$ and its neighborhood N_x, the histogram $H_k(N_x)$ (with k bins) is a good representation of the local distribution around x. Our is simply defined by a ℓ_2 distance on the histograms vector. The gradients for back propagation is the differentiation of the previous distance.

Due to the very wide value range, the histograms are computed without scales (first bin at min value, last bin at max value). To retrieve the missing location information (value information at x), we define our loss as a weighted sum of ℓ_1 (for location) and histogram loss (for local arrangement):

$$\mathcal{L}(\Phi(X), X^*) = (1-\lambda)\|X - X^*\|_1 + \lambda\|H(X) - H(X^*)\|_2 \quad (4)$$

Inference and training processes. The input of the network is normalized (mean and standard deviation are memorized to restore the original dynamics). In order to scale up to large images, we do not feed the network with the whole image but patches. To prevent border effect, we use sub-images with an overlap of half the patch size. The result Y is then convoluted with a mask M (linear ramp with zero value at border):

$$Y = (\Phi(X) \,.\, \sigma(I) + \bar{I}) * M \quad (5)$$

where X is the input of the network, I the original input image and, $\bar{I}$ the mean of I and $\sigma(I)$ its standard deviation, and $*$ denotes the convolution.

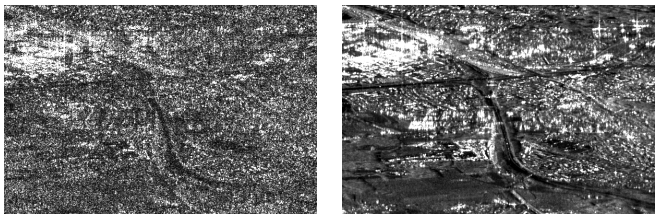
To train the network, given a stack of registered Sentinel-1, we generate on the fly patch pairs by randomly picking sub-images in one image and the time-ordered next or previous one with the same polarization. We also randomly flip or transpose the image for data augmentation.

3. EXPERIMENTS

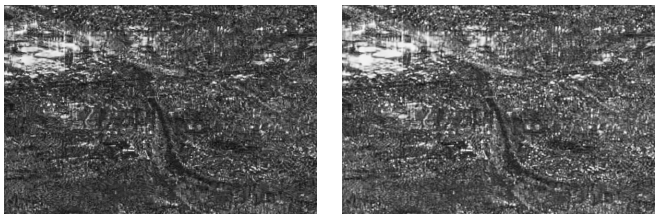
Data and networks. A Sentinel-1 stack composed of 64 images with both VH and VV polarization has been trained in a unsupervised fashion. The footprint is around Saclay at the South of Paris, images were acquired between 2015 and 2017. Results will be presented both on Saclay for the auto encoder and on another stack centered on Valencia. For the experiments we use the network from [9] (referred in the following as DCNN for Dilated Convolutional Neural Network) trained in a residual fashion. It does not operate a dimension reduction and use only seven convolutions, resulting in a lightweight network. Results will also be compared with another deep approach, Unet [10], and two classical temporal approaches based on patches but without training, BM3D and SAR-BM3D.

Denoising. Figure 1 presents different results obtained on a detail of the training set (a) and the temporal mean over all the training set (b). Except for the BM3D (c) and SAR-BM3D (d), the dynamic is the same on each image, black (resp. white) is set to the 2^{nd} percentile (resp. 98^{th} percentile) of the original image. The DCNN network has been trained with different loss functions: ℓ_2, ℓ_1 and Histogram loss ($\lambda = 0.9$, 3 bins in histograms and a radius of 3 for neighborhood).

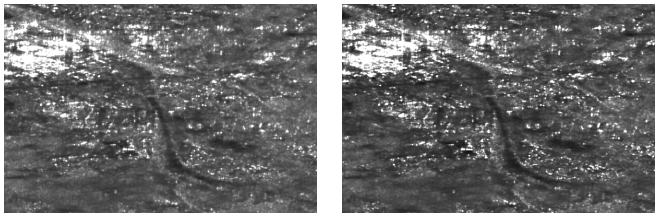
Compared to the original image, applying our network improves greatly the visual aspect of the images. The DCNN performs well with ℓ_1 (g) and Histogram loss (h). ℓ_2 loss (e) leads to more blurry images. The same method applied with a different architecture, Unet [10], trained with ℓ_1 loss performs worse than DCNN (f). The network succeeds in removing the noise while keeping the bright pixels corresponding to reflectors such as human structures. The histogram loss on DCNN removes greatly the bright halo around scatterers, sometimes hallucinating very dark pixel around the bright point.



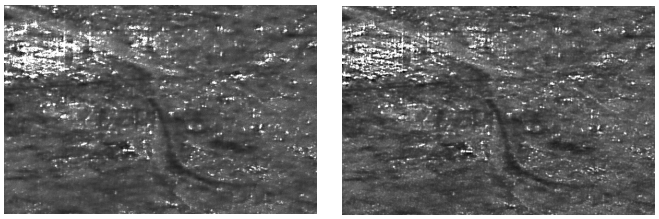
(a) Original image. (b) Temporal mean.



(c) BM3D with 10 as std. (d) SAR-BM3D.



(e) Unet, ℓ_2 loss. (f) Unet, ℓ_1 loss.

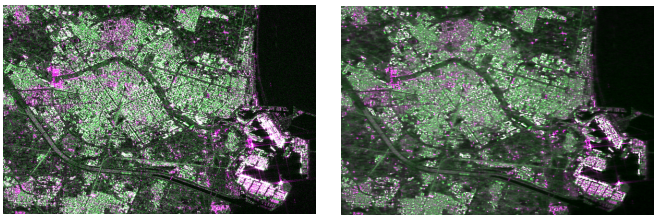


(g) Dilated CNN, ℓ_1 loss. (h) Dilated CNN, Histo. loss.

Fig. 1. Detail of Saclay stack original and processed with DCNN, different loss functions and other alternative approaches.

For comparison with existing methods, we have also run BM3D and SAR-BM3D on the same image. The standard deviation parameter of BM3D has been set to 10 and the input is the log image because it has given better results. In both cases, the result is noisier than ours. This can partly be explained by the differences between the Sentinel-1 and the data they were developed for: optical data (BM3D) and higher resolution SAR, e.g. TerraSAR-X (SAR-BM3D).

Figure 2 presents the results on Valencia, a scene not part of the autoencoding set (red and blue channels for VV and green channel for VH). Most of the noise has been removed, e.g. around the harbor. However, dense urban areas tend to be smoothed. In our opinion, this is the result of three phenomena. First, the training set is not representative: it does not contain dense urban area and has no variability in acquisition parameter such as incidence angle. Second, it only contains one scene, the network may have over-fitted. Finally, the training set is in SLC geometry while Valencia scene is GRD geometry. Still the approach proved robust to various changes in the scene.



(a) Original image. (b) DCNN, Histo. loss.

Fig. 2. Detail of Valencia original (left) and denoised (right).

Multi-temporal data processing Speckle filtering can also be considered to improve the performance of change detection or activity detection in time series. Also, we considered the impact of the DCNN filtering process for two classic scenarios: bi-date change detection and activity detection on the entire stack.

For the first case, we consider the first and the last image of the stack, we compute a RGB color composition, and also the ratio C of equation (6). For the second case, we compute the temporal coefficient of variation γ, calculated on the whole stack, equation (7), where μ and σ are the temporal mean of the backscattered amplitude, and standard deviation.

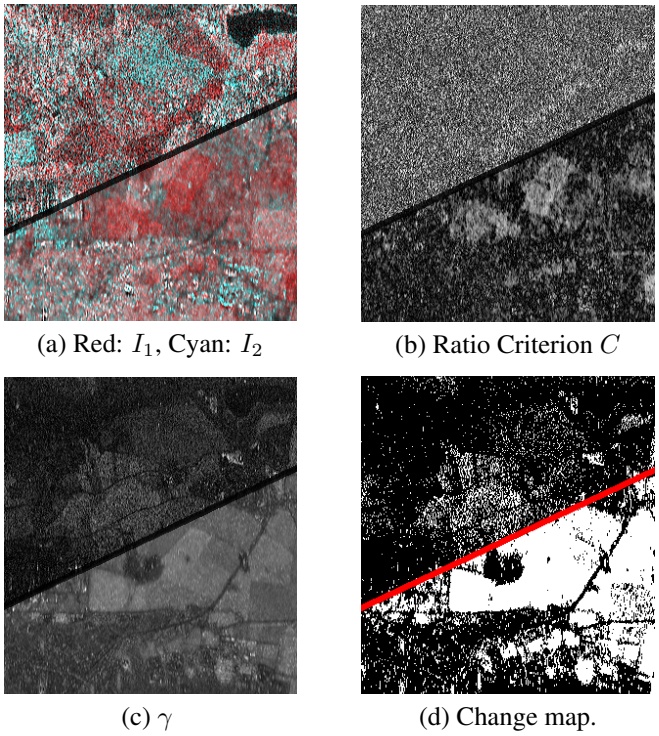
$$C = \min(\frac{X_1}{X_2}, \frac{X_2}{X_1}) \quad (6) \qquad \gamma = \frac{\sigma}{\mu} \quad (7)$$

The different results are presented in figure 3, both without and with filtering, on an area of Saclay that contains mostly cultivated fields.

Regarding the change detection between two images, the colored composition clearly shows the effects of smoothing on the various cultivated crops. The ratio criterion is much more contrasted after the filtering than before, revealing more easily changes.

The statistics of the coefficient of variation are completely modified by the filtering. It is known that for a classical speckle distribution such as Nakagami law, this coefficient is constant and depends only on the Equivalent Number of Looks. Indeed, without filtering, we find that its statistical distribution is uni-modal all over the image. It is therefore difficult to extract the changes. With filtering, the distribution of this coefficient becomes bimodal: it is then much easier to extract changes by thresholding, mainly the fields. In any case, removing the noise greatly helps the interpretation, particularly for changes on natural areas such as agricultural areas. Finally, figure 3(d) shows the change map after thresholding using simple mean / standard deviation for raw data

and Otsu's method [11] for the bimodal filtered product.



(a) Red: I_1, Cyan: I_2 (b) Ratio Criterion C

(c) γ (d) Change map.

Each thumbnail contains both a result without filtering on the top and with filtering on the bottom. (a) Image Pair composition. Red: I_1, cyan: I_2, (b) Ratio Criterion between I_1 and I_2, (c) Temporal Coefficient of Variation on N images, (d) Threshold on (c).

Fig. 3. Detail of Saclay products.

4. CONCLUSION

An original method to learn a denoising convolutional neural network has been presented. The network is trained in an autoencoder fashion by trying to predict another image of the same temporal stack. The network fails to predict the random noise of the objective image, producing a denoised version of the input image. It is therefore envisaged to extend the training set from one time series to multiple in different places (urban, agricultural, mountains ...) and acquisition conditions (incidence, sensors) to improve robustness and generality of the approach. We also plan to test simulated data allowing us to provide quantitative results.

Acknowledgements This work is supported by two research programs at ONERA: MEDUSA[1], remote sensing images processing in big data context and DeLTA[2], machine learning for aerospace applications.

[1]MEDUSA project at `w3.onera.fr/medusa`

[2]DeLTA project at `delta-onera.github.io`

5. REFERENCES

[1] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Computer Vision, 1998. Sixth International Conference on*. IEEE, 1998, pp. 839–846.

[2] J. S. Lee, "Digital Image Enhancement and Noise Filtering by Use of Local Statistics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 2, no. 2, pp. 165–168, Feb. 1980.

[3] J. S. Lee, J. H. Wen, T. L. Ainsworth, K. S. Chen, and A. J. Chen, "Improved Sigma Filter for Speckle Filtering of SAR Imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 1, pp. 202–213, Jan 2009.

[4] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. IEEE, 2005, vol. 2, pp. 60–65.

[5] C. A. Deledalle, L. Denis, F. Tupin, A. Reigber, and M. Jger, "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, April 2015.

[6] 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, 2007.

[7] S. Parrilli, M. Poderico, C. V. Angelino, and L. Verdoliva, "A Nonlocal SAR Image Denoising Algorithm Based on LLMMSE Wavelet Shrinkage," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 50, no. 2, pp. 606–616, Feb 2012.

[8] D. Yang and J. Sun, "BM3D-Net: A Convolutional Neural Network for Transform-Domain Collaborative Filtering," *IEEE Signal Processing Letters*, vol. 25, no. 1, pp. 55–59, Jan 2018.

[9] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *IEEE, CVPR*, July 2017.

[10] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[11] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.